CONTRACT REPORT ARBRL-CR-00420

SRI PUFF 8 COMPUTER PROGRAM FOR
ONE-DIMENSIONAL STRESS WAVE PROPAGATION

Prepared by

SRI International
333 Ravenswood Avenue
Menlo Park, CA 94025

March 1980
SRI PUFF 8 is a Lagrangian finite difference computer program for calculating one-dimensional stress wave propagation through solid, liquid, gaseous, and porous materials. The stress waves may be caused by the deposition of radiant energy, impact of materials, explosive detonation, or by a prescribed pressure or velocity history.

The calculational procedure is the standard leapfrog method of von Neumann and Richtmyer using artificial viscosity to smooth shock fronts. Rezoning and

(cont'd)
material separation by spall are permitted. Planar, cylindrical, and spherical flow are treated.

The constitutive relations include the standard Mie-Grüneissen equation of state and elastic, plastic (Mises or Coulomb) work-hardening deviator stress relations with thermal softening. Other pressure relations provided are a polytropic gas for explosives, GRAY and Philco-Ford three-phase equations of state, and a tabular pressure-volume relation. Special deviator stress models include the standard linear viscoelastic model, a Bauschinger model, dislocation models, and a nonlinear work-hardening model. Ductile and brittle fracture and shear banding are provided by nucleation and growth models. Porous materials may be represented by the Seaman-Linde model, Holt model, Herrmann P-α model, a cap plasticity model, a variable modulus model, Butcher P-α-τ model, or by a linear viscous void compaction model. A model for layered composites is also present.

The code is constructed for easy insertion of additional material models. The number of extra variables required for each cell for a material model can be specified in the input deck.

This manual includes many sample problems, a derivation of the flow equations, discussion of material models, and an outline of other aspects of wave propagation calculations.
FOREWORD

This volume constitutes a portion of the three-volume final report to Ballistics Research Laboratory on Contract DAAK11-77-C-0083, SRI Project 6802. Volume I reports on ballistic experiments and calculations, and describes work on the latest version of the SRI brittle fracture subroutine. Volume III is the manual for the two-dimensional wave propagation code TROTT.
CONTENTS

FOREWORD iii

LIST OF ILLUSTRATIONS vii

LIST OF TABLES xi

1. INTRODUCTION 1
   1.1 Background 1
   1.2 Scope 2

2. ORGANIZATION OF THE CODE 5
   2.1 Summary 5
   2.2 Main Program: SRI PUFF 8 9

3. PROPAGATION CALCULATIONS: HYDRO GROUP 11
   3.1 Solution Procedure for Wave Propagation Equations 11
   3.2 Pressure-Energy Calculation 16
   3.3 Artificial Viscous Stress 18
   3.4 Time-Step Control 20
   3.5 Outline of Subroutines 26

4. CONSTITUTIVE RELATIONS 29
   4.1 Standard Constitutive Models 29
      4.1.1 Standard Pressure Models 29
      4.1.2 Standard Deviator Stress Model 42
   4.2 Constitutive Model Types and Switching Routine 47
      for Selecting Models
   4.3 Spall Calculations 52

5. INITIALIZATION: THE GENRAT GROUP 55
   5.1 Input of General Running Information 56
   5.2 Input of Material Properties 59
   5.3 Layers and Cell Layout 61
   5.4 Thermal Energy Deposition 67
      5.4.1 Deposition Types 67
      5.4.2 Data Required 68
      5.4.3 Special Features 70
   5.5 Initialization of Arrays and Indicators 71
   5.6 Initial Status Printouts 74

6. PRINTED OUTPUT: GENRAT, EDIT, AND SCRIBE 77
APPENDIXES

A  THERMAL ENERGY DEPOSITION  87
B  CALCULATIONS FOR EXPLOSIVES  105
C  DESCRIPTION OF INPUT  111
D  FMELT: THERMAL REDUCTION FUNCTION  147
E  RESIZING THE CELLS: REZONE  151
F  ONE-DIMENSIONAL CYLINDRICAL AND SPHERICAL FLOW  159
G  DEVIATOR STRESS MODELS  169
H  INSERTION PROCEDURE  179
I  LISTING OF SRI PUFF 8  183
J  GLOSSARY  319

REFERENCES  331

DISTRIBUTION LIST  335
ILLUSTRATIONS

2.1 Flow Chart of SRI PUFF 8 6
3.1 Grid for Depicting Coordinates and Time Increments 14
3.2 X-t Plot for the Time Step Computation 22
4.1 Pressure Paths for Shock Loading and Unloading of a Material 31
4.2 Energy-Pressure-Volume (E-P-V) Surface for a solid Material 31
4.3 Loading and Unloading Paths for Pressure and for Stress in the Direction of Propagation 33
4.4 Pressure-Volume Relations at Constant Internal Energy for an Aluminum 38
4.5 Sketch of Mie-Grüneisen and modified PUFF-Expansion Equation-of-State Model 39
4.6 Effects of Yield and Modulus Thermal-Softening Functions on Pure Shear Stress-Strain Relations 48
4.7 Flow Chart of HSTRESS, Stress-Switching Routine 50
5.1 Coordinate Layout for SRI PUFF 62
5.2 Coordinate Layout Showing Variable Cell Size Capability 63
5.3 Typical Variation of Photoelectric Mass Absorption Coefficient with Photon Energy 69
5.4 Initial Status Output from GENRAT for an impact of a 1145 Aluminum Plate onto a Hot Aluminum Plate at 1.46 x 10^4 cm/sec 76
6.1 Sample GENRAT Listing of Input Data for a Radiation Problem 78
6.2 Sample GENRAT Listing of the Cell Layout for an Impact of a Steel Projectile onto Reinforced Concrete 80
6.3 Sample DEPOS Listing of the Cell Layout for a Radiation Problem 81
6.4 Sample Listing of Output from PRESCR Showing Variables for which Historical Listings Will be Made 82
6.5 Sample EDIT Listing of Cell Variables at One Time During the Calculation 83
6.6 Sample Historical Listing of Variables from SCRIBE 84
6.7 Miscellaneous Messages and Listings Generated during Calculations 85
A.1 Typical Variation of Photoelectric Mass Absorption Coefficient with Photon Energy
A.2 History of Radiation Sources Considered in the Program
A.3 Emittance Spectrum for Black Body
A.4 Spectrum of a Radiation Source
A.5 Penetration of Energy into a Material
A.6 Radiation Intensity as a Function of Depth for Several Incident Angles
A.7 Patterns for Interpolation of the Radiant Energy in a Depth-Dose Profile to Obtain Energies for PUFF Cells
B.1 Pressure-Volume Paths Followed in Detonation Processes
C.1 Data Bank Containing General Running Information, Material Property Data, and a Spectrum (on Tape 4)
C.2 Data Bank Containing X-ray Absorption Data for Several Elements (on Tape 2)
C.3 Input Deck for Hot Aluminum Impact Calculation
C.4 GENRAT Output for Input Deck of Figure C.3
C.5 Data Deck and Data Bank, Illustrating Use of Data Banks for Radiation Problems with Multiconstituent Materials
C.6 Input Constructed by GENRAT for Radiation Problem in which General Running Information, Material Properties, Radiation Absorption Data, and Spectrum are on Data Banks
C.7 Input Deck for Impact in 1145 Aluminum, Illustrating Ductile Fracture Data and JEDITS with No Type Indicator
C.8 Input Deck for Impact in ARMCO Iron, Illustrating Brittle Fracture and a Geometric Cell Layout
C.9 Input Deck for a Symmetric Impact of Tonalite, Showing Coulomb Friction with Tan \( \phi = 0.056 \), Multiple Zones in a Layer, and Use of the POREQST Model
C.10 Input Deck for Impact of a Steel Plate onto Reinforced Concrete, Showing the Use of CAP and REBAR Subroutines and Multiple Layers of a Single Material
C.11 Input Deck for the Cylindrical Calculation of a Fragmenting Round, Showing Detonation of an Explosive, Shear Band Model, and English Units in the Layout
C.12 Input Deck for the Cylindrical Calculation of a Fragmenting Round of HFI Steel, Showing Detonation of an Explosive
C.13 SIGMAT and Input Deck for a Cylindrical Calculation, Illustrating a Hollow or Empty First Layer, Pressure Boundary, and Use of POREQST
C.14 Input Deck for Spherical Explosion of PETN in Rock Matching Group, Showing Use of a Tabular Equation of State, Cap Model, and a NAMELIST Statement 136
C.15 Input Deck for Explosively Thrown Flyer Plate Impacting OTWR and Illustrating the Use of Explosive, NAMELIST, Comments, and Gaps in the Layers 137
C.16 Input Deck for a Running Detonation Through Three Explosives, Illustrating that PUFF Permits Overdriving of Explosion 138
C.17 Input Deck for Radiation into Beryllium and Alumina, Showing the Use of a Depth-Dose Profile, Geometric Layout, and Automatic Rezoning 139
C.18 Input Deck for Radiation into Aluminum, Showing a Depth-Dose Profile and Multiple Zones in One Layer 140
C.19 Input Deck for Radiation from Three Black Bodies into Aluminum, Showing Four Rezones and Multiple Zones in One Layer 141
C.20 Input Deck for Radiation by a Depth-Dose Profile into Several Layers and Illustrating Use of HDATA and TEDIT 142
C.21 Input Deck for Pressure Loading on a Three-Layered Plate, Showing Use of the NAMELIST Statement 143
C.22 Input Deck for Simulating an Air Shock by Applying a Pressure Boundary Through NAMELIST 144
D.1 Variation of Strength with Temperature for Aluminum 1100 148
D.2 Definition of Input for Thermal Strength and Modulus Reduction Function 149
E.1 Layout for Computing Properties at Rezoned Coordinates 152
E.2 Representative Layout of Coordinates before Rezoning 153
E.3 Rezonable Set of Cells Terminated on Left by a JEDIT 153
F.1 Cell Geometry Considered for One-Dimensional Spherical Momentum Calculations 160
G.1 Vectorial Representation of Principal Strains during an Interval of Plastic Flow with Work Hardening 174
TABLES

C.1 Definitions of Directive Groups ................................................. 123
C.2 Sample Input Directives ......................................................... 125
F.1 Stress and Strain Quantities in One-Dimensional Flow ................. 165
H.1 Material Parameter Array ESC ................................................ 181
1. INTRODUCTION

The SRI PUFF code is a computer program for calculating one-dimensional stress wave propagation through solid, porous, liquid, or gaseous materials. The stress waves being computed are initialized by the deposition of radiated energy from x-ray, electron beam, or laser sources; impact of one material on another; detonation of an explosive; or by prescription of a pressure or velocity history at a boundary. Computations are made with the Lagrangian form of the equations of motion so that the coordinates move with the materials. An artificial viscosity is used to smear wave fronts over several computational cells.

1.1 Background

In 1950 von Neumann and Richtmyer (Ref. 1) initiated the artificial viscosity (or Q) method for solving the equations of wave propagation. With this technique infinitely steep shock fronts cannot develop, and the entire field can be treated as one of continuous flow. Shock fronts appear as regions of high stress gradient, not as discontinuities. The viscosity tends to dampen all oscillations or perturbations in the flow field. Several integration schemes based on the Q method have been developed, notably the Lax-Wendroff method (Ref. 2), the Runge-Kutta-Gill method (Ref. 3), and the "leapfrog" scheme (Ref. 1) which is used by most PUFF codes.

The present line of PUFF codes seems to have originated around 1958 with the development of the SHARK (Ref. 4) and SHARP (Ref. 5) codes. With later developments at the Air Force Special Weapons Center, Kirtland Air Force Base, the generic name PUFF was given to the program. Recent versions include PUFF (Refs. 6-8), PUFF III (Ref. 9), PUFF IV (Ref. 10), PUFF IV-EP (EP for elastic-plastic), (Ref. 11), PUFF V-EP (Ref. 12), PUFF VTS (variable time step), (Ref. 13) FOAM PUFF (Ref. 14), PUFF 66 and P PUFF 66 (Ref. 15).
Most of the PUFF codes have been described in classified reports, so their characteristics cannot be outlined here. A useful review of the capabilities of each of these codes has been provided by Bothell and Archuleta (Ref. 11). Other PUFF-type codes are available under the names of WONDY (Ref. 16), SRI PUFF (Ref. 17), and RIP (Ref. 18). RIP is a well-documented code with special capabilities including detailed treatment of composite materials and laser deposition. All the PUFF-type codes use artificial viscosity with the leapfrog integration scheme. The SRI PUFF series of codes began as a modification of the PUFF 66 and P PUFF 66 codes.

1.2 Scope

This volume outlines the essential theory on which the wave propagation calculations of the SRI PUFF series of computer programs is based and describes some of the constitutive models (stress-strain relations) currently available. The constitutive models include several that provide deviator stress only, several for pressure only, and several that provide a combination of pressure and deviator stress. The descriptions given here outline the simplest constitutive models only, indicate sources for information on the others, and show how to insert additional constitutive models.

The current version of SRI PUFF includes the features of earlier versions plus provisions for cylindrical and spherical flow as well as one-dimensional planar flow; use of a data bank; ductile and brittle fracture, fragmentation, and shear banding; several porous material models; a hypoelastic (variable modulus); a cap (advanced plasticity) model; detonation by constant volume explosion or by running detonation; improved rezoning; and Coulomb-friction without dilatation.

The code calculations make use of both linear and quadratic artificial viscosity. An integral approach is used to solve the mass and energy conservation relations. The stress is determined from the equation of state or constitutive relations for known volume and energy. Because the energy is not known at the time stress must be calculated, an energy estimate is made and then adjusted after the stress calculation.
Since its outgrowth from PUFF 66 in 1967, SRI PUFF has undergone many changes and is expected to undergo more. The code is written in a modular form so that initialization and running are usually separated, deposition problems use subroutines that are unused for other runs, and constitutive relations are in separable subroutines. Thus the code is planned for ease of change. Subroutines for new constitutive relations can be added as new material models are generated.

This manual is intended to assist not only the users of the program, but also those who wish to understand it well enough to modify it, and those who wish to investigate the analytical basis of the program. For users, the chapter on Initialization (Section 5), and the Appendices C (Input) and J (Glossary) will be of primary interest. Alterers of the program may notice the following features: a brief description of each subroutine in Section 2 and a discussion of major subroutines is at the end of Sections 2 through 6. For the analyst, the bases of the program are discussed in Sections 3 through 5, which is organized around certain fundamental problems in the program: initializing, integration of the propagation equations, equations of state, and so forth. The order of presentation is general theory first, then application to the current analysis, and finally details of implementation in the program. It is hoped that this organization will provide answers to specific questions about the program.
2. ORGANIZATION OF THE CODE

2.1 Summary

SRI PUFF 8 is a one-dimensional Lagrangian hydrodynamic program for the computation of stress waves caused by impact, radiation deposition, detonation of an explosive, or prescription of a stress or particle velocity at a boundary. The numerical integration of the governing equations is carried out with the leapfrog method of von Neumann and Richtmyer. The computations proceed by increments of time. For each increment, a cycle of computations is made throughout the active regions of the materials to determine stress, particle velocity, specific internal energy, density, sound speed, yield strength, pressure, coordinate location, and other variables. The primary routines of the program are SRI PUFF 8 (overall control), GENRAT (initialization), HYDRO (control of wave propagation calculations for each cell), HAFSTEP (density and energy calculations), and HSTRESS (control of stress calculations).

The flow of program control is illustrated schematically in Figure 2.1, which shows the interrelationship between the subroutines and the main program. The subroutines are grouped according to type of activity. Thus the GENRAT group (GENRAT plus all subroutines with arrows from GENRAT) initializes and the HYDRO group (HYDRO, HAFSTEP, and HSTRESS) treats propagation and stress calculation. The arrows designate direction of calling. A brief description of the work of each subroutine follows:

- SRI PUFF 8, the main program, sets the size of each time increment, calls HYDRO to perform a cycle of computations, and calls for printout and resizing of cells.
- BANDRLX computes deviator stresses according to the Band or Gilman stress-relaxation models (see Ref. 19).
- BAUSCHI computes deviator stresses from a Bauschinger model (see Refs. 19, 20).
- BECOM and BEMOD, in combination with STRESS, compute deviator stress for beryllium according to a stress-relaxing, Bauschinger model (see Ref. 21).
FIGURE 2.1 FLOW Chart OF SRI PUFF 8

*Starred stress-strain routines are also called by GENRAT for initialization.
- BFRACT computes stress and crack sizes in material undergoing brittle fracture and fragmentation (see Refs. 22-25).
- CAP1 computes stress and tensile fracture in materials with a combined Mohr-Coulomb yield and compaction behavior (see Ref. 33).
- DEPOS controls deposition of radiant energy into the cell layout during initialization (see Section 5.4, Appendix A).
- DFRACT computes stress and void growth in material undergoing ductile fracture (see Refs. 23, 26).
- EDIT prints a listing of velocities, stresses, and other variables at specified times (see Section 6).
- EOSTAB computes pressure from a table of pressures as a function of density and energy.
- EPLAS computes elastic plastic behavior of the reinforcing steel treated in the REBAR subroutine (see Ref. 33).
- EQST provides the Mie-Gruneisen and PUFF expansion equations of state for determining pressure (see Section 4).
- EQSTPF contains the Philco-Ford equation of state, which treats explicitly solid, liquid, and gaseous as well as mixed phases (see Refs. 27, 28).
- ESA is an equation of state written in a form that is easy to fit to experimental data (see Ref. 28).
- EXPLODE provides the equation of state for explosives and for constant volume or running detonation (see Appendix B).
- EXTRA reads in additional input outside the normal set (see Appendix C).
- FMELT computes the variation of strength with temperature (see Appendix D).
- GENRAT reads or controls input, and initializes arrays and indicators (see Section 5).
- GRAY provides the Gray equation of state, which treats explicitly solid, liquid, gaseous, and mixed phases (see Refs. 28, 29).
- HAFSTEP computes density and estimates internal energy, then calls HSTRESS for the stress calculation (see Section 3).
- HDATA reads extra input lines for initializing the H(J,I) indicator array.
- HSTRESS computes the stresses through calls to appropriate subroutines. All constitutive relations are reached through the calls by HSTRESS (see Sections 3, 4).
- HYDRO conducts each cycle of calculations through the coordinate array, computes coordinate location and velocity, and calls HAFSTEP for midcell calculations (see Section 3).
- HYPO computes pressure and deviator stress from a variable modulus or hypoelastic stress-strain relation (see Ref. 30).
- PEST provides a stress-strain relation for porous materials, including strain-rate effects, hysteresis, thermal strength reduction, and fracture (see Ref. 28).
- POREQST computes pressure in a porous material, allowing for hysteresis and thermal strength reduction (see Ref. 31).
- PORHOLT computes pressure in a porous material according to the Holt curve for compaction (see Refs. 32, 28).
- PRESCR initializes the indicators required to obtain historical listings (see Appendix C).
- REBAR computes stresses in a layered composite such as reinforced concrete (see Ref. 33).
- REDR positions the tape for reading when input is from a tape file (see Appendix C).
- RELAX computes relaxation of the deviator stress for the anelastic model and a two-parameter, varying yield model (see Ref. 19).
- REZONE resizes the cells and recomputes all coordinate quantities (see Appendix E).
- SCATTO distributes the radiated energy of a depth-dose profile into the cells of the PUFF layout (see Appendix A).
- SCRIBE stores historical data during the computation and provides stress histories at selected coordinates and at each material interface at the end of each computation (see Section 6).
- SHEAR2 contains stress-strain relations for material undergoing yielding and fragmentation by shear banding (see Refs. 34, 35).
- SIGMAT provides a pressure history for a boundary condition.
- SSCALH computes the energy deposited at midcell points during each time increment in which radiation is occurring (see Appendix A).
- STORR stores variables during the calculation for the historical listing (see Appendix C).
- STRES2 computes the deviator stress for beryllium from a stress-relaxing, Bauschinger model (see Refs. 21, 36).
- TSQE provides a computation of density from the Mie-Gruneisen equation of state, given the pressure and energy (see Ref. 28).
2.2 Main Program: SRI PUFF 8

The main program controls sequencing of the operations of initialization, calculation, printout, rezoning, and stopping of the program. It also governs the time increment. The order of operations in the main program is as follows:

1. Call GENRAT to read data and initialize COMMON storage.
2. Call HYDRO to make computations of all array variables at each time increment.
3. Call STORR to store data from HYDRO cycle for later printout.
4. Check whether the program should be terminated because:
   a) the problem time (TIME) has exceeded the specified stop time TS;
   b) the number of cycles N has exceeded the specified total number of cycles JCYCS;
   c) the coordinate of the zone of maximum stress has exceeded the specified coordinate CKS;
   d) LSUB(7) has been set to 1 because of an error detected in the computations. If termination is indicated, SCRIBE is called to print a history of stresses. Then the program returns to step 1 to read in the next data deck. If termination is not called for, the program continues to step 5.
5. Calculate next time increment DTNH.
6. Call EDIT for printout if TIME equals one of the TEDITS (input quantities).
7. Call REZONE if the TIME equals a TEDIT time designated for rezoning or if N is a cycle designated for periodic rezoning.
8. Prepare for the next EDIT listing. (After completion of this sequence, the program returns to step 2 for the next call to HYDRO.)

The time increment is based on the minimum of the natural time steps allowed (for stability of the calculations) at any point in the mesh. This calculation of permitted time step is described in Section 3.4 on Propagation. The time increment is initialized in GENRAT at $10^{-12}$ second for the first cycle. Thereafter, the time step increases gradually in successive cycles, to 80% of the natural time step. The increment is never required to be less than 2.8% of the natural time step; then, if a short increment occurs, the increment returns to its normal value within 20 cycles.
To ensure that an adequate number of cycles occurs during the radiation deposition, the time increment during deposition is required not to exceed 0.03 times the duration of any currently active radiation sources. After deposition is complete (\( \text{TIME} > \text{SSTOPM} \)), \( \text{SDURM} \) is reset to 1.0 to indicate that the radiation time step control should be skipped.
3. PROPAGATION CALCULATIONS: HYDRO GROUP

The motion and stresses throughout the material are determined as a function of time in the code. The solution is obtained by solving the mass, momentum, and energy conservation relations together with constitutive relations for the material. This section presents the conservation relations and their general solutions and shows specific solutions for interior points and boundaries of material layers.

In the solution procedure, the material is first divided into discrete units or cells. Motions, energies, and other quantities are initialized in cells as required for the particular problem. Then a time step is taken and the motions and stresses are calculated for each cell using the conservation and constitutive relations. This process of stepping forward in time and performing calculations for each cell is repeated until the time has reached the duration of interest. The time step used is controlled by stability and smoothness criteria in the code. The stability considerations are described in this section. At the end of the section, the major work of the HYDRO group (HYDRO, HAFSTEP, HSTRESS) is summarized.

3.1 Solution Procedure for Wave Propagation Equations

The PUFF programs are all based on the solution of the Lagrangian equations governing one-dimensional motion of a continuous medium. The solution technique is called the method of artificial viscosity because of the introduction of viscous forces to permit a continuous-flow computation in regions of high-stress gradients. Such regions are interpreted as locations of shock fronts, although no discontinuities occur in the computed flow field. With this artificial viscosity method, the equations of continuous flow can be used everywhere and no special equations are required for shock fronts. SRI PUFF uses the leapfrog method of von Neumann and Richtmyer to integrate the flow equations.
The following paragraphs introduce the governing differential equations for planar flow. These are changed to an integral form for solution in the program. The corresponding equations for one-dimensional cylindrical and spherical flow are given in Appendix F.

The one-dimensional planar Lagrangian differential equations to be solved are

\[
\begin{align*}
\frac{\partial U}{\partial t} &= -\frac{1}{D_0} \frac{\partial R}{\partial H} \\
\frac{\partial X}{\partial t} &= U \\
\frac{\partial D}{\partial t} &= -D^2 \left(\frac{\partial U}{\partial X}\right) \\
\frac{\partial X}{\partial H} &= D_0/D \\
\frac{\partial E}{\partial t} &= -R \left(\frac{\partial U}{\partial H}\right) + \left(\frac{\partial E_{\text{rad}}}{\partial H}\right) \\
\frac{\partial E}{\partial t} &= -R \left(\frac{\partial V}{\partial H}\right) + \left(\frac{\partial E_{\text{rad}}}{\partial H}\right)
\end{align*}
\]

where

- \( H \) = Lagrangian coordinate location (original position in laboratory coordinates)
- \( X \) = Eulerian coordinate location (current position in laboratory coordinates)
- \( t \) = time
- \( U \) = particle velocity
- \( D, D_0 \) = current and original density
- \( R \) = total mechanical stress
- \( E \) = internal energy
- \( E_{\text{rad}} \) = radiated energy
- \( V = D^{-1} \) = specific volume
These equations relate velocity to the coordinate motion and provide for conservation of momentum, mass, and energy. In addition to these differential equations, there is an equation of state (or constitutive relation (which is a relationship between stress or pressure quantities and the density, internal energy, history of loading, and so forth).

\[ R = F(E, D, \ldots) \text{ (equation of state)} \quad (3.5) \]
\[ = P + \sigma' + Q \quad (3.6) \]

The total mechanical stress (in the direction of propagation), \( R \), is composed of the pressure \( P \), the deviator stress \( \sigma' \) in the direction of propagation, and an artificial viscous stress, \( Q \).

In the code the five preceding equations are solved simultaneously by dividing the material into small elements. Then the quantities \( X, U, D, R, E \), and so forth, are evaluated only at the discrete positions and times shown in Figure 3.1. The coordinate quantities \( X \) and \( U \) are obtained at integral values of \( j \) and \( n \), whereas all other quantities pertain to the midcell \( (j+\frac{1}{2}, n+\frac{1}{2}) \) points. Here the cells are treated as constant strain finite elements (each cell has a constant value of all three principal strains throughout its volume). This derivation contrasts slightly with the finite difference approach normally used, but the resulting equations differ only for cylindrical and spherical flow (see Appendix F).

The discrete values of the flow quantities are obtained from Eqs. (3.1) through (3.4), using the nomenclature of Figure 3.1. Here it is convenient to solve for quantities in the order \( D, E, R, U \), and \( X \). The density is obtained from conservation of mass by dividing the stored value of the cell mass, \( Z \), by the thickness of half time, \( t_{n+\frac{1}{2}} = t_n + \frac{1}{2}At^{n+\frac{1}{2}} \). The first form of Equation (3.3) is not used here because it can give erroneous results for large density changes; instead, the second form of Equation (3.3) is used:
FIGURE 3.1  GRID FOR DEPICTING COORDINATES AND TIME INCREMENTS
\[ D_{j+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{Z_{j+\frac{1}{2}}}{X_{j+1}^{n} - X_{j}^{n} + \frac{\Delta t_{n+\frac{1}{2}}}{2}(U_{j+1}^{n} - U_{j}^{n})} \] (3.7)

The energy conservation relation is also used in integral form rather than relying on the differential form of Eq. (3.4). As shown in the second form of Eq. (3.4), the strain energy term is the stress times the volume change.

\[ E_{j+\frac{1}{2}}^{n+\frac{1}{2}} = E_{j+\frac{1}{2}}^{n-\frac{1}{2}} + \left( \frac{1}{p_{j+\frac{1}{2}}} - \frac{1}{p_{j-\frac{1}{2}}} \right) R_{j+\frac{1}{2}}^{n-\frac{1}{2}} + (\Delta E_{j+\frac{1}{2}}^{n}) \text{ rad} \] (3.8)

For correct centering of the equations, the stress quantity here should be \( R_{n} \) obtained by averaging \( R_{n-\frac{1}{2}} \) and \( R_{n+\frac{1}{2}} \). However, \( R_{n+\frac{1}{2}} \) is obtained in the next step; hence, Eq. (3.8) is only the first approximation to the energy. The complete procedure for obtaining energy is described in Section 3.2. The stress is next calculated with a constitutive relation represented by Eq. (3.5). Some of the available constitutive relations are described in Section 4.

The velocity is obtained by a discretization of Eq. (3.1), or equivalently, by using "force equals mass times acceleration": and considering a mass pertaining to the \( j \) coordinate point.

\[ U_{j}^{n+1} = U_{j}^{n} - \frac{R_{j+\frac{1}{2}}^{n+\frac{1}{2}} - R_{j-\frac{1}{2}}^{n+\frac{1}{2}}}{(Z_{j+\frac{1}{2}} + Z_{j-\frac{1}{2}}) / 2} \Delta t_{n+\frac{1}{2}} \] (3.9)

Finally, the Eulerian position of the coordinate is computed from Eq. (3.2)

\[ X_{j}^{n+1} = X_{j}^{n} + \frac{1}{2} \left( U_{j}^{n+1} + U_{j}^{n} \right) \Delta t_{n+\frac{1}{2}} \] (3.10)

The computations proceed from left to right, one cell and coordinate at a time, updating the flow quantities to the new time \( t_{n+\frac{1}{2}} \) or \( t_{n+1} \), as appropriate. This process is continued until the right boundary is reached. Then computations resume at the left for the next time increment.
The foregoing integration method is essentially the leapfrog method of von Neumann and Richtmyer. With this approach the derivatives in the equations of mass, momentum, and energy are correctly centered. That is, each of the conservation relations is replaced by a numerical approximation in which all terms pertain to the same point in time and space. For example, in the momentum equation (3.9), ∂U/∂t and ∂R/∂Z are both centered precisely at \((n+\frac{1}{2},j)\), and therefore, the solution scheme is of second order, although no numerical approximations to ∂²U/∂t² or ∂²R/∂Z² are needed.

In the code, the names of quantities are essentially those given above in the discretized equations. The coordinate quantities are \(U(J) = U_{n+1}^j\) and \(X(J) = X_{n+1}^j\), and the cell quantities are of the form \(RHL(J) = R_{n+\frac{1}{2}}^j\). The time step is \(\Delta t_{NH} = \Delta t_{n+\frac{1}{2}}\). Hence the coordinate point and the cell to the right are both labeled \(J\), and the midcell quantities at \(n+\frac{1}{2}\) and the coordinate quantities at \(n+1\) are stored in the arrays. Boundaries between materials are treated in the same fashion as coordinates within a material except that an extra coordinate is provided to permit separation of the layers.

3.2 Pressure-Energy Calculation

A special solution method for obtaining stress and energy simultaneously was necessary to permit use of arbitrarily complex equations of state. The set of equations governing wave propagation includes expressions for pressure as a function of energy and density and for energy as a function of stress and density.

\[
P = P(E, \rho)
\]

\[
E = E_0 + \int_0^{\rho \rho_0} \rho \, d\rho + \Delta E_r
\]

(3.11)  

(3.12)

where \(\Delta E_r\) is radiant energy. These expressions may be solved simultaneously as in WONDY16 if the pressure function is linear in energy, by multiple calls to the equation-of-state routine as in PUFF 66,15 or by extrapolation as in a two-step integration scheme.2,17 A combined extrapolation and simultaneous solution method was developed for use in the
current one-step integration scheme of SRI PUFF 8. First we estimate the internal energy at the current step. This energy is used to compute the stress. Then these provisional values of stress and energy, plus derivatives of the pressure, are used to solve simultaneously for the stress and internal energy. The process is described algebraically below: it is implemented in HAFSTEP, the subroutine that computes density and energy and calls HSTRESS for the stress calculation.

The total mechanical stress \( R \) and the internal energy \( E \) are the variables to be determined. The stress \( R \) is defined as

\[
R = Q + \sigma = Q + \sigma^\prime + P
\]

where \( Q, \sigma, \sigma^\prime, \) and \( P \) are the artificial viscous stress, thermodynamic stress, deviatoric stress, and pressure. For the simultaneous solution for \( R \) and \( E \), \( R \) is presumed to be derivable from the previous value \( R_1 \) and the pressure derivatives as follows:

\[
R = R_1 + \frac{\partial P}{\partial \rho} \Delta \rho + \frac{\partial P}{\partial E} \Delta E
\]

(3.14)

Thus only changes in \( P \) are considered; changes in \( Q \) and \( \sigma^\prime \) are presumed to be small. The derivative \( \frac{\partial P}{\partial E} \) is derived analytically from the expression for pressure, while the other derivative is derived from the solution of Eq. (3.14) following the stress determination in the previous time step.

\[
\frac{\partial P}{\partial \rho} = \frac{R - R_1 - \frac{\partial P}{\partial E} \cdot \Delta E}{\Delta \rho}
\]

(3.15)

The two derivatives have approximately the following values:

\[
\frac{\partial P}{\partial \rho} = \Gamma E + \frac{C}{\rho_o}
\]

(3.16)

\[
\frac{\partial P}{\partial E} = \Gamma \rho
\]

(3.17)
where $\Gamma$ is the Grüneisen ratio, $C$ is the bulk modulus, and $\rho_o$ is the initial density. The estimate of internal energy $E$ is made by evaluating Eq. (3.2) with the available densities $\rho_1$ and $\rho_2$ at the previous and current times, the average of stresses $R$ and $R'_1$ (using Eq. 3.14), and the increment of radiant energy $\Delta E_r$

$$E' = E_1 + 0.5 \left( \frac{1}{\rho_1} - \frac{1}{\rho_2} \right) \left( 2R_1 + \frac{\partial p}{\partial \rho} \Delta \rho + \frac{\partial p}{\partial E} \Delta E \right) + \Delta E_r \quad (3.18)$$

(This is the actual expression used instead of Eq. (3.8).) With this value of internal energy, HSTRESS is called to compute the new stresses: $R'$, $\sigma_2$, and $P_2$. The simultaneous equations to be solved for the state variables $R_2$ and $E_2$ are derived from Eqs. (3.12) and (3.14).

$$E_2 = 0.5 \left( \frac{1}{\rho_1} - \frac{1}{\rho_2} \right) (R_1 + R_2) + E_1 + \Delta E_r \quad (3.19)$$

where $R'$ and $E'$ are the provisional values. The simultaneous solution of Eqs. (3.19) and (3.20) provides the required values of stress and energy. The thermodynamic stress quantities $\sigma$ and $P$ are not altered but are used as they are computed in HSTRESS.

3.3 Artificial Viscous Stress

The artificial viscous stress is required in finite difference wave propagation calculations to smooth out shock waves so that the entire flow field can be treated by the conservation equations of continuous flow, Eqs. (3.1) through (3.4). The artificial viscous stress $Q$ is the difference between the nonequilibrium mechanical stress $R$ and the equilibrium thermodynamic stress $\sigma$ given by the constitutive relations. Hence $Q$ represents real stresses occurring in the nonequilibrium states of the shock front. But the basis for computing $Q$ is artificial, depending on the computational cell size and on viscosity coefficients, which are not related to real physical processes.
In SRI PUFF the usual linear and quadratic viscosity forms are provided. The linear form is computed by the equation

\[ Q = -C_1 \rho C_s \Delta U \]  

(3.21)

where \( C_1 \) = dimensionless coefficient of linear artificial viscosity, 
\( C_s \) = sound speed, 
and \( \Delta U = U_{j+1} - U_j \).

The linear artificial viscosity is similar in form and operation to the standard linear viscosity models used to represent material behavior. However, here, the coefficient \( C_1 \) is chosen to provide enough damping to minimize oscillations in the calculations and not to represent the real material viscosity. In the code \( C_1 \) is given different values for compressive and rarefaction waves so that less damping can be provided for unloading processes. For compression, useful values are in the range of 0.05 to 0.30; for rarefaction, we have used 0.05.

The quadratic artificial viscosity proposed originally by von Neumann and Richtmyer has the form

\[ Q = C_o^2 \rho (\Delta U)^2 \]  

(3.22)

where \( C_o^2 \) is the dimensionless viscosity coefficient, and 
\( \Delta U = U_{j+1} - U_j \), as before.

The quadratic viscosity is permitted to act only on compressive waves. For normal values of \( C_o^2 \) of 3 or 4, the shock front is rapidly spread over three to four cells and then maintains essentially a constant thickness as the wave propagates. Because of the quadratic nature of the expression for \( Q \), very little damping occurs outside the shock front. By contrast, the linear viscosity tends to continue to erode the wave fronts as long as they propagate.

Normally, both linear and quadratic artificial viscosities are used, so the artificial viscous stress \( Q \) is the sum of the linear and quadratic terms from Eqs. (3.21 and 3.22). The quadratic viscosity quickly
establishes the shock front thickness. The linear viscosity damps the small oscillations that would otherwise occur near the shock front, but is given a small enough coefficient so that the wave front is not seriously eroded.

3.4 Time-Step Control

For the calculations to proceed in a stable manner, the time increment between cycles must be kept smaller than that given by the Courant-Friedrichs-Lewy condition (see Ref. 2, p. 262). This criterion is simply

\[ \Delta t \leq \frac{\Delta x}{C_e} \]  

(3.23)

where \( \Delta x \) is the cell size and \( C_e \) is the local effective sound speed (defined later).

The criterion means that the time step cannot be so large that the new points are outside the characteristic domain of dependence of the previous points. Referring to Figure 3.1, the new point \((n+1, j-1)\), for which the variables are computed from values at \((n, j-2), (n,j-1), \) and \((n,j)\), must lie within the domain of dependence or range of waves from those points. This domain is contained between lines with speeds of \( C_e \). A physical interpretation of the requirement is that a wavelet cannot be allowed to proceed from one coordinate point to beyond another in one time step, since this would allow a material point to "see," and be affected by, conditions at material points outside the true domain of dependence. This simple criterion is modified to provide for added safety (the time step used is 80% of the time step at the limit of stability), to allow for the effect of artificial viscosity, and to allow for the influence of high particle velocities.

Artificial viscosity stiffens the material and therefore increases the apparent sound speed, reducing the allowable time step. For linear and quadratic viscosity coefficients \((C_1 \text{ and } C_2)\), Herrmann et al. (Ref. 16, p. 37) derived the following reduction factor \( F \) to be applied to the time step:
\[
F = \frac{1}{\sqrt{1 + (C_1 + C_2 \cdot |\Delta U|/C_s)^2 + C_1 + C_2 \cdot |\Delta U|/C_s}}
\]  

(3.24)

where \( C_s \) is the material sound speed and \( \Delta U \) is the change in particle velocity between mesh points. To speed the computation by eliminating the square root process, the denominator of Eq. (3.24) is approximated by

\[
\sqrt{1 + C_F^2} \approx 1 + 0.5 C_F^2 + C_F
\]  

(3.25)

where \( C_F = C_1 + C_2 \cdot |\Delta U|/C_s \) because \( C_F \) should be a small fraction.

Our experience with radiation deposition computations has indicated that instabilities can arise when the particle velocities get very large. For example, in the vaporized region near the front surface, particle velocities may approach or exceed sound velocities. In such cases the usual stability criterion, \( \Delta t = \Delta X/C_s \), is no longer sufficient.

Consider the \( X, t \) plot in Figure 3.2. The point \( X_N \) is the intersection of a forward-going sound wave from \( (X_1, t_o) \) and the cell boundary, which was at \( (X_2, t_o) \). Then

\[
X_N = U_2 \Delta t + X_2
\]  

(3.26)
The time required for a wave to travel from \( X \) to \( X_N \), that is, to traverse the cell, is

\[
\Delta t = \int_{X_1}^{X_N} \frac{dX}{U + C_s}
\]  

(3.27)

It will be assumed that \( U + C_s \) varies linearly from \( X_1 \) to \( X_2 \) so that

\[
U + C_s = U_1 + C_{s1} + \tau(U_2 + C_{s2} - U_1 - C_{s1})
\]  

(3.28)

where \( \tau \) goes from 0 to 1. Then \( dX = d\tau(X_N - X_1) \) and the integral is

\[
\Delta t = (X_N - X_1) \int_0^1 \frac{d\tau}{U + C_s}
\]

\[
= \frac{X_N - X_1}{C_{s2} + U_2 - C_{s1} - U_1} \ln \left( \frac{C_{s2} + U_2}{C_{s1} + U_1} \right)
\]  

(3.29)
\[
\frac{X_N - X_1}{C_{s2} + U_2 - C_{s1} - U_1} \ln \frac{1 + y}{1 - y}
\]

where
\[
y = \frac{C_{s2} + U_2 - C_{s1} - U_1}{C_{s2} + U_2 + C_{s1} + U_1}
\]

The series expansion of the logarithm term is \(2(y + y^3 + \ldots)\). Only the first term is used here, giving
\[
\Delta t = \frac{2(X_N - X_1)}{C_{s2} + U_2 + C_{s1} + U_1} = \frac{2(X_2 + U_2 \Delta t - X_1)}{C_{s2} + U_2 + C_{s1} + U_1}
\]

When the \(\Delta t\) terms are collected on the left side, the result is
\[
\Delta t = \frac{2(X_2 - X_1)}{C_{s2} + C_{s1} + U_1 - U_2}
\]

If the value of \(\Delta t\) computed from this equation is negative, the two paths do not intersect and \(\Delta t\) can be set to an arbitrarily large value. The criterion used in the program is a simple combination of this equation and the safety factors, (0.8 and \(F\)), presented earlier:
\[
\Delta t = 0.8 \left( \frac{2(X_2 - X_1) F}{C_{s2} + C_{s1} + U_1 - U_2} \right)
\]

The time-step computations are begun in HSTRESS, continued in HYDRO, and completed in the main program.

Note that an effective sound speed accounting for artificial viscosity and particle velocity is
\[
C_e = \frac{C_{s2} + C_{s1} + U_1 - U_2}{2F}
\]

\(23\)
The sound speed $C_s$ is required only to control the time step. The analytical expression for sound speed is

$$C_s^2 = \frac{\partial \sigma}{\partial \rho} \frac{1}{S} = \frac{\partial P}{\partial \rho} \frac{1}{S} + \frac{\partial \sigma'}{\partial \rho} \frac{1}{S}$$  \hspace{1cm} (3.34)$$

where $\sigma, \sigma'$ are the stress and deviator stress in the direction of propagation and $S = $ entropy; as a subscript it means that the derivative is taken at constant entropy. The elastic or low stress approximation to the sound speed of compressional waves is

$$C_s^2 = \frac{C}{\rho} + \frac{4}{3} \frac{G}{\rho}$$  \hspace{1cm} (3.35)$$

where $C$ is the bulk modulus and $G$ is the shear modulus.

In the PUFF code the sound speed is used only to determine the permissible size of the next time step and to compute the artificial viscosity. The minimum time is governed by maximum speed, the speed of a small elastic unloading wave; hence, expressions (3.34) or (3.35) can be evaluated to give an upper bound on the sound speed. Thus $\partial \sigma'/\partial \rho$ or $G/\rho$ is computed from the largest shear modulus associated with the current stress, thereby neglecting that the material may be at yield so the effective modulus is actually zero.

At high stress, the bulk modulus is expected to increase significantly, so the derivative $\partial P/\partial \rho$ should be evaluated instead of using $C/\rho$. A procedure for numerically evaluating the partial derivative was developed for the program. The first law of thermodynamics for an isentropic ($dS = 0$) process is

$$dE = -PdV = -Pd\left(\frac{1}{\rho}\right)$$  \hspace{1cm} (3.36)$$

The usual rule for partial differentiation provides
\[ dE = \left( \frac{\partial E}{\partial P} \right)_\rho dP + \left( \frac{\partial E}{\partial \rho} \right)_P d\rho \]  

(3.37)

From these two equations and the chain rule

\[ - \left( \frac{\partial E}{\partial \rho} \right)_P = \left( \frac{\partial E}{\partial P} \right)_\rho \left( \frac{\partial P}{\partial \rho} \right)_E \]  

(3.38)

the required derivative is obtained:

\[ \frac{dP}{d\rho} = \left( \frac{\partial P}{\partial \rho} \right)_E + \frac{P}{\rho^2} \left( \frac{\partial P}{\partial E} \right)_\rho \]  

(3.39)

The derivative \( dP/d\rho \) was taken along an isentrope and therefore is properly written \( (\partial P/\partial \rho)_S \).

As an example of the sound speed calculation, the derivative is obtained for the Mie-Grüneisen equation with \( \Gamma \rho \) a constant.

\[ P = P_H (1 - \frac{\Gamma \mu}{2}) + \Gamma \rho E \]  

(3.40)

where

\[ P_H = C\mu + D\mu^2 + S\mu^3, \]  

the pressure on the Hugoniot

\[ C, D, S = \text{material constants with units of bulk moduli} \]

\[ \Gamma, \Gamma_0 = \text{the current and initial values of Grüneisen's ratio} \]

\[ \mu = \rho/\rho_0 - 1, \text{ a strain.} \]

Then the expression for sound speed, derived from Eq. (3.39) is
\[ c_s^2 = \left( \frac{\partial p}{\partial \rho} \right)_S + \frac{4G}{3\rho} = \frac{C+2D\mu + 3S\mu^2}{\rho_o^2} \left[ 1 - \frac{\Gamma_o}{2} \left( 1 - \frac{\rho}{\rho_o} \right) \right] \]

(3.41)

\[ + p_h \left( -\frac{\Gamma_o \rho_o}{2\rho^2} \right) + \frac{p}{\rho^2} \Gamma_o \rho + \frac{4G}{3\rho} \]

3.5 Outline of Subroutines

The subroutines that control the wave propagation calculations and contain the equations developed in this section are HYDRO, HAFSTEP, and HSTRESS. HYDRO contains the position and particle velocity calculations, whereas HAFSTEP has the density and energy calculations as well as the simultaneous pressure-energy solution. HSTRESS contains the artificial viscous stress (Q) and mechanical stress (R) equations, but is mainly a switching routine for selecting appropriate constitutive relations for each material. HYDRO and HAFSTEP are described below. Because of the involvement with constitutive relations, HSTRESS is described in Section 4.

HYDRO. For each call to HYDRO from SRI PUFF, a calculation is made for all cells and coordinates which are currently active. HYDRO contains separate paths for the several coordinate conditions provided. The coordinate conditions and their indicators are:

Normal (N) - interior coordinate point (within a layer of material).

Interface (L,R) - left and right coordinate points at an interface between layers.

Separated interface (S) - right coordinate point at a separated interface. First and last coordinates are treated by this path.

Mirror or reflective boundary (M) - a constant-velocity boundary (arbitrary velocity histories should be imposed by modifying this path).

Pressure boundary (P) - first and last boundaries may have a pressure history with a shock front and exponential decay, or a history provided by a series of pressure and time values.
Infinite boundary (I) - first or last boundaries are treated as if a mass of the same material continued indefinitely to the left or right past the actual first or last coordinate points (implemented only for planar case).

The path to be taken for each coordinate is determined by an indicator array, \( H(J,2) \). Values of the indicator are given above in parentheses following the path title.

In each path a call is first made to HAFSTEP to compute density, energy, and stress; then the new coordinate's position and velocity are computed. A test is made for spallation at the end of the interface path and for recombination in the separated path.

At the end of HYDRO, brief calculations are made to determine the largest \( J \) value (\( J^{\text{STAR}} \)) for which EDITs should be printed and to determine the stable time step for the next cycle.

**HAFSTEP.** The HAFSTEP subroutine is called by HYDRO for each cell and each time step to compute the midcell quantities of density, energy, and stress. To preserve accuracy in the stress calculations, the time step may be divided into small intervals (subcycles) for calculating the midcell quantities. Not more than 1% density change is permitted in any subcycle. This subcycling feature is important for constitutive relations in which internal energy is important and for relations based on differentials.

The internal energy is estimated using Eq. (3.18) and then HSTRESS is called for the stress calculation. Following the completion of HSTRESS, the final solution is made for energy and mechanical stress \( \sigma \) from Eqs. (3.19) and (3.20). The derivatives \( \partial P/\partial E \) used to determine the energy estimate are computed before returning to HYDRO.
4. CONSTITUTIVE RELATIONS

The constitutive relations provide the stress as a function of density, strains, internal energy, and other quantities. This section describes the common constitutive relations and outlines the available constitutive models. The subroutine HSTRESS, which selects the correct constitutive subroutine for each material, is also described.

4.1 Standard Constitutive Models

In the standard constitutive relations, the stress tensor is separated into a pressure and a stress deviator tensor. The pressure is the average stress

\[ P = \frac{1}{3} \Sigma \sigma_{ii} \]  

(4.1)

and the stress deviator elements are

\[ \sigma'_{ij} = \sigma_{ij} - P \delta_{ij} \]  

(4.2)

where \( \sigma_{ij} \) are stress tensor elements and \( \delta_{ij} \) is the Kronecker delta. The pressure is usually presented as a function of density and internal energy. The deviator stress is calculated by elastic, plastic relations, which may include thermal softening, rate-dependent effects, and work hardening. The standard pressure and deviator models are presented in the following sections.

4.1.1 Standard Pressure Models

The pressure is computed from a simplified form of an equation of state, the locus of all possible thermodynamic equilibrium states for a substance. Each state is a set of values of the following thermodynamic quantities: stress tensor, specific volume, entropy, specific internal...
energy, and temperature. In the simplified equation of state used here and in most wave propagation codes, the only variables considered are pressure (the deviator components of stress are treated separately), specific volume \( V \) or density \( \rho = 1/V \), and internal energy \( E \). The equation of state is then

\[ P = P(E, V) \]  

(4.3)

which defines a surface or locus of points in energy-pressure-volume space.

An equation of state represents equilibrium states. Therefore, as a material undergoes gradual changes, such as heating or compression, the successive states describe a path on the equation-of-state surface. If the material is compressed by passing through a steady-state shock front, the initial and final states lie on the P-V-E surface. These initial and final states are connected by a straight line, the Rayleigh line, which does not lie on the surface, but above the P-V-E surface. The states of transition within a shock front are not states of thermodynamic equilibrium. The equation of state describes the material behavior in solid, liquid, and gaseous phases. The standard pressure model gives a detailed treatment of the solid behavior, but the other phases are described by approximate relations without specific determination of the particular phase.

First, we examine the paths taken on the equation-of-state surface by material under shock loading. Shock experiments lead to the determination of a Hugoniot or Rankine-Hugoniot equation of state that is represented by one curve on the equation-of-state surface. This line is the locus of final states that can be obtained by a steady-state shock transition from a given initial state. The pressure-volume path taken by the material during the shock and a subsequent unloading is shown in Figure 4.1. The shock path follows a Rayleigh line to a point on the equation-of-state surface. Pressures on the Rayleigh line can be considered to be decomposed into an equilibrium pressure represented
FIGURE 4.1 PRESSURE PATHS FOR SHOCK LOADING AND UNLOADING OF A MATERIAL

FIGURE 4.2 ENERGY-PRESSURE-VOLUME (E-P-V) SURFACE FOR A SOLID MATERIAL
by a point on the equation-of-state surface plus a nonequilibrium
pressure component. In code calculations the equilibrium pressure is
computed from the equation of state; and the nonequilibrium component
is computed as the artificial viscous stress. Figure 4.2 shows the
Rayleigh line and unloading isentrope on the equation-of-state surface
with a Hugoniot curve. During the shock loading the internal energy
increases, as indicated in this figure. Less internal energy is used in
the elastic recovery on unloading down the isentrope; hence the unloading
does not coincide with loading, and the final, unloaded state is warmer
than the initial state and at a larger specific volume (for materials
that expand during heating).

As a reminder of the role of stress in the compression of the solid,
consider the stress-volume Hugoniot of Figure 4.3. Here only the stress
component in the direction of propagation is shown. During compression,
the stress is greater than the pressure; on unloading, the stress
decreases rapidly to yielding and then follows a stress isentrope below
the pressure isentrope.

Several other lines of interest are shown in Figure 4.2. The adia-
batic compression path is followed by a rapid but nonshock loading in which
no heat conduction occurs. The unloading isentrope is a similar, equilib-
rium process without heat conduction. The zero pressure line is the
locus of points obtained by simply heating the material without external
mechanical confinement. Heating increases the internal energy, and
thermal expansion occurs. For small increases in internal energy, the
zero pressure curve describes the usual expression for volumetric thermal
expansion

\[ V = V_o (1 + \alpha \Delta \theta) \]  \hspace{2cm} (4.4)

where \( V_o \) = the initial specific volume
\( \alpha \) = the volumetric thermal expansion coefficient
\( \Delta \theta \) = the change in temperature.
FIGURE 4.3 LOADING AND UNLOADING PATHS FOR PRESSURE AND FOR STRESS IN THE DIRECTION OF PROPAGATION
The zero pressure curve becomes asymptotic to the line described by

\[
\begin{align*}
E &= \text{vaporization energy} \\
p &= 0
\end{align*}
\]

for large \( V \).

The spall path is shown only to indicate the direction taken in tension. Spall, or fracture, is a rate-dependent process that generally depends on the stress tensor (not simply the pressure) and on the internal energy. Regions where the energy is high enough that the material is liquid or vapor are to the right in Figure 4.2. The vapor region extends indefinitely to the right.

The equation-of-state surface depicted in Figure 4.2 is an idealized form that is applicable to a material that does not experience solid phase changes or other phenomena that lead to regions of negative curvature in the \( P-V \) plane. While this surface represents the material behavior qualitatively, only certain regions of the surface are well understood quantitatively. The best-understood region is in the vicinity of the Hugoniot because of the availability of experimental data along that curve. The least-understood regions are those near spalling and those at high energies and to the right of \( V = V_o \).

Having outlined some properties of the equation of state, we now introduce the analytical forms used in the standard pressure model. In the model two expressions are used: one for compression to states with density greater than the initial density and one for extended states.

The equation used to describe compression is the Mie-Grüneisen equation

\[
P - P_{REF} = \frac{\Gamma(V)}{V} (E - E_{REF})
\]  \hspace{1cm} (4.5)

where
\( \text{P}_\text{REF} \) and \( \text{E}_\text{REF} \) = a point on some reference curve at the same specific volume \( V \)

\( \Gamma(V) \) = the Grüneisen ratio.

Equation (4.5) was derived by assuming that \( \Gamma \) is a function of \( V \) only. Equation (4.5) provides a means for extending the information of a known P-V relation (such as the Hugoniot) to other values of internal energy. Because the Hugoniot is the P-V relation that is most likely to be known, the computations are constructed so that the Hugoniot is the reference curve used. The Hugoniot P-V equation is presumed to be in the form

\[
\text{P}_\text{H} = C\mu + D\mu^2 + S\mu^3
\]  

(4.6)

where

\[
\mu = \frac{\rho}{\rho_0} - 1 = \frac{V_o}{V} - 1
\]

\( C = \) bulk modulus

\( D, S = \) coefficients with the units of moduli.

The internal energy along the Hugoniot is

\[
\text{E}_\text{H} = \frac{3}{2}\beta\text{P}_\text{H}(V_0 - V_\text{H})
\]

(4.7)

Equation (4.7) assumes that the initial internal energy is zero and that the Hugoniot is concave upward throughout. In general, the latter assumption excludes consideration of changes of state. Although the relation is strictly true only for the stress Hugoniot, not the pressure Hugoniot, little inaccuracy is introduced by this approximation. With the aid of Eqs. (4.6) and (4.7), the Mie-Grüneisen equation takes the following form in the program

\[
\text{P} = (C\mu + D\mu^2 + S\mu^3) \left(1 - \frac{\Gamma\mu}{2}\right) + \Gamma\rho\text{E}
\]

(4.8)
When material is held at a particular volume and heated (internal energy is added), it goes through states that are straight lines on the equation-of-state surface. This indicates that, for constant volume $V_1$, the analytical equations for the surface have the form

$$E = A(V_1) \cdot P$$  \hspace{1cm} (4.9)

where $A(V_1)$ = a function of $V_1$ only. The equation-of-state surface is constructed simply by translating the Hugoniot curve parallel to itself to higher energy states. The line $V = V_o$ is the boundary between the Mie-Grüneisen equation and an expansion equation.

The expansion equation, which is similar to that used in PUFF 66, must meet four requirements. It must:

- Join smoothly to the Mie-Grüneisen equation along $V = V_o$.
- Expand like $PV = E(\gamma - 1)$ at large expansions (like a perfect gas).
- Provide a linear relation between $P$ and $E$ for constant $V$.
- Account for the partition of internal energy into components for kinetic energy and for vaporization energy.

The equation that satisfies these requirements is

$$P = \rho \Gamma_e \left( E - E_e \left( 1 - \exp \left[ N(1 - \frac{\rho_{so}}{\rho}) \frac{\rho_{so}}{\rho} \right] \right) \right)$$ \hspace{1cm} (4.10)

where

$$\rho, \rho_{so} = \text{current and initial density}$$

$$\Gamma_e = H + (\Gamma_o - H) \left( \frac{\rho}{\rho_{so}} \right), \text{the effective Grüneisen ratio for expanded states.}$$

$$H = \gamma - 1 \text{ for expansion at low densities and } \gamma \text{ is the polytropic gas exponent.}$$

$$E_e = E_s \text{ in general}$$

$$= E_s \left[ 1 + \ln \left( \frac{E}{E_s} \right) \right] \text{ for } E > E_s \text{ and } n \neq 0.5$$

$$E_s = \text{sublimation energy for metals}$$

$$= \text{incipient vaporization energy for mixed-oxide ceramics.}$$

36
\[ n = \text{a constant, usually 0.5 for metals, 1.67 for mixed-oxide ceramics} \]

\[ N = \frac{C}{\Gamma_0} \] for \( \Gamma_0 = 0 \)

\[ = \frac{C}{\Gamma_0} + \frac{\text{Min}(E, E_s)}{E_e} \left( \frac{\Gamma_1}{\Gamma_0} + n \left( \frac{H}{\Gamma_0} - 1 \right) \right) \] for \( \Gamma_1 \neq 0 \)

\[ \Gamma = \Gamma_0 + \Gamma_1 \rho, \text{ the effective Grüneisen ratio for } \rho \geq \rho_{so} \]

\[ C = \text{coefficient in Eq. (4.6), the bulk modulus at low pressures.} \]

The present expansion equation differs from that in PUFF 66 because of improvements in \( N \) to provide continuity of \( \partial P/\partial V \) at \( \rho = \rho_{so} \) with the Mie-Grüneisen relation and to provide a variable vaporization energy, which seems to be required for some materials.

As an indication of the shape of the P-V-E surface generated by the expansion equation, several pressure-volume curves are given in Figure 4.4 for aluminum. Note that the curves are all continuous at \( \rho = 2.7 \), the density at which the joint to the Mie-Grüneisen equation occurs. The expansion equation permits a large tensile pressure excursion at low internal energies and then, for decreasing densities, gradually takes on the form of a perfect gas law. Figure 4.5 exhibits the modified PUFF expansion equation (typical of a mixed-oxide ceramic) in P-V-E space for compressive states. The initial solid (SO), solid melt (SM), and liquid boil (LB) points are labeled.

Many of the equation-of-state parameters are available in standard handbooks. For example, \( C \) is the isentropic bulk modulus at low pressures. According to Rice, McQueen, and Walsh,\(^{37} \) \( D \) in Eq. (4.6) may be estimated from \( D = \frac{\Gamma_0}{C} \). The sublimation energy, \( E_s \), is the difference between the internal energy of the solid material at ambient conditions and the internal energy of the fully expanded vapor at a temperature of absolute zero. This quantity is referred to as \( \Delta H_0^{0} \) in the JANAF tables\(^{38} \) for the gas state.

The Grüneisen ratio \( \Gamma \) may be estimated from thermal expansion data, using the relation...
FIGURE 4.4 PRESSURE-VOLUME RELATIONS AT CONSTANT INTERNAL ENERGY FOR AN ALUMINUM
FIGURE 4.5  SKETCH OF MIE–GRÜNEISEN AND MODIFIED PUFF–EXPANSION EQUATION–OF–STATE MODEL
\[ \Gamma = \frac{C\alpha}{\rho_{so} C_p} \]  \hspace{1cm} (4.11)

where

\[ \alpha = \text{the volumetric thermal expansion coefficient} \]
\[ C_p = \text{the specific heat at constant pressure}. \]

The result from Eq. (4.11) should be relied on only if all quantities pertain to the same density, pressure, and temperature. For many materials, \( \Gamma \) lies between 1.0 and 2.0; if internal energy is not important in the problem, an estimate can be made in this range.

The Hugoniot form traditionally used with PUFF calculations is the three-term expansion in Eq. (4.6). At large strains this form has the disadvantage that it does not have a physically reasonable behavior, especially if some of the coefficients are negative. Two alternative Hugoniot forms are discussed here: the Murnaghan form and the linear \( U_s - U_p \) relation. Both are provided as options in the standard pressure model.

The Murnaghan equation for the Hugoniot results from an integration of the following linear expression for bulk modulus.

\[ -V \left( \frac{\partial P}{\partial V} \right)_H = a + bP \]  \hspace{1cm} (4.12)

where the derivative is taken along the Hugoniot, \( V \) is the specific volume, \( P \) is pressure, and \( a \) and \( b \) are constants. On integration of Eq. (4.12), the Hugoniot pressure is obtained in the Murnaghan form

\[ P_H = \frac{a}{b} \left[ \left( \frac{V}{V_0} \right)^b - 1 \right] \]  \hspace{1cm} (4.13)

This form has the distinct advantage over (4.6) in always increasing monotonically. Hence, if it is used for pressures somewhat above the data on which the fitting parameters \( a \) and \( b \) are based, the computed pressures should be physically reasonable. The data from many materials
have been shown to fit this Murnaghan form well. The parameters $a$ and $b$ can be easily related to the coefficients in Eq. (4.6) by taking the derivatives of (4.6) with respect to volume and comparing terms. From Eq. (4.6)

$$-V \left( \frac{\partial P}{\partial V} \right) = C + \mu(C + 2D) + \mu^2(3S + 2D) + \ldots \quad (4.14)$$

Eq. (4.12) can be expanded to

$$-V \left( \frac{\partial P}{\partial V} \right)_H = a + b(C\mu + D\mu^2 + S\mu^3) \quad (4.15)$$

Therefore

$$a = C \quad (4.16)$$

$$b = 1 + 2D/C$$

Another estimate of $b$ is obtained from the Rice, McQueen, and Walsh relation $\Gamma = D/C$.

Then

$$b = 1 + 2\Gamma \quad (4.17)$$

For many solids the value of $b$ is approximately 5.

Shock wave data are often presented in the form of a linear relation between shock velocity ($U_s$) and particle velocity ($U$). The basic relation is

$$U_s = C_L + S_L U \quad (4.18)$$

where $C_L$ and $S_L$ are parameters determined by the fit to data. For a material with no deviator stresses, the pressure from Eq. (4.18) is

$$P_H = \rho_o U U_s = \rho_o (C_L U + S_L U^2) \quad (4.19)$$
Next we replace the velocities by using the expression for the conservation of mass across a shock front

\[ \frac{\rho}{\rho_0} = \frac{U_s}{U_s - U} \]  \hspace{1cm} (4.20)

and a Lagrangian strain \( \varepsilon \)

\[ \varepsilon = 1 - \frac{\rho_0}{\rho} \]  \hspace{1cm} (4.21)

By combining the foregoing four equations, we determine the Hugoniot pressure as a function of strain

\[ p_H = \frac{\rho_0 C_L^2 \varepsilon}{(1 - S_L \varepsilon)^2} \]  \hspace{1cm} (4.22)

This is the form used in calculations. By an expansion of the term in Eq. (4.22) and comparison of coefficients with those in Eq. (4.6), it can be shown that

\[ S_L = \frac{1}{2} \left( \frac{D}{C} + 1 \right) \]  \hspace{1cm} (4.23)

\[ C_L = \sqrt{\frac{C}{\rho_0}} \]  \hspace{1cm} (4.24)

From Eq. (4.23) and the standard value of 2 for \( \Gamma = D/C \), it is expected that \( S_L \) is approximately 1.5. The value of \( C_L \) is simply the bulk sound speed at low pressures.

### 4.1.2 Standard Deviator Stress Model

The deviator stress is the part of the stress tensor that arises because of the resistance of the material to shearing deformation. In PUFF the standard model for deviator stresses accounts for elastic response, plastic flow, work hardening, and thermal softening. The yield strength that governs plastic flow can be either of the Mises or Coulomb types. Here the relations are developed in a general form applicable...
to planar, cylindrical, or spherical flow. More advanced deviator models are found in Appendix G. Simplified forms specifically applicable to planar, cylindrical, and spherical flow are in Appendix F.

Elastic Relations. The elastic relations between stress and strain are cast in the following form

$$
\sigma_{ij}^{\prime} = 2G \left( \varepsilon_{ij}^E - \frac{1}{3} \delta_{ij} \varepsilon_{kk}^E \right) \quad (4.25)
$$

$$
P = CE_{ii} \quad (4.26)
$$

Here, $\sigma_{ij}^{\prime}$ and $\varepsilon_{ij}^E$ are the deviatoric stress and elastic strain in the $ij$ direction, $G$ is the shear modulus, $\delta_{ij}$ is the Kronecker delta, $P$ is pressure, and $C$ is the bulk modulus. For the elastic case, $\varepsilon_{ij} = \varepsilon_{ij}^E$, all the strain is elastic. But Eqs. (4.25) and (4.26) are also applicable to the plastic case where the strain increments are separated into elastic and plastic components.

$$
de_{ij} = de_{ij}^E + de_{ij}^P \quad (4.27)
$$

where $de_{ij}$ is the total strain increment and $de_{ij}^P$ is the plastic strain increment. For convenience, the terms in the parentheses of Eq. (4.25) can be named a deviator strain defined as follows:

$$
\varepsilon_{ij}^{\prime} = \varepsilon_{ij}^E - \frac{1}{3} \delta_{ij} \varepsilon_{kk}^E \quad (4.28)
$$

Then Eq. (4.25) becomes

$$
\sigma_{ij}^{\prime} = 2G \varepsilon_{ij}^{\prime E} \quad (4.29)
$$

Plastic Relations. The Reuss plasticity relations or "incremental plasticity with an associated flow rule" are considered here first. Modifications to treat Coulomb friction are described later. Yield occurs when the effective stress reaches the yield strength. The effective stress is
\[
\bar{\sigma} = \sqrt{\frac{3}{2} (\sigma'_{ij} \sigma'_{ij})} \quad (4.30)
\]

where the repeated subscripts indicate summation. The yield criterion is

\[
\bar{\sigma} = Y \quad (4.31)
\]

where \(Y\) is the current yield strength. The Reuss flow rule indicates that the deviator stress in any direction is proportional to the plastic strain in that direction:

\[
de_{ij}^P = \sigma_{ij}' \, d\lambda \quad (4.32)
\]

where \(d\lambda\) is a proportionality constant. Now we define a scalar plastic strain quantity as follows:

\[
de^P = \sqrt{\frac{2}{3} \, d\varepsilon_{ij}^P \, d\varepsilon_{ij}^P} \quad (4.33)
\]

As before, the repeated subscripts indicate summation. Now we square Eq. (4.32) and make use of the definitions of \(\bar{\sigma}\) and \(d\varepsilon^P\). Then

\[
d\varepsilon^P = \frac{2}{3} \bar{\sigma} \, d\lambda \quad (4.34)
\]

Combining this definition with Eq. (4.32), we find that

\[
d\varepsilon_{ij}^P = \sigma_{ij}' \, \frac{3d\varepsilon^P}{2\bar{\sigma}} \quad (4.35)
\]

To obtain a solution for an increment of strain, we compute first the stress that would occur if the strain were entirely elastic, that is,

\[
\sigma'_{ij}^N = 2G \left( \varepsilon'_{ij}^E + \Delta \varepsilon_{ij}' \right) = 2G \left( \varepsilon'_{ij}^E + \Delta \varepsilon_{ij}^P \right) \quad (4.36)
\]
where
\[ \varepsilon_{ij}^{E} = \text{the elastic deviator up to the current strain step} \]
\[ \Delta \varepsilon_{ij}^{E} = \text{the total deviator strain increment} \]
\[ \varepsilon_{ij}^{E} = \text{the elastic deviator strain after the current increment} \]
\[ \Delta \varepsilon_{ij}^{P} = \text{the plastic strain increment}. \]

The second equality in Eq. (4.36) is obtained by using Eq. (4.27) to decompose \( \Delta \varepsilon^{E} \) and by adding \( \varepsilon_{ij}^{E} + \Delta \varepsilon_{ij}^{E} \) to obtain \( \varepsilon_{ij}^{E} \). Quantities \( \varepsilon_{ij}^{E} \) and \( \Delta \varepsilon_{ij}^{P} \) can both be replaced by stress quantities through the use of Eq. (4.29) and Eq. (4.35). Then,

\[ \sigma_{ij}^{N} = \sigma_{ij}^{E} (1 + 3Gd^{E}/\bar{\sigma}) \]  

(4.37)

If both sides of Eq. (4.37) are squared and a quantity \( \bar{\sigma}_{N} \) is introduced in analogy to the definition of \( \bar{\sigma} \), then we obtain

\[ \bar{\sigma}_{N} = \bar{\sigma}(1 + 3Gd_{P}/\bar{\sigma}) \]  

(4.38)

Here, \( \bar{\sigma} = Y \).

Combining Eqs. (4.37) and (4.38) yields a solution for \( \sigma_{ij}^{E} \)

\[ \sigma_{ij}^{E} = \frac{\sigma_{ij}^{N} \bar{\sigma}}{\sigma_{ij}^{N} \bar{\sigma} + \sigma_{ij}^{P}} \]  

(4.39)

Then, the elastic strain can be obtained from Eq. (4.39) and the effective plastic strain from Eq. (4.38)

\[ d\varepsilon_{P} = \frac{\sigma_{ij}^{N} \bar{\sigma}}{3G} \]  

(4.40)

and finally, each component of plastic strain is found from Eq. (4.32).

The preceding process is especially appropriate for perfect plasticity where \( Y \) is constant. The equations are appropriate for steps from one plastic state to another or from an elastic state to a plastic state.
When Coulomb friction is introduced, the preceding equations for Mises plasticity are modified slightly. The fundamental relation provides a shear yield stress \( \tau_c \), which is a function of a cohesion \( c \), normal stress \( \sigma_N \), and the angle of internal friction \( \phi \)

\[
\tau_c = c + \sigma_N \tan \phi
\]  

(4.41)

Following Terzaghi, this expression is transformed to

\[
\sigma_1 = 2c \sqrt{\frac{N_\phi}{\phi}} + \sigma_3 \frac{N_\phi}{\phi}
\]  

(4.42)

where \( N_\phi = \tan^2 (45^\circ + \phi/2) \); and \( \sigma_1 \) and \( \sigma_3 \) are the most and least compressive principal stresses. In the derivation we consider that yielding has no effect on volume change (a Coulomb-without-dilation model).

Instead of using Eq. (4.42), which is not symmetric because the intermediate principal stress is absent, we introduce the expression of Drucker and Prager

\[
\sqrt{J_2^*} = k + 3\alpha P
\]  

(4.43)

where \( J_2^* \) is the second invariant of the stress deviator tensor, and \( k \) and \( \alpha \) are constants. Replacing \( J_2^* \) by the effective stress \( \overline{\sigma} = \sqrt{3J_2^*} \), we can obtain the following form for Eq. (4.43)

\[
\overline{\sigma} = \frac{3c \sqrt{N_\phi} + \frac{3}{2}(N_\phi - 1)P}{1 + N_\phi/2}
\]  

(4.44)

The constants \( k \) and \( \alpha \) have been replaced by \( c \) and \( N_\phi \) by equating Eqs. (4.42) and (4.43) for the case \( \sigma_2 = \sigma_3 \). The individual deviator stresses are then obtained from Eq. (4.39).

**Work Hardening.** A linear work hardening is assumed in the following form:

\[
Y = Y_o + Y_D |\Delta \rho|
\]  

(4.45)
where \( Y_D \) is a work-hardening coefficient with the units of \( \text{dyn/cm}^2/(\text{g/cm}^3) \). This form is used mainly for historical reasons because it was present in PUFF 66. The input value of \( Y_D \) is discussed in Section 5.5. The present formulation is satisfactory for planar flow in which all strain is related to density changes. More appropriate work-hardening processes for other flows are discussed in Appendix G.

**Thermal Softening.** Material that is heated to an internal energy near melting generally loses considerable strength. In PUFF, thermal softening is permitted to reduce both the yield strength and the shear modulus of a material. Physically each of these parameters probably reduces as a different function of the temperature. Figure 4.6 shows stress-strain relations for two possible thermal softening relations. In each case, it is assumed that the material has been loaded through yielding to the point labeled \( Y_1 \) and then heated sufficiently to produce a decrease in yield and modulus. For the case where the thermal softening functions \( F_Y \) and \( F_G \) for both yield \( Y \) and modulus \( G \) are equal, complete elastic unloading from either point \( Y_1 \) or point \( Y_2 \) would reach the same value of shear strain; hence no change in plastic strain is involved. However, when \( F_Y \) is not equal to \( F_G \), some adjustment occurs in \( \varepsilon^P \), as shown. When \( F_G \) is greater than \( F_Y \), there is an apparent increase in \( \varepsilon^P \), although no strain has actually occurred in proceeding from point \( Y_1 \) to point \( Y_2 \). In the code calculations, different thermal reduction functions are permitted for \( Y \) and \( G \); however, no adjustment is made in \( \varepsilon^P \).

### 4.2 Constitutive Model Types and Switching Routine for Selecting Models

Constitutive or material models may take many forms besides the standard types presented above. Some of the available nonstandard models are introduced here, and the routine for calling them in the code is described. Procedures for inserting new models are described in Appendix H.

Our work in porous materials, fracture, composites, and explosives has led us to require the use of very general material models. PUFF models have been constructed to reflect these requirements. For example, a porous material may consolidate; therefore, calculations should be able to begin with the porous material model, but transfer to a solid model after consolidation. For fracture calculations it should be
EQUIVALENT PLASTIC STRAIN, $\varepsilon_p$

(a) EQUAL THERMAL-SOFTENING FUNCTIONS

\[ \varepsilon_p \]

\[ \varepsilon_p ^{(a)} = \varepsilon_p^{(1)} \]

\[ G_1 = G_1 F_C = G_1 F_Y \]

EQUIVALENT PLASTIC STRAIN, $\varepsilon_p$

(b) UNEQUAL THERMAL-SOFTENING FUNCTIONS

\[ \Delta \varepsilon_p \]

\[ \varepsilon_p ^{(b)} = \varepsilon_p^{(2)} \]

\[ G_2 = G_1 F_C = G_1 F_Y \]

FIGURE 4.6  EFFECTS OF YIELD AND MODULUS THERMAL-SOFTENING FUNCTIONS ON PURE SHEAR STRESS-STRAIN RELATIONS
possible to treat the material with a continuum model up to incipient fracture and then transfer to a fracture model. Furthermore, the material state should determine which type of fracture model to call. Composites should be simulated either by a single model or by a combination of models representing the constituents. If pressure and deviator stresses are treated separately for the material, then any pressure model should be combinable with any deviator model. These general requirements have been followed in setting up the model types.

At present, five model types are accounted for in PUFF.

- **Composite**, for multiconstituent materials. Total stresses are computed by the model.
- **Fracture**. A continuum model is called until fracture begins. The use of a fracture model is triggered by a criterion preceding the CALL statement. Total stresses are computed.
- **Porous**. Either total stress or pressure are computed, depending on the model. At consolidation, transfer may occur to a continuum model.
- **Deviator**. Only deviator stresses are computed, so one of these models is used in conjunction with a pressure model.
- **Pressure**. Only pressure is computed. Explosives are treated under this heading.

Occasionally still greater flexibility is required in modeling complex materials. For example, it may be necessary to use a particular deviator model first with a pressure model and then as part of a fracture model. Or it may be desirable to call a fracture model from a porous model. The capability of calling any model subroutine from any other routine is made possible by eliminating the COMMON variables from all models. All information enters each subroutine through its CALL statement. Hence special combinations of models can be obtained fairly readily with small changes in the program. Some guidance on making such changes is included in Appendix H.

The subroutine HSTRESS has been constructed to serve as a switch between the various subroutines computing pressure, deviator stress, and total stress. The flow chart in Figure 4.7 emphasizes these
FIGURE 4.7 FLOW CHART OF HSTRESS, STRESS-SWITCHING ROUTINE
FIGURE 4.7 FLOW CHART OF HSTRESS, STRESS-SWITCHING ROUTINE (Concluded)
stress-switching features. Material models that are currently available are listed in the figure and in Section 2. The list in Section 2 also shows where to find more information about each model.

4.3 Spall Calculations

A simple spall model is provided to permit material separation when the stress exceeds a critical level, \( T \). The spall criterion is checked and separation calculations are made in HSTRESS following the normal stress calculations.

The spall criterion is based on \( R = \sigma_1 + Q \) in the direction of propagation and on \( \sigma_2 \) and \( \sigma_3 \) in the other two directions. The stresses in the three directions are

\[
\begin{align*}
(1) \quad & R = \sigma_1 + Q = \sigma_1^\prime + P + Q = SDH + PHL + Q \\
(2) \quad & \sigma_2 = \sigma_2^\prime + P = -\frac{1}{2}\sigma_1 + P \text{ for planar and spherical} \\
& \quad = P + \sigma_2' = PHL + SDT \text{ for cylindrical} \\
(3) \quad & \sigma_3 = \sigma_2 \text{ for planar and spherical} \\
& \quad = P - \sigma_1^\prime - \sigma_2^\prime = PHL - SDH - SDT \text{ for cylindrical}
\end{align*}
\]

The first step in the spall calculations is to compare the stresses in all three directions with the spall criterion \( T \).

If spall has occurred in any direction, the stress in that direction is zeroed and elastic rebound (recompression) occurs in the other two directions. The final stress configuration is obtained by applying a compressive stress \( \Delta \sigma_1^f \) equal in magnitude to the tensile stress \( \Delta \sigma_1^f \) in the spall direction. Because the stress \( \Delta \sigma_1^f \) is applied while allowing only strain (opening) in the spall direction, the pressure and deviator components are computed from the usual elastic relations for planar flow.

\[
\Delta P_1^f = \frac{C}{C + \frac{4}{3}G} \Delta \sigma_1^f \quad (4.46)
\]
\[ \Delta \sigma_1^f = \frac{4}{3} \frac{G}{C + \frac{4}{3} G} \Delta \sigma_1^f \quad (4.47) \]

The change in deviator stress in the other two directions is \(-\Delta \sigma_1^f/2\). From these relations, the final stresses in the three principal directions are computed.

\[
\begin{align*}
\sigma_1 & \rightarrow 0 \\
\sigma_j & \rightarrow \sigma_j - \Delta P^f + \frac{1}{2} \Delta \sigma_1^j \\
\sigma_k & \rightarrow \sigma_k - \Delta P^f + \frac{1}{2} \Delta \sigma_1^k
\end{align*}
\]

Similarly, the pressure becomes \(P - \Delta P^f\) and the deviators are modified as follows

\[
\begin{align*}
\sigma_1^* & \rightarrow \sigma_1^* - \Delta \sigma_1^f \\
\sigma_j^* & \rightarrow \sigma_j^* + \frac{1}{2} \Delta \sigma_1^f \\
\sigma_k^* & \rightarrow \sigma_k^* + \frac{1}{2} \Delta \sigma_1^f
\end{align*}
\]

The spall model now in PUFF correctly treats spallation and continued separation. Since separation strain is not stored, reconsolidation is determined only by a return to a consolidated density. Spall is permitted in only one direction at a time.
5. INITIALIZATION: THE GENRAT GROUP

The GENRAT subroutine is called once at the beginning of each problem to read in all the data and initialize the COMMON storage. The GENRAT group includes DEPOS, EXTRA, HDATA, PRESCR, REDR, SCATTO. The sequence of major operations conducted by these subroutines is:

- Read general running instructions for the problem
- Read properties for each material
- Lay out a coordinate grid over all the materials
- Compute the absorption of radiated energy (for a radiation problem)
- Initialize the coordinate and cell arrays
- Print initial coordinate values.

This section describes the philosophy of the input, shows the derivation of equations, and contains guidance on the choice of input parameters.

The next four subsections describe the input deck used with PUFF. All the input information is organized to reflect the following guidelines:

- Each card or group of cards is labeled for ease of identification. For example, equation-of-state data begin with the identifier EQST; yield data begin with YIELD.
- Each input line is read and then printed immediately in the same format (echo printing) so that the first page of printout looks like the input deck.
- The first column of each card is treated as an indicator to control the reading process, but it is not data.
- The minimum amount of data is used for each problem. For example, the required data for a material are contained on just two cards. On the first card are indicators that show whether more property cards are required because of special models used for the material.

GENRAT has the capability of performing several problems one after the other and for reading material properties or spectral data from a data bank on disk, tape, or cards. The input deck structure required for using these capabilities is described in the following subsection.
and shown in sample decks in Appendix C. The initialization operations require the subroutines GENRAT, DEPOS, SCATTO, REDR, HDATA, and EXTRA. The following subsections describe four sets of data cards that may be used for each problem: general running data, materials data, cell layout, and radiation data. The first three sets are required for every problem, but the fourth is needed only for radiation problems.

5.1 Input of General Running Information

The first group of data identifies the computation and contains indicators controlling the length of the computation, the amount of printing, the number of materials and the type of computation.

The first or title card contains a brief title for the run. This line of information (plus the date) serves as a heading for each page of all major prints from the GENRAT, Scribe, and EDIT subroutines. The first character of this card serves as an indicator:

Blank - normal input continues.
D    - Deposition layout only; the next required input card is the NMTRLS card.
T    - The remainder of the general running data should be read from tape. On the tape these data records follow a title record containing the last 10 characters of the title card (See Appendix C).
X    - Same as "T", but in addition, data will be read in through the EXTRA routine following the NMTRLS card.

When the first character is blank, any number of comment cards may follow this first card if these cards contain a nonblank first column.

The second normal input (NTEDT) card contains some of the print controls (NTEDT and NJEDIT), the rezoning control (NREZON), and the geometry designator (NALPHA). NTEDT is the number of EDITs (print of condition of the coordinate array at a specified time) to be called, and NJEDIT is the number of lines containing coordinate locations (JEDITS) for which a stress history is to be printed. If NTEDT is nonzero, the next cards contain a list of the TEDITS or times at which the prints
will occur. If NJEDIT is nonzero, the following NJEDIT lines contain a list of indicators of the variables and J values of cells for which historical listings are needed: the format for these lines is described in Appendix C under Historical Prints. NREZON controls rezoning, i.e., resizing of cells and recomputation of associated coordinate and cell quantities at intervals during the computation. The type of rezoning depends on the sign of NREZON:

- A positive NREZON is the number of rezones desired, and two additional input cards containing lists of NTR and JREZON are required. NTR is the number of the TEDIT (hence the time at which each rezone is called); JREZON is the rightmost coordinate in each rezone.

- If NREZON is negative, an input card containing DTMAX, TREZON, NARZ, and TARZ is required. DTMAX is the desired size of the time step DTNH and TREZON is the time interval between rezones. Rezoning is terminated if the number of rezones exceeds NARZ or the time exceeds TARZ. If NARZ and TARZ are zero, then rezoning continues at intervals of TREZON until DTNH exceeds DTMAX. If DTMAX is negative on the input card, it is interpreted as the number of cells desired in the material whose layer number is L = - NREZON. From this input value, DTMAX (in its usual significance) is computed in GENRAT.

The geometry designation NALPHA has the meaning:

0 or 1  Planar grid
2  Cylindrical layout with X = 0 at axis of cylinder
3  Spherical layout with X = 0 at center of sphere.

The subroutine REZONE can only increase the size of cells; therefore, cells should be laid out as small as desired initially. In REZONE it is presumed that the cell at JREZON (an input quantity for NREZON > 0 or a cell selected by REZONE for NREZON < 0) is of proper size, then all cells with smaller J values are resized to about the same thickness.

In a radiation deposition computation, small cells are needed at early times near the front surface to properly model the deposition, expansion, and spallation that occur in that region. After the deposition is complete, there is less need for the very small cells near the front. A reasonable approach to handling these requirements is to lay out the coordinates initially with a geometric size variation starting
with $10^{-4}$ to $10^{-5}$ cm cells at the front and possible increasing to 0.1 cm at the rear. Following deposition, the cells may be increased in size by rezoning. Because of the averaging operations that occur in REZONE, there is a loss of kinetic energy and some smoothing of the stress wave; therefore, rezoning should not be used excessively and cell sizes should not be more than doubled at each rezone.

For impact problems, a different procedure should be followed for rezoning and initial layout. To properly represent the stress history in the impact of a thin flyer on a target, 10 to 20 cells should be used in the flyer and similar sized cells should be used in the target at the impact point. Larger cells can be used deeper into the target. The appropriate time for rezoning is following the completion of the impact (twice the propagation time through the flyer). Usually one rezoning is sufficient to establish a suitable cell size for the balance of the computation.

Following the NTEDT card and the cards containing TEDITs, JEDITs, and rezoning controls is a card containing NEDIT and three termination criteria. NEDIT is the number of cycles between calls to EDIT. These EDIT calls are independent of those provided by the TEDIT array; hence this is a second procedure for requesting an EDIT printout. The parameters that are used to stop the running of the problem are JCYCS, the number of major cycles or calls to the HYDRO subroutine that can be made before the program stops; CKS, the depth into the material beyond which the maximum stress should not move; and TS, the stop time. The calculation halts when any of these three is reached.

The last required data card in this group contains NMTRLS, the number of materials; MATFL, the number of the last layer of the flyer plate (neglect gaps in counting these layers); UZERO, the velocity of the flyer plate, and NSCRB, a set of 10 flags indicating whether plotting is called for from DEPOS. For problems other than impacts, MATFL acts as an indicator for the type of problem:

- Explosive detonation: set MATFL = 1, UZERO = 0. The problem is initiated by the energy insertion procedure in EXPLODE.
• Radiation deposition: set MATFL = 0. Then DEPOS is called to provide the energy deposition.
• Mirror impact: set MATFL = -1 for a symmetric impact.
• Pressure boundary at J = 1: set MATFL = -2 and provide a pressure history in FUNCTION SIGMAT (1, TIME) or read in P6(1) and T6(1) through the EXTRA routine following the normal data deck. The applied pressure has the form P = P6(1) \exp(TIME/T6(1)).
• Pressure boundary at J = JFIN: set MATFL = -3 and provide a pressure history in SIGMAT(2, TIME) or read in P6(2) and T6(2) as for pressure at J = 1.

The plotting called for by the flags NSCRB(1) to NSCRB(3) occurs at the end of the layout and is controlled by DEPOS. The three flags pertain to plots of energy, pressure, and temperature, respectively, as functions of distance into the target. If one or more of these flags are nonzero, then x and y ranges for each plot are read in.

5.2 Input of Material Properties

Following the general running information are several sets of cards, one set for each material. The material properties information is grouped in the following categories:

• Material name, solid density, and a set of flags—NCMP, NFR, NPOR, NDS, NPR, NYAM, and NCON—which control the reading of additional data, plus NVAR, which controls the number of extra variables per cell available for the material (See Appendix C for NVAR). In the input listings in Appendix C, the first 6 indicators are labelled with the contracted titles CFP and DPY.

• Solid equation-of-state parameters: EQSTC, EQSTD, EQSTE, EQSTG, EQSTH, and EQSTS. EQSTC, EQSTD, and EQSTS are the parameters of the Hugoniot pressure function. The C,D,S form Eq. (4.6), the linear shock velocity relation or Murnaghan equation can be represented; EQSTS indicates which form is used. The three parameters have the following meanings:

| Material | C, D, S form | Murnaghan | Linear $U_s - U$
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>EQSTC</td>
<td>C</td>
<td>a/b</td>
<td>$C_L$</td>
</tr>
<tr>
<td>EQSTD</td>
<td>D</td>
<td>b</td>
<td>$S_L$</td>
</tr>
<tr>
<td>EQSTS</td>
<td>S</td>
<td>1.0</td>
<td>2.0</td>
</tr>
</tbody>
</table>
The parameters EQSTG and EQSTH are the Grüneisen ratios \( \Gamma \) and \( \mathcal{H} \). EQSTE is the sublimation energy.

- Special data required for composite (NCMP), fracture (NFR), porous (NPOR), deviator stress (NDS), or pressure (NPR) models. Some of these are read in GENRAT, and some in the subroutine containing each model. See Nomenclature for meaning of each indicator.

- Optional material properties. TENS, spall strength values (Section 4.3); COSQ or VISC, artificial viscosity coefficients (Section 3.3); YIELD, yield strength and shear modulus (Section 4 and Appendix G); and EMELT, or MELT, GMELT, thermal strength reduction parameters (Appendix D). The number of these optional lines is NYAM.

- Radiation absorption data (NCON). NCON is the number of constituents of a material for which radiation absorption data are provided; hence mixtures, alloys, and composites are accounted for (Section 5.4 and Appendix A).

Of this imposing array, only the first and second lines are required. The flags that are read in on the first card indicate which, if any, of the other data items are supplied. The data under the control of NYAM are all given nominal values by GENRAT: these nominal values are used unless they are over-written by data from input. The spall strength within materials is initialized high to avoid spall, the spall strength between layers is low to permit separation, the yield strength is zero, quadratic and linear artificial viscosity coefficients are 4.0 and 0.15, respectively, and the thermal strength reduction function is set to degrade the strength gradually and permit melting at one-tenth the sublimation energy (EQSTE).

The material data in all the above categories may be provided either from a data bank or as part of the input deck. Details of the data deck setup for these two alternatives are given in Appendix C. If a data bank is used, it contains a series of card images corresponding exactly to those that would appear in a material properties deck. To indicate that a data bank is being used for the material data, one card is inserted containing a nonblank first column and the material name. Examples of such data decks are shown in Appendix C. The use of a data bank is especially convenient for multiple runs with an identical set of materials.
5.3 Layers and Cell Layout

The materials in the problem are laid out in a series of layers, and each layer is discretized into a number of finite difference cells. The total number of layers, including any empty layers or gaps, is given as NLAYER. The array JMAT then provides the relationship between layers and materials. For example, for layer L, JMAT(L) is M, the material number. For an empty layer, JMAT(L) is zero. No finite difference cells or coordinates are used to represent gaps; adjacent layers of material are merely separated by the gap distance. Following the cell layout, NLAYER is reduced by GENRAT to the number of layers containing material.

The materials in each layer are laid out in a Lagrangian grid (Lagrangian because the grid moves with the material), with variable spacing between the grid points. This variability allows for flexibility in planning the layout of the grid, for concentrating small cells near regions of interest, and for using large cells elsewhere. For best results in the computations, the cell sizes should be allowed to vary slowly. Each material is divided into one or more zones; within each zone the cell sizes are uniform or they vary in either an arithmetic or geometric progression. The numbering of cells and coordinates is shown in Figure 5.1. Each cell has the same number as the coordinate to the left. Energy (EHL), mass (ZHL), density (DHL), pressure (PHL), and stress (SHL) are the basic quantities associated with the cells or mid-cell points. The coordinate location (X) and velocity (U) refer to coordinate points. Figure 5.2 shows a possible variation of cell sizes (five different zones of varying sizes are possible for each layer; zones of geometric and arithmetic cell variations may be intermixed). The numbering system that is used for the grids is also shown in Figure 5.2; two coordinate numbers are assigned to interfaces between materials. The last coordinate point in each layer is called JBND(NL) where NL is the layer number. The last coordinate used is JFIN, which is one greater than the last JBND value; this definition of JFIN is useful for the operations in HYDRO.

The zoning input data are provided on a series of cards, one for each zone of each layer. The first card of the zoning set gives the
FIGURE 5.1 COORDINATE LAYOUT FOR SRI PUFF
Figure 5.2 Coordinate layout showing variable cell size capability
number of layers (NLAYER) and the array (JMAT), which tells which material is in each layer of the layout. On the first card for each layer is the number of zones, NZONES; the number of cells in the first zone, NCELLS; the zone thickness, TH; the size of the first cell, DELX; and the size of the last cell of the zone in an arithmetic progression, DELFIN. For a geometric progression, DELFIN is interpreted as the RATIO between sizes of successive cells, and DELX is disregarded. For uniform cell sizes, DELX and DELFIN are omitted. For an arithmetic layout, either DELX or DELFIN is specified. A geometric layout is indicated by values for both DELX and DELFIN, although the actual value of DELX is disregarded. Examples of the cards appear in Appendix C.

The following analyses of the zoning are made to give the bases for the computer calculation and also to present formulas for the incremental thickness change between cells and for thicknesses of the first and last cells. It is often desirable to compute the increment and cell sizes before a propagation calculation to guarantee proper matching between cell sizes in different zones and ensure that the change in cell size is not too great within a zone. For the analysis of an arithmetic progression, the thickness of a zone $T_h$ is first represented as a sum of the cell thickness $\Delta X_1$.

\[
T_h = 2\Delta X_1 = \Delta X_1 + (\Delta X_1 + \delta) + (\Delta X_1 + 2\delta) + \ldots + (\Delta X_1 + (N_c - 1)\delta) \tag{5.1}
\]

where $\Delta X_1 = \text{DELX}$ is the thickness of the first cell, $\delta$ is the incremental change in thickness from one cell to the next, and $N_c$ is the number of cells in the zone. Using the formula for the sum of an arithmetic series Eq. (5.1) is changed to the following form:

\[
T_h = N_c \Delta X_1 + \delta(N_c - 1)N_c / 2 \tag{5.2}
\]

Equation (5.2) can then be rearranged to obtain the equation for the incremental change in cell thickness, $\delta$

\[
\delta = 2 \left( \frac{T_h}{N_c} - \Delta X_1 \right) / (N_c - 1) \tag{5.3}
\]
According to Eq. (5.1), the expression for the thickness of the last cell, $\Delta x_f = \text{DELFIN}$, is the following

$$\Delta x_f = \Delta x_1 + (N_c - 1) \delta$$

(5.4)

Then Eq. (5.2) can be altered to give the form

$$T_h = N_c (\Delta x_1 + \Delta x_f)/2$$

(5.5)

Equation (5.5) can then be rearranged to provide expressions for evaluating the thickness of either the last cell in the zone, given the thickness of the first, or vice versa

$$\Delta x_f = 2T_h / N_c - \Delta x_1$$

(5.6)

$$\Delta x_1 = 2T_h / N_c - \Delta x_f$$

(5.7)

When an arithmetic progression zoning is desired, either $\Delta x_1$ or $\Delta x_f$ may be entered. For a uniform distribution of cell sizes, both $\Delta x_1$ and $\Delta x_f$ should be left at zero.

For the geometric progression the input quantity DELFIN is interpreted as $R_x$, the ratio between successive cell sizes. The first cell has the thickness $\text{DELX} = \Delta x_1$, and the last cell thickness is

$$\Delta x_f = \Delta x_1 R_x (N_c - 1)$$

(5.8)

The thickness of the zone is given by the usual sum of a geometric progression.

$$T_h = \Delta x_1 \frac{(1 - R_x^{N_c})}{1 - R_x}$$

(5.9)

The geometric cell layout is actually overspecified by the input. Therefore the input value of $\Delta x_1$ is disregarded and $\Delta x_1$ is computed from Eq. (5.9) and the given values of $T_h$, $R_x$, and $N_c$. The nonzero value of $\Delta x_1$ in the input merely indicates a geometric layout. The geometric progression is particularly useful in radiation deposition problems.
in which it may be necessary to vary cell thicknesses from $10^{-5}$ cm at
the surface to $10^{-1}$ cm deep inside the material.

Correct sizing of the cells can be very important in getting useful
results from a computation. No complete theory is available for optimiz-
ing cell sizes, but the following guide lines have been obtained:

- Small cells should be used at the surface of deposition in
  a radiation deposition problem. The cells should be small
  enough that no more than 1% of the energy is absorbed in
  the first cell. If vaporization occurs, several vaporizing
  cells should be provided.

- In an impact problem the cells on either side of the inter-
  face should be matched in such a way that the interface
  particle velocity computed on the first cycle is approxi-
  mately equal to the final steady-state value. This sizing
  can be accomplished adequately if the cells are sized so
  that the times to traverse them are about equal; i.e.,
  materials with low velocities should have smaller cells.
  The correct interface velocity need not be obtained on the
  first cycle, as the program will iterate to the correct
  value in a few cycles if the artificial viscosity coeffi-
  cients have normal values. Large amounts of viscosity will
  slow the convergence of the iterations. It does not appear
  necessary to match cell sizes precisely across an interface,
  even the impact interface. A series of computations was
  made with an impact of C-7 epoxy ($\rho = 1.19$) and tungsten
  ($\rho = 19.3$). The "equal time" criterion above dictated that
  C-7 cells should be $5/8$ as large as tungsten cells. Compu-
  tations were made with C-7 cells $1/4$, $5/8$, $1.0$, and $2.5$
  times as large as the tungsten cells. Even the most mis-
  matched cases gave an initial overstress only 8% higher
  than the best matched case.

- For porous materials that are compacted during the computa-
  tions, a large number of cells should be used to represent
  the material. This number is required to provide adequate
  definition of the material response during the compacting
  process. Generally, a half-consolidated cell is not a
  good average of an uncompacted cell and a solid cell.

- Rise times of stress waves are equal to several traverse
  times for the cells. Hence, the definition of the stress
  history can be used as a basis for defining acceptable
  cell sizes.

- Cell sizes can be varied gradually (less than 5% per cell)
  so that the cells are small and stress waves are sharply
  defined in regions of interest and large at other points
  in the flow. The material boundaries need be extended only
  far enough from the region of interest that no disturbing
  wave from the boundary reaches the region of interest
during the problem time.
5.4 Thermal Energy Deposition

Thermal energy is deposited into the cells to simulate radiation from x-ray, electron beam, or laser sources. The energy is deposited into the cells at a constant rate during the shine time of the source. This section outlines the deposition options and required input. Appendix A contains more information about the energy deposition process. Initialization of deposition is handled in the subroutine DEPOS.

In SRI PUFF, several radiation sources may be used at once, each with its own spectrum and shine time. The sources may radiate at normal incidence onto the material layers (planar geometry is assumed) or at oblique angles. Each layer may have a different angle to treat radiation through several separate layers at different inclinations.

5.4.1 Deposition Types

Three deposition procedures are available for representing radiation from each source.

- **Black body x-ray source.** The radiation source is represented as a series of black bodies. The required data are energy reaching the surface (cal/cm²), temperature of each black body (keV), and absorption coefficients for each material.

- **Arbitrary x-ray spectrum.** The radiation source is represented by a table of energies in calories/cm² versus hv (photon energy or temperature, in keV) for each spectrum. Absorption coefficients for each material are required.

- **A depth-dose profile in the form of a table of deposited energies (calories/g) versus depth (cm).** This option permits use of x-ray deposition profiles from a code that treats scattering, fluorescence, and photoelectric effect or deposition from laser or electron beam sources. No absorption data are required with this option.

With the black body option, DEPOS constructs a spectrum consisting of 95 energy values at specific hv (photon energy) points. Then the radiant energy that will be deposited in each finite difference cell is computed and stored in the SS array. The photoelectric absorption coefficients are used for the deposition calculation. Because absorption by Compton scattering and fluorescence is neglected, the DEPOS deposition calculation should not be relied on for black body temperatures greater than...
a few keV. DEPOS treats the arbitrary spectrum the same as the black body spectrum for deposition calculations.

For the third deposition option, the depth-dose profile is used to calculate the radiant energy to be deposited in each finite difference cell. The energy for each cell is calculated (in SCATTO) by passing an interpolation function through sets of points in the profile and integrating the area under the function between the limits of the cell dimension.

5.4.2 Data Required

Three types of data are read into DEPOS for deposition calculations: photoelectric absorption data, spectra or black body temperatures, and depth-dose profiles. The absorption coefficient for x rays has the form shown in Figure 5.3. In a log-log plot there are sharp discontinuities at hv values corresponding to the electron energy levels or edges. Between these edges the absorption function is usually fairly linear. The following function is used to fit the absorption data between edges:

$$\ln \sigma_a = A_0 + A_1 w + A_2 w^2 + A_3 w^3$$  \hspace{1cm} (5.10)

where  
\(\sigma_a\) = mass absorption coefficient (barns/atom)  
\(w = \ln(hv)\), with \(hv\) in keV  
\(A_0 \ldots A_3\) = coefficients of the fit.

The required data are the atomic weight, number of edges, and values of the edges and A's for each interval between edges. Samples of the absorption data input are given in Appendix C.

The deposition data for any radiation problem include the number of spectra or sources (NSPEC), angles (ANGLE) between the shine direction and normal incidence, type of deposition, fluence (ECAL), and shine duration (SSTOP-START). For black body spectra, one line containing the temperature (TEMP) and fluence (ECAL) is required for each black body.
FIGURE 5.3 TYPICAL VARIATION OF PHOTOELECTRIC MASS ABSORPTION COEFFICIENT WITH PHOTON ENERGY
For an arbitrary spectrum, the additional data are the number of hv values (NHNU, not more than 109), the format for reading the table of data, and NHNU pairs of hv (TBL) and energy (El) values.

The data for a depth-dose profile are the number of pairs of points (NPOINT), the format for reading the profile data, and the pairs of depth (TBL) and dose (El) values. Samples of all these radiation options are given in Appendix C.

5.4.3 Special Features

Many special features are often required for handling radiation problems: the available options are mentioned here.

Angles. If the layers are positioned at different inclinations, several values of ANGLE are required. The multiple angles require that positions 21 to 27 of the NSPEC line contain "ANGLES." Otherwise, all layers are assumed to have the inclination ANGLE(1).

Impulse. The impulse calculated by the McCloskey-Thompson formula is computed at each coordinate point. The impulse at point J is

$$I_J = 1.2 \sqrt{\int_1^J Z \left( E - E_m (1 + \ln \frac{E}{E_m}) \right) dZ}$$

where $E$ = the deposited energy at a point

$E_m$ = the melt energy

$Z$ = the mass per unit area

$I_J$ = impulse in dyn-sec/cm$^2$.

Multiple Sources. The present arrays are dimensioned for five sources. If more are required, SSTOP and START in the COMMON labeled /RAD/ should be redimensioned. The SS array may also require more storage. The SS array in labeled COMMON/SS/ should have a dimension at least as large as the number of sources times the number of coordinates.

Source Type Indicator. The source type indicator A1 is on the spectrum name line following the NSPEC line. A1 fills the 5 spaces
Normalization of the Depth-Dose Profile. The depth-dose profile may be modified to permit changes in material density and in the fluence. For a porous material, the depth-dose profile is modified by changes in density only in proportion to the ratio of densities. To permit a density change, NARB is set less than zero on the line following the NSPEC line, and an additional line containing RHOOLD (the density associated with the depth-dose profile) is inserted. Then the profile is automatically adjusted for the new density.

The input depths in the depth-dose profile need not correspond to the x-values in the coordinate array because the depths will all be adjusted to match the first coordinate of each layer.

The depth-dose profile is usually provided normalized to a fluence of 1 cal/cm². Then the input variable ECAL is multiplied times the dose energies to obtain the energy in the problem. If the profile is not normalized, the fluence ECAL can be obtained by setting NARB to ±1. Then the profile is normalized before applying the factor ECAL.

5.5 Initialization of Arrays and Indicators

The input data are used to initialize the cell and coordinate arrays and various indicators. Included in this initialization are yield and work-hardening factors, sound speed, the H indicator array, the NEM, NET and LVAR arrays, and several scalar indicators.

The standard deviator model treats a yield strength that varies with work-hardening and Coulomb friction as follows:

\[ Y = Y_1 + Y_D \Delta \rho + \beta P = YHL + YADD(M) \Delta \rho + EXMAT(M,1) \cdot P \]  

(5.12)

where

- \( Y_1 \) = the yield at the previous time (YHL)
- \( Y_D \) = a work-hardening modulus (YADD)
\( \beta \) = a Coulomb friction factor (EXMAT)

\( P \) = the pressure.

The input value of the work-hardening modulus, YADD, has the strange formulation inherited from PUFF 66, where the increase in yield strength is

\[
\Delta Y = \frac{\rho_2 - \rho_1}{\rho_0 (0.2 - \varepsilon_{EL})} \text{YADD} \quad (5.13)
\]

where

\( \rho_0, \rho_1 \) and \( \rho_2 \) = the initial density and the densities before and after a time increment

\( \varepsilon_{EL} \) = strain to the Hugoniot elastic limit = \( Y_o / (2G) \)

\( Y_o \) = input yield strength = YOS in the code

\( G \) = shear modulus = MU(M).

To put this work-hardening relation into the form of Eq. (5.12), the modulus \( Y_D \) is defined as

\[
Y_D = \frac{\text{YADD}(M)}{\text{RHOS}(M)*(0.2 - 0.5 * \text{YOS/MU}(M))} \quad (5.14)
\]

In GENRAT, \( \text{YADD}(M) \) is reset to \( Y_D \).

The value of \( \beta \) is derived by examining the usual form of the Coulomb law (actually Coulomb-without-dilatation, a special form that permits no plastic volume change):

\[
\tau_c = c + \sigma_N \tan \phi \quad (5.15)
\]

where

\( \tau_c \) = the shear stress at yield
\( c \) = the cohesion
\( \sigma_N \) = the normal stress on the yielding surface
\( \phi \) = the angle of internal friction.
As shown in Section 4, this Coulomb law can be rewritten into the following form

\[ Y = \frac{3c \sqrt{N_\phi}}{1 + N_\phi/2} + \frac{1.5 (N_\phi - 1)p}{1 + N_\phi/2} \]  

(5.16)

where \( N_\phi = \tan^2 (\pi/4 + \phi/2) \). Now Eq. (5.16) has the form of Eq. (5.12); we only need to determine the required constants from the input data. During input, YOS is read in with the valve 2c and EXMAT(M,1) is read in as \( \tan \phi \). Then YO and EXMAT are reset in GENRAT as follows:

\[ YO(M) = \frac{3c \sqrt{N_\phi}}{1 + N_\phi/2} \]  

(5.17)

\[ \text{EXMAT}(M,1) = \frac{1.5 (N_\phi - 1)}{1 + N_\phi/2} \]  

(5.18)

The sound speed, CHL, is initialized in GENRAT according to the following rules:

\[ \text{CHL} = \sqrt{\frac{\text{bulk modulus} + 4/3 \text{ (shear modulus)}}{\text{density}}} \]  

for normal solids

\[ = \text{detonation velocity for explosives} \]

\[ = \text{EXMAT}(M,3) \text{ for porous materials.} \]

Here the value of EXMAT(M,3) is calculated in the porous subroutine--POREQST, PORHOLT, PEST or CAP1--during its initialization and passed back to GENRAT.

For explosives that are to undergo either a running detonation or constant volume explosion, GENRAT calls EXPLODE to insert the chemical energy in the EHL array and initialize NEM to the fraction detonated.

For some deviator models the NEM array is given special initial values as follows:

- Band model: \( \text{NEM} = \text{TSR}(M,21) \)
- Gilman model: \( \text{NEM} = \text{TSR}(M,19) \)
- Bauschinger model: \( \text{NEM} = \text{yield strength.} \)
For the Band and Gilman models, the NEM values are the initial number of mobile dislocations.

The triple indicator array H is set so that H(J,1) shows the solid or porous state of the material, H(J,2) shows boundary conditions, and H(J,3) shows the path taken by the deviator stress. The boundary indicator has the meanings:

\[ H(J,2) = \begin{align*} 
N, & \text{ normal coordinate inside a material} \\
L, & \text{ left interface of a layer} \\
R, & \text{ right interface of a layer} \\
S, & \text{ spalled interface or free surface} \\
M, & \text{ mirror or symmetric boundary} \\
P, & \text{ pressure history boundary} \\
I, & \text{ infinite boundary.} 
\end{align*} \]

When extra cell variables in the COM array are required for a material model, NVAR is set by the user to the required number of variables. In GENRAT, NVAR is used to divide the COM array as described in Appendix C. The starting location in the COM array for variables of the Jth cell is LVAR(J): the LVAR array is initialized in GENRAT.

Several scalar indicators are also initialized in GENRAT. In non-radiation problems, the factor SDurm is set to 1.0 to eliminate calls to the deposition routines. For an impact problem, the particle velocity of the flyer materials is set to UZERO, the flyer velocity. For a symmetric impact, the velocity of the first boundary (the impact interface) is set to UZERO/2. The time-step variable DTNH is initialized to \(10^{-12}\) second to begin the first cycle of wave propagation calculations.

5.6 Initial Status Printouts

The initial configuration for the entire grid is printed out in either a deposition edit from DEPOS or velocity edit from GENRAT. Included in the deposition edit are the values of J, coordinate of each cell; DX,
the cell thickness; \( X \), the coordinate in inches and centimeters; four variables indicating the energy in the cells; the cell temperature in degrees centigrade; pressure from an instantaneous deposition; impulse from the McCloskey-Thompson integral; the material name, MATL; and the condition variables, H. The energy quantities are the deposited energy in \( \text{erg/g} \) and \( \text{cal/g} \), the cumulative amount of energy absorbed in \( \text{cal/cm}^2 \), and the fraction transmitted through each coordinate plane.

The velocity edit lists \( J, DX, X, U \) (particle velocity), yield strength, sound speed, density, spall strength, mass, internal energy and the H indicators. A sample edit listing is given in Figure 5.4.
6. PRINTED OUTPUT: GENRAT, EDIT, AND SCRIBE

Several types of printed output are provided during and at the conclusion of a calculation. During the reading of the input, the input lines are printed by GENRAT with some additional comments. Some material property subroutines read their own input and provide printout. After the input is read, a layout listing is given by GENRAT (or DEPOS for a deposition problem). During the calculation several listings of the layout with current cell variables are made by EDIT (on a call from SRI PUFF). A final EDIT listing is made at the end of the calculation. The SCRIBE is called by SRI PUFF to print historical listings of all requested variables. Besides these standard listings, there are error messages, periodic messages, and special listings by some material models. Samples of these listings are given in this section.

During the reading of input by GENRAT and other routines, an echo listing is made of the input, as shown in Figure 6.1. In addition to this echo printing, the GENRAT listing includes prints to the right of the input lines and some interpolated prints between input lines. The prints to the right show the contents of the first column on the input line (IND), the file containing the input (IN), and the units of the data if read in GENRAT. If the input is read by another subroutine, that subroutine's name is listed (e.g., DEPOS and EXTRA in Figure 6.1).

Inserted lines in the input listing include the spaces separating data groups and the notation of an end-of-file found by EXTRA. EQST provides messages when the McCloskey-Thompson logarithmic variation of sublimation energy is used: some messages are explanatory, others indicate errors that will cause a program stop. When either the Murnaghan or linear shock velocity Hugoniot forms are used, EQST provides a message. FMELT provides a message if the FMELT function does not monotonically decrease with increasing internal energy. EXPLODE lists
Figure 6.1 Sample Neutron Listing of Input Data for a Radiation Problem
the type of detonation that will occur, and the C-J parameters if a running detonation is indicated.

For all problems, a listing of the initial cell layout and principal cell quantities is given. A GENRAT layout listing is shown in Figure 6.2. A sample radiation deposition layout from DEPOS is in Figure 6.3. In the DEPOS listing, a J = 0 line is provided for each layer to permit printing quantities pertaining to the first coordinate point in addition to quantities for the first cell.

Following the layout listing is the printing from PRESCR of the variables for which a historical listing is requested. A sample is given in Figure 6.4. This list is provided before the propagation calculations so that a verification of the correct histories may be made without a complete run.

During the calculation there are usually many calls to EDIT to produce listings such as that in Figure 6.5. The last two columns contain a variety of variables depending on the material models used and the material state. For the explosive (COMPB) in the first layer, the penultimate column provides FBURN, the fraction of explosive detonated. For the HF-1 in the second layer, the columns initially contain the yield strength (Y) and deviator stress (SD), but after shear banding begins at a cell, they contain TAU = ENL^3 and N, where N is the number of shear bands per cubic centimeter and L is the radius of the bands.

A sample of the historical listings provided by SCRIBE at the end of a calculation is in Figure 6.6. The variables in the first columns are provided automatically: cycle number N, problem time TIME, time step DTNH, calculational time for the cycle DELTIM, and the cell number controlling the time step JTS. Interface stresses are labeled S-INT(n) where n is the interface number and n = 0 means the front surface. For all other quantities, a standard label for the quantity is followed by the J value in parentheses.

Figure 6.7 contains other listings and messages found in PUFF output. Every 25 cycles a message like the periodic print in the figure is given. Preceding the final EDIT listing at the termination of the run, there is
<table>
<thead>
<tr>
<th>J</th>
<th>DX</th>
<th>X(J)</th>
<th>U(J)</th>
<th>YHL(J)</th>
<th>CHL(J)</th>
<th>DHL(J)</th>
<th>T(J)</th>
<th>ZHL(J)</th>
<th>EHL(J)</th>
<th>MATERIAL</th>
<th>COND</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7.257E-01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>IMPACTOR</td>
<td>S</td>
<td>S 1</td>
</tr>
<tr>
<td>2</td>
<td>7.257E-01</td>
<td>7.257E-01</td>
<td>2.23E+03</td>
<td>1.222E+10</td>
<td>5.843E+05</td>
<td>7.850E+00-1.000E+11</td>
<td>5.697E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>7.257E-01</td>
<td>1.451E+00</td>
<td>2.23E+03</td>
<td>1.222E+10</td>
<td>5.843E+05</td>
<td>7.850E+00-1.000E+11</td>
<td>5.697E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>7.257E-01</td>
<td>2.177E+00</td>
<td>2.23E+03</td>
<td>1.222E+10</td>
<td>5.843E+05</td>
<td>7.850E+00-1.000E+11</td>
<td>5.697E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>7.257E-01</td>
<td>2.903E+00</td>
<td>2.23E+03</td>
<td>1.222E+10</td>
<td>5.843E+05</td>
<td>7.850E+00-1.000E+11</td>
<td>5.697E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>7.257E-01</td>
<td>3.629E+00</td>
<td>2.23E+03</td>
<td>1.222E+10</td>
<td>5.843E+05</td>
<td>7.850E+00-1.000E+11</td>
<td>5.697E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>7.257E-01</td>
<td>4.354E+00</td>
<td>2.23E+03</td>
<td>1.222E+10</td>
<td>5.843E+05</td>
<td>7.850E+00-1.000E+11</td>
<td>5.697E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>7.257E-01</td>
<td>5.080E+00</td>
<td>2.23E+03</td>
<td>1.222E+10</td>
<td>5.843E+05</td>
<td>7.850E+00-1.000E+11</td>
<td>5.697E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>7.257E-01</td>
<td>5.806E+00</td>
<td>2.23E+03</td>
<td>1.222E+10</td>
<td>5.843E+05</td>
<td>7.850E+00-1.000E+11</td>
<td>5.697E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>7.257E-01</td>
<td>6.531E+00</td>
<td>2.23E+03</td>
<td>1.222E+10</td>
<td>5.843E+05</td>
<td>7.850E+00-1.000E+11</td>
<td>5.697E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>7.257E-01</td>
<td>7.257E+00</td>
<td>2.23E+03</td>
<td>1.222E+10</td>
<td>5.843E+05</td>
<td>7.850E+00-1.000E+11</td>
<td>5.697E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>7.257E-01</td>
<td>7.983E+00</td>
<td>2.23E+03</td>
<td>1.222E+10</td>
<td>5.843E+05</td>
<td>7.850E+00-1.000E+11</td>
<td>5.697E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>7.257E-01</td>
<td>8.709E+00</td>
<td>2.23E+03</td>
<td>1.222E+10</td>
<td>5.843E+05</td>
<td>7.850E+00-1.000E+11</td>
<td>5.697E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>7.257E-01</td>
<td>9.434E+00</td>
<td>2.23E+03</td>
<td>1.222E+10</td>
<td>5.843E+05</td>
<td>7.850E+00-1.000E+11</td>
<td>5.697E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>0.</td>
<td>1.016E+01</td>
<td>2.23E+03</td>
<td>1.222E+10</td>
<td>5.843E+05</td>
<td>7.850E+00-1.000E+11</td>
<td>5.697E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>5.080E-01</td>
<td>1.016E+01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td>4.995E+05</td>
<td>2.220E+00-1.000E+11</td>
<td>1.128E+00</td>
<td>0.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>0.</td>
<td>1.067E+01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td>4.995E+05</td>
<td>2.220E+00-1.000E+10</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>5.080E-01</td>
<td>1.067E+01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td>5.458E+05</td>
<td>2.501E+00-1.000E+11</td>
<td>1.271E+00</td>
<td>0.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>0.</td>
<td>1.118E+01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td>5.458E+05</td>
<td>2.501E+00-1.000E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>5.080E-01</td>
<td>1.118E+01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td>4.995E+05</td>
<td>2.220E+00-1.000E+11</td>
<td>1.128E+00</td>
<td>0.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>5.080E-01</td>
<td>1.168E+01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td>4.995E+05</td>
<td>2.220E+00-1.000E+11</td>
<td>1.128E+00</td>
<td>0.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>5.080E-01</td>
<td>1.219E+01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td>4.995E+05</td>
<td>2.220E+00-1.000E+11</td>
<td>1.128E+00</td>
<td>0.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>5.080E-01</td>
<td>1.279E+01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td>4.995E+05</td>
<td>2.220E+00-1.000E+11</td>
<td>1.128E+00</td>
<td>0.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>5.080E-01</td>
<td>1.321E+01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td>4.995E+05</td>
<td>2.220E+00-1.000E+11</td>
<td>1.128E+00</td>
<td>0.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>5.080E-01</td>
<td>1.372E+01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td>4.995E+05</td>
<td>2.220E+00-1.000E+11</td>
<td>1.128E+00</td>
<td>0.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>0.</td>
<td>1.422E+01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td>4.995E+05</td>
<td>2.220E+00-1.00E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>5.080E-01</td>
<td>1.422E+01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td>5.458E+05</td>
<td>2.501E+00-1.000E+11</td>
<td>1.271E+00</td>
<td>0.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>0.</td>
<td>1.473E+01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td>5.458E+05</td>
<td>2.501E+00-1.000E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>5.080E-01</td>
<td>1.473E+01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td>4.995E+05</td>
<td>2.220E+00-1.000E+11</td>
<td>1.128E+00</td>
<td>0.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>0.</td>
<td>1.524E+01</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
<td>4.995E+05</td>
<td>2.220E+00-1.000E+00</td>
<td>0.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Time to complete GENRAT is 530 seconds.

FIGURE 6.2 SAMPLE GENRAT LISTING OF THE CELL LAYOUT FOR AN IMPACT OF A STEEL PROJECTILE ONTO REINFORCED CONCRETE
<table>
<thead>
<tr>
<th>OUTPUT FROM PRESCR</th>
</tr>
</thead>
<tbody>
<tr>
<td>S-INT( 1) S1( 20) S1( 21) S1( 22) S1( 23) S1( 24) S1( 25) SD1( 20) SD1( 21) SD1( 22) SD1( 23) SD1( 24)</td>
</tr>
<tr>
<td>SD1( 25) Y( 20) Y( 21) Y( 22) Y( 23) Y( 24) Y( 25) NEW( 20) NEW( 21) NEW( 22) NEW( 23) NEW( 24)</td>
</tr>
<tr>
<td>NEM( 25) NET( 20) NET( 21) NET( 22) NET( 23) NET( 24) NET( 25)</td>
</tr>
<tr>
<td>S-INT( 1) S-INT( 2) S-INT( 3) S1( 1) S1( 3) S1( 3) S1( 8) S1( 10) S1( 13) S1( 17) S2( 1) S2( 3) S2( 3) S2( 8) S2( 10) S2( 13) S2( 17) COM 7( 1) COM 7( 3) COM 7( 3) COM 7( 3) COM 7( 8) COM 7( 10) COM 7( 13)</td>
</tr>
<tr>
<td>COM 7( 17) COM 12( 1) COM 12( 3) COM 12( 3) COM 12( 8) COM 12( 10) COM 12( 13) COM 12( 17)</td>
</tr>
</tbody>
</table>

**Figure 6.4** Sample listing of output from PRESCR showing variables for which historical listings will be made.
**DATE = 77/02/17, IDENT FR & FRAGMENT RT OF HF-1 TO SIMULATE CROW'S TESTS 1 & 2.**

**TIME EDIT NO. 30 AT N = 300, TIME = 5.03722E-05 SECS, JSTAR = 22, CALC TIME IS 48,348 SECS, DTINV = 1.149E-07 SECS**

<table>
<thead>
<tr>
<th>J</th>
<th>X</th>
<th>U</th>
<th>DPL</th>
<th>PML</th>
<th>SML</th>
<th>EML</th>
<th>DML</th>
<th>CHL</th>
<th>COND</th>
<th>MGM</th>
<th>TAPS</th>
<th>FRURN</th>
<th>NET</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0.000000</td>
<td>0.01</td>
<td>0.162E+09</td>
<td>1.162E+09</td>
<td>1.162E+09</td>
<td>1.980E+09</td>
<td>1.3187</td>
<td>0.017E+05</td>
<td>S B CCMB</td>
<td>0.0</td>
<td>1.00E+00</td>
<td>0.0</td>
</tr>
<tr>
<td>2</td>
<td>1.685987</td>
<td>1.4551</td>
<td>0.110E+09</td>
<td>1.305E+09</td>
<td>1.305E+09</td>
<td>2.123E+09</td>
<td>0.3344</td>
<td>0.053E+05</td>
<td>S B CCMB</td>
<td>1.817E+03</td>
<td>1.00E-00</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1.741660</td>
<td>3.5093</td>
<td>0.158E+09</td>
<td>1.581E+09</td>
<td>1.581E+09</td>
<td>2.410E+09</td>
<td>0.3563</td>
<td>1.122E+05</td>
<td>S B CCMB</td>
<td>1.884E+04</td>
<td>1.00E-00</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>2.555809</td>
<td>4.6000</td>
<td>0.189E+09</td>
<td>1.899E+09</td>
<td>1.899E+09</td>
<td>2.717E+09</td>
<td>0.3796</td>
<td>1.192E+05</td>
<td>S B CCMB</td>
<td>6.411E+04</td>
<td>1.00E-00</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>3.340054</td>
<td>5.1712</td>
<td>0.183E+09</td>
<td>1.834E+09</td>
<td>1.834E+09</td>
<td>2.655E+09</td>
<td>0.3752</td>
<td>1.175E+05</td>
<td>S B CCMB</td>
<td>1.456E+05</td>
<td>1.00E-00</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>4.141754</td>
<td>5.9596</td>
<td>0.161E+09</td>
<td>1.613E+09</td>
<td>1.613E+09</td>
<td>2.443E+09</td>
<td>0.3547</td>
<td>1.130E+05</td>
<td>S B CCMB</td>
<td>2.707E+05</td>
<td>1.00E-00</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>4.981754</td>
<td>6.2292</td>
<td>0.179E+09</td>
<td>1.793E+09</td>
<td>1.793E+09</td>
<td>2.618E+09</td>
<td>0.3721</td>
<td>1.170E+05</td>
<td>S B CCMB</td>
<td>4.301E+05</td>
<td>1.00E-00</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>5.792237</td>
<td>6.1817</td>
<td>0.181E+09</td>
<td>1.813E+09</td>
<td>1.813E+09</td>
<td>2.639E+09</td>
<td>0.3732</td>
<td>1.174E+05</td>
<td>S B CCMB</td>
<td>6.449E+05</td>
<td>1.00E-00</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>6.033248</td>
<td>8.0913</td>
<td>0.171E+09</td>
<td>1.712E+09</td>
<td>1.712E+09</td>
<td>2.548E+09</td>
<td>0.3650</td>
<td>1.154E+05</td>
<td>S B CCMB</td>
<td>9.185E+05</td>
<td>1.00E-00</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>7.433143</td>
<td>9.5163</td>
<td>0.165E+09</td>
<td>1.653E+09</td>
<td>1.653E+09</td>
<td>2.523E+09</td>
<td>0.3554</td>
<td>1.148E+05</td>
<td>S B CCMB</td>
<td>1.292E+06</td>
<td>1.00E-00</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>8.282992</td>
<td>10.9313</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>4.469E+10</td>
<td>1.7200</td>
<td>7.950E+05</td>
<td>S B CCMB</td>
<td>1.777E+06</td>
<td>1.00E-00</td>
<td>0.0</td>
</tr>
</tbody>
</table>

**FIGURE 6.5 SAMPLE EDIT LISTING OF CELL VARIABLES AT ONE TIME DURING THE CALCULATION**
PERIODIC PRINT

N  =  300;  JSTAN= 22;  TIME= 5.037E-05;  CALC TIME= 183.290 SECS;  JTS= 14  DTAN= 1.189E-07  SMAX= 1.059E+09  JSMAX= 4

STOP CRITERION

***** CRITERION FOR STOP *****

N  =  300;  JCYS= 300;  TIME= 5.037E-05;  TS = 8.000E-05;  X(JSMAX) = 7.555E+00;  CKS = 3.000E+01;  LSUB(7) = 0;  DTAN= 1.189E-07

SHEAR2 OUTPUT

K  =  21  J=  21  IM=  16  HT=  0.  
NG=  4
CN=  .513E+01  .266E+01  .133E+01  .595E+00  .222E+00  .635E+01  .124E-01  .141E-02
CL=  .101E+00  .242E+00  .440E+00  .717E+00  .116E+01  .151E-01
NG=  5
CN=  .159E+01  .102E+01  .576E+00  .276E+00  .107E+00  .307E+01  .591E+02  .856E-03
CL=  .103E+01  .247E+01  .449E+00  .732E+01  .113E+00  .168E+00  .246E+00  .354E+00
NG=  6
CN=  .169E+01  .109E+01  .615E+00  .293E+00  .112E+00  .322E+01  .619E+02  .884E-03
CL=  .888E+02  .213E+01  .387E+01  .631E+01  .972E+01  .145E+00  .212E+00  .305E+00
TOT PL STRAIN = 0.000  0.000  0.000  0.256  0.224  0.219
TAUZ=  0.0000000  0.000000  0.000000  0.697120  0.00435  0.00293

BFRACT3 OUTPUT

FRAG N=  4.008E+01  1.142E+01  1.142E+01  5.294E+01  5.294E+01  5.294E+01  5.294E+01
J=  1  CJ=  2.495E+01  2.495E+01  2.495E+01  2.495E+01  2.495E+01
K=  2  CJ=  5.332E-01  5.332E-01  5.332E-01  5.332E-01  5.332E-01
V:  1.31E+02  3.452E-03  3.452E-03  3.452E-03  3.452E-03

FIGURE 6.7 MISCELLANEOUS MESSAGES AND LISTINGS GENERATED DURING CALCULATIONS
a message containing the criteria used for stopping the run, as in Figure 6.7. In this case the halt occurred when \( N = JCYCS \). Other possibilities are \( \text{TIME} \geq \text{TS} \), \( X(JSMAX) \geq \text{CKS} \), \( \text{LSUB}(7) = 1 \), and \( \text{DTNH} < 1.\text{E} - 12 \). \( \text{LSUB}(7) \) is set to 1 in \text{HSTRESS} and \text{FMELT} to trigger an error stop.

Several material property subroutines provide regular listings in the cycle just preceding an \text{EDIT}. The samples in Figure 6.7 are from \text{SHEAR2} and \text{BFRACT3}. \text{EXPLODE} also prints a line whenever the detonation is completed at a cell. If the iterations do not converge in \text{CAP1}, \text{BFRACT3}, \text{REBAR}, \text{PEST}, \text{EQSTPF}, \text{TSQE}, \text{BECOM}, \) or \text{DFRACT}, an error message and some information about the cause and location of failure is given. \text{REZONE} lists all its major operations so that difficulties can be traced.
Appendix A

THERMAL ENERGY DEPOSITION

In SRI PUFF, radiant energy is deposited gradually into the finite difference cells over a time corresponding to the source duration. This appendix gives some background on source characteristics, radiation absorption information for materials, and procedures for depositing the energy into the material layers for both normal and oblique incidence of the radiation. These processes are all treated in the DEPOS subroutine. The interpolation procedure used with depth-dose profiles and contained in SCATTO is also described.

Specific information for constructing the input deck for radiation problems is in Section 5.4, and sample input is given in Appendix C.

Radiation Absorption Characteristics

The radiation absorption calculations in PUFF provide a means for determining the radiant energy absorbed in each finite difference cell for x-ray sources. Only absorption associated with the photoelectric effect is considered in the calculations. If scattering and fluorescence are important, as they are for photon energies larger than a few keV, an appropriate deposition code like FSCATT should be used to obtain a depth-dose profile for the PUFF calculations.

The geometry assumed in the absorption calculations is planar. Cylindrical or spherical geometries must be treated by means of a depth-dose profile or by detailed initialization of internal energy (EHL) or the SS array through a NAMELIST statement.
Typical radiation absorption characteristics associated with the photoelectric effect are illustrated in Figure A.1. The sharp discontinuities in the absorption occur at photon energies related to the orbits of the electrons. The discontinuity farthest to the right is called the K edge because it is associated with electrons in the K shell. The next edges to the left are L, M, and N edges. Between the edges the absorption function varies smoothly, approximately following the function

$$\sigma_a \propto (hv)^{-3}$$  \hspace{1cm} (A.1)

where $\sigma_a$ is the mass absorption coefficient and $hv$ is the photon energy ($\nu$ is frequency and $h$ is Planck's constant). In standard tables, such as those of McMasters et. al. and Fisher and Wiehe, the absorption coefficient is expanded in the following form between edges:

$$\ln \sigma_a = A_0 + A_1 w + A_2 w^2 + A_3 w^3$$ \hspace{1cm} (A.2)

where $\sigma_a$ the mass absorption coefficient, barn/atom

- $w = \ln(hv)$
- $hv$ = photon energy, keV
- $A_0 \ldots A_3$ = coefficients of the fit.

Because the absorption coefficient follows Eq. (A.1), $A_1$ is approximately equal to -3.

For absorption by the photoelectric effect, there is an exponential attenuation of energy through a layer of material. The fraction of the fluence $I_0$ (with a specific photon energy) transmitted through a thickness $\Delta X$ is

$$\frac{I}{I_0} = \exp (-\mu_a \Delta X)$$ \hspace{1cm} (A.3)
FIGURE A.1  TYPICAL VARIATION OF PHOTOELECTRIC MASS ABSORPTION COEFFICIENT WITH PHOTON ENERGY
where $\mu_a$ is the linear absorption coefficient with units of $1/cm$
appropriate to the incident photon energy (Here we are considering
normal incidence only; Section 4 treats the case of oblique incidence.)
The coefficient $\mu_a$ is related to $\sigma_a$ as follows.

$$\mu_a = \frac{\rho C_b N_a \sigma_a}{A_w} = 0.602252 \frac{\rho \sigma_a}{A_w}$$  \hspace{1cm} (A.4)

where $\rho = $ density, $g/cm^3$

$C_b = 10^{-24} cm^2/barn$, a conversion factor

$N_a = 6.02252 \times 10^{23}$, Avogadro's number, atom/mole

$A_w = $ atomic weight, $g/mole$

$\sigma_a = $ mass absorption coefficient, barn/atom.

With the coefficients $A_0, \ldots, A_3$ and Eqs. (A.2) to (A.4), the attenuation
and absorption of energy can be calculated for any source with a single
photon energy.

$$\mu_a = 0.602252 \frac{\rho}{A_w} \sum_{i=0}^{3} A_i [\ln(h\nu)]^i$$  \hspace{1cm} (A.5)

Use of these absorption characteristics to treat attenuation of radiation
from a source with a range of photon energies is described in the
following sections of this appendix.

For multiple constituent materials, absorption coefficient informa-
tion is entered for each constituent. Such materials may be either
mixtures or compounds of any kind. Common examples are a metal alloy
or an epoxy resin. In such materials the absorption coefficients are
defined and entered in the usual fashion for each constituent, and then
a composite absorption coefficient is calculated in the program. The
composite absorption coefficient is
\[ \mu_a = 0.602252 \sum_{n=1}^{N_c} \frac{A_{wn}}{A_n} \sum_{i=0}^{3} A_{in} \ln(\nu) \]  
(A.6)

where \( N_c \) = the number of constituents

\( A_{wn} \) = the atomic weight of the nth constituent

\( A_{in} \) = the coefficients in the absorption function for the nth constituent.

Here \( \rho_n \) is the weight fraction of the nth constituent times the composite density. Hence

\[ \rho = \sum_{n=1}^{N_c} \rho_n \]  
(A.7)

**Radiation Sources**

Since the radiation sources permitted in the program are all steady state, only an emittance history and a single emittance spectrum are required. The emittance or flux history is that shown in Figure A.2, with an abrupt start, a constant value for the duration of deposition, and an abrupt stop.

Two types of sources are accounted for in the absorption calculations: an arbitrary spectrum and one made up of several black body radiators. For the arbitrary spectrum the user divides the energy into several energy packets, each at a specific photon energy, and pairs of values of energy and \( h\nu \) (photon energy) are read in.

For the black body source, some standardization is possible because of the simple relation between radiant emittance and the photon energy. According to Sears \(^{43}\) for a black body of unit energy, the radiant emittance \( dW \) is

\[ dW = \frac{15}{\pi^4} \left( \frac{\omega^3}{e^\omega - 1} \right) d\omega \]  
(A.8)
Figure A.2  History of radiation sources considered in the program.
where

\[ \omega = \frac{h \nu}{kT}, \]  

a nondimensional quantity proportional to photon energy

\[ h = \text{Planck's constant} \]

\[ \nu = \text{frequency of the photons} \]

\( h \nu = \text{photon energy, usually in keV} \)

\[ k = \text{Boltzmann constant} \]

\[ T = \text{Kelvin temperature} \]

\[ kT = \text{temperature in energy units, usually keV (Planckian temperature).} \]

The variation of radiant emittance with photon energy is shown in Figure A.3. The total emittance of the black body is the area under the curve. For calculations in the program, the spectrum has been divided into 95 energy packets. Each energy packet is located at a discrete \( h \nu \) value (BBDY in the program). The energy (EIBB) in each packet was determined by integrating the area under the emittance curve over appropriate ranges of \( h \nu \) to determine \( dW \) from Eq. (A.8) (as shown in Figure A.4). The black body spectrum is completely specified by a Planckian Temperature \( kT \) (TEMP in the program, keV) and the total fluence (ECAL, cal/cm\(^2\)).

**Deposition Computations**

Radiation deposition by means of an absorption calculation is provided for two types of sources: a black body or bodies, and an arbitrary spectrum. The deposition of radiation from either a black body or an arbitrary spectrum is obtained by computing the absorption of each energy packet (located at a discrete value of \( h \nu \)) using the absorption coefficient corresponding to that value of \( h \nu \). The penetration of the radiant energy into the material is given by an exponential relation as shown in Figure A.5. Then within a cell thickness \( X \), the increment of energy is
PEAK AT $\omega = 2.83$

FIGURE A.3  EMITTANCE SPECTRUM FOR BLACK BODY
FIGURE A.4  SPECTRUM OF A RADIATION SOURCE

\[ dW = W_{\nu_i} d(\nu) \]
\[ \Delta E = E_1 \left( 1 - e^{-\mu_a \Delta X} \right) \]  

(A.9)

where

\[ E_1 = \text{the amount of energy reaching the left face of the cell} \]

\[ \Delta X = \text{the thickness of the cell}. \]

Because \( \mu_a \) is a function of \( h \nu \), Eq. (A.9) can be used only for particular values of \( h \nu \), that is, for energy packets located at the \( h \nu \) values. To provide reasonable accuracy in the deposition, it is necessary to provide a large number of \( h \nu \) values (109 values of \( h \nu \) are permitted in the present dimension statement). The large number of \( h \nu \) values is desirable because the program selects a single value of \( \mu_a \) for each abscissa, and the function of \( \mu_a \) versus \( h \nu \) is extremely uneven, as shown in Figure A.1.

In DEPOS the deposition into the grid is accomplished by inserting the energy from the various spectral sources into an array \( SS \) for each cell. During the wave propagation calculations, this energy will be gradually inserted into the internal energy in the cell. A value of \( SS \) is computed for each cell and for each source. The equation for the energy deposited in the \( j \)th cell in an increment of time \( \Delta t \) is given by

\[ \Delta E_j = C \frac{E_{R,j}^n \Delta t}{Z_j \Delta T^n} \]  

(A.10)

where

\[ E_{R,j}^n = \text{the total energy in cal/cm}^2 \text{ to be deposited in the} \]

\[ \text{\( j \)th cell from the \( n \)th source} \]

\[ C = \text{a conversion factor, } 4.186 \times 10^7 \text{ erg/cal} \]

\[ Z_j = \text{the mass of the \( j \)th cell, g/cm}^2 \]

\[ \Delta T^n = \text{the duration of the \( n \)th source}. \]
FIGURE A.5  PENETRATION OF ENERGY INTO A MATERIAL

\[ E = E_0 e^{-\mu X} \]
Then the deposited energy $\Delta E_j$ is in erg/g. The array SS is defined to include all the constant quantities in Eq. (A.10), that is, all except $\Delta t$.

$$S_{S_j}^n = \frac{C E^n}{E_{j}^{R_n} \Delta T^n}$$  \hspace{1cm} (A.11)

During the wave propagation calculations, the manipulations with the array SS are conducted in the function SSCALH.

**Radiation Deposition at Oblique Incidence**

For a monoenergetic source at normal incidence, the radiation is absorbed into a material according to the standard exponential law:

$$E = E_0 e^{-\mu X}$$  \hspace{1cm} (A.12)

where

- $E_0$ = the incident energy
- $E$ = the intensity at any depth, $X$.

If the incidence is not normal then Eq. (A.12) is modified in two ways: the intensity at the front is reduced by the cosine of the angle, and the depth is increased by the cosine. Thus the equation becomes

$$E_s = E_0 \cos \theta e^{-\mu X \sec \theta}$$  \hspace{1cm} (A.13)

where $\theta$ is the angle from normal incidence. Equation (A.13) is shown in Figure A.6.

The absorbed energy in erg/g is determined as the difference between incident and transmitted fluence, divided by the mass. Considering a small cell of material with lengths $\Delta X$, $\Delta Y$, and $\Delta Z$, the incident fluence is
FIGURE A.6 RADIATION INTENSITY AS A FUNCTION OF DEPTH FOR SEVERAL INCIDENT ANGLES

\[ E_s = E_0 \cos \theta e^{-\mu a x \sec \theta} \]
\[ E_{si} \Delta Y \Delta Z = E_o \cos \theta e^{-\mu_l X_1 \sec \theta} \Delta Y \Delta Z \]

and the mass is

\[ Z_j = \rho \Delta X \Delta Y \Delta Z \]

Therefore, the absorbed energy is

\[ \Delta E_a = \frac{(E_{si} - E_{s2}) \Delta Y \Delta Z}{\rho \Delta X \Delta Y \Delta Z} \]

\[ = \frac{E_o \cos \theta e^{-\mu_a X \sec \theta}}{Z_j} \]

(A.14)

Depth-Dose Profile Interpolation

When a depth-dose profile for the radiation is provided by a table of energy-distance values, the energy for each PUFF cell is determined by interpolation. These interpolations are performed in the subroutine SCATTO. The depth-dose profile may represent depositions from an electron beam, a laser, or an x-ray source, and may be determined either experimentally or analytically. To account for x-ray absorption by scattering, fluorescence, and the photoelectric effect, we have used the FSCATT code of Fisher and Wiehe. The FSCATT results provide deposited energy (e.g., cal/g) at coordinate points in a finite difference grid for a unit of radiant energy (e.g., 1 cal/cm²). All depth-dose profiles are assumed to have this form.

For PUFF calculations the deposited energy is an average quantity over the cell thickness, whereas the depth-dose profile provides energies at discrete depths. The PUFF cell energies are derived by interpolating between points in the depth-dose profile and then integrating over the PUFF cell dimensions.
The approach taken for the interpolation is to assume that the deposited energy is representable by a smooth function that can be defined by energy values at the depths given in the depth-dose profile. This function is then integrated over each PUFF cell dimension to find the energy deposited therein. The energy is assumed to span across three depths in the given profile and to have the form of a parabola in a semi-log plot. An expression for this parabolic form is

$$E_s = E_{s1} \cdot \xi_1 \cdot E_{s2} \cdot \xi_2 \cdot E_{s3} \cdot \xi_3$$ \hspace{1cm} (A.15)

where $E_s$ = the energy at any depth $E_{s1}, E_{s2}, E_{s3}$ = energies at the given depths in the depth-dose profile

$$\xi_1 = \frac{(X - X_2)(X - X_3)}{(X_1 - X_2)(X_1 - X_3)}$$

$$\xi_2 = \frac{(X - X_1)(X - X_3)}{(X_2 - X_1)(X_2 - X_3)}$$

$$\xi_3 = \frac{(X - X_1)(X - X_2)}{(X_3 - X_1)(X_3 - X_2)}$$

$X_1$ = depths in the depth-dose profile.

The form of Eq. (A.15) is suggested by the shapes of deposition curves that are essentially exponential, except near material boundaries, where they may be more rounded. The energy ($E_j^*$) deposited in the jth PUFF cell per unit of fluence is the average of $E_s$ between the cell coordinates, $X_j$ and $X_{j+1}$. This average is expressed by the integral

$$E_j^* = \frac{1}{X_{j+1} - X_j} \int_{X_j}^{X_{j+1}} E_s \, dX$$ \hspace{1cm} (A.16)

This integration is performed numerically using Simpson's rule.
The function in Eq. (A.15) best represents the variation of deposited energy in some middle portion of the three depths used in the interpolation. Therefore, it was decided to use the function defined by three depths only from the middle of the first pair of depths to the middle of the second pair. Figure A.7 shows the profile depths that contribute to the deposition in each PUFF cell.

The final step in the deposition is to initialize the SS array in a manner similar to that described above for deposition computations. The energy in each PUFF cell, $E_j'$, is based on one unit of radiated energy. Therefore, the actual absorbed energy in any cell from a source with a total fluence of $E_{\text{cal}}$ is $E_j' \cdot E_{\text{cal}}$. Then the expression for computing values for the SS array is

$$SS_j^n = \frac{C \cdot E_j' \cdot E_{\text{cal}}}{\Delta T^n}$$

The SS array is used in the function SSCALH to provide energy increments for each cell during the propagation calculations.
FIGURE A.7  PATTERNS FOR INTERPOLATION OF THE RADIANT ENERGY IN A DEPTH-DOSE PROFILE TO OBTAIN ENERGIES FOR PUFF CELLS
Appendix B

CALCULATIONS FOR EXPLOSIVES

This appendix outlines a simple detonation theory based on standard references such as Taylor. Then the types of detonation provided in PUFF, the input required, and the algebra of the code calculations are described.

Background on Detonation Processes

Three substances are involved in a detonation process: the unreacted explosive, the reacting explosive, and the product gases. Here we will presume that the unreacted explosive and the product gases can be represented by equations of state with the pressure-volume isentropes shown in Figure B.1. During detonation, the chemical energy in the explosive is transformed to internal energy and the state point moves from the unreacted curve to the product curve of Figure B.1. In Chapman Jouguet detonation theory, the reaction occurs within the shock front. In a steady detonation, the material follows a Rayleigh line from the initial density to a point of tangency on the products curve as shown. The point of tangency is the Chapman-Jouguet or C-J point. The pressure, volume, and energy at this point are labeled $P_{\text{CJ}}, V_{\text{CJ}},$ and $E_{\text{CJ}}$. If the product gases are assumed to follow a polytropic gas equation of state, that is,

$$PV^\gamma = \text{constant} \quad (B.1)$$

then relations for the detonation velocity ($D_x$), $P_{\text{CJ}}, V_{\text{CJ}}, E_{\text{CJ}},$ and the particle velocity ($u_{\text{CJ}}$) can be derived. These are all derived from the polytropic gas relations, Hugoniot jump conditions, energy conservation, and the condition of tangency at the C-J point.

$$D_x = \sqrt{\frac{2Q_x(\gamma + 1)(\gamma - 1)}{\gamma}} \quad (B.2)$$
FIGURE B.1 PRESSURE-VOLUME PATHS FOLLOWED IN DETONATION PROCESS
\[ P_{\text{CJ}} = 2Q_x (\gamma - 1) \rho_o \] (B.3)

\[ V_{\text{CJ}} = \frac{\gamma}{\rho_o (\gamma + 1)} \] (B.4)

\[ E_{\text{CJ}} = \frac{2Q_x \gamma}{\gamma + 1} \] (B.5)

\[ u_{\text{CJ}} = \sqrt{\frac{2Q_x (\gamma - 1)}{\gamma + 1}} \] (B.6)

where \( Q_x \) = the energy of the explosive

\( \rho_o \) = the initial density.

The polytropic gas exponent is related to the Grüneisen ratio as follows

\[ \gamma = \Gamma + 1 \] (B.7)

For many common explosives, \( \gamma \) values range from 2.5 to 3.0. This exponent describes the product gas isentrope adequately down to a few kilobars. For lower pressures, the apparent \( \gamma \) value decreases gradually to about 1.5 at ambient conditions.

Besides the Chapman-Jouguet process, several other detonation processes may occur in explosives. Von Neumann suggested that in a steady-state running detonation, the pressure in the shock rises to the point V.N. in Figure B.1 and then reduces gradually to C-J as the chemical reaction occurs. Path C is typical of computed pressure-volume paths followed during the buildup to a steady-state detonation. Here the chemical reaction is occurring during the loading by the stress wave. If the explosion occurs without a change in volume, the vertical path to the constant-volume point C-V is followed. The Chapman-Jouguet, von Neumann, constant-volume, and various gradual detonation processes have all been used to represent explosive phenomena. Only the Chapman-Jouguet and constant-volume processes are currently available in PUFF.

The detonation type used in the calculation should match as nearly as possible the explosive behavior and geometry being considered. For example, if a block of explosive next to a plate is detonated at a point
on the block opposite the plate, the detonation front will reach the plate as a plane wave; this process should be simulated as a running detonation. If the detonation occurs such that the wave front sweeps past the plate, however, a constant-volume explosion gives a better representation of the impulse applied to the plate (the actual wave front is not moving in the direction of motion in PUFF). In some problems the stress histories in the explosive are not important (as in the impact of an explosively driven flyer plate); then a constant-volume calculation will adequately represent the impulse applied by the explosive.

Computation of Detonation Processes with the Subroutine EXPLODE

The Chapman-Jouguet and constant-volume detonation processes are incorporated into the EXPLODE subroutine. This routine may be called to perform three different functions: reading input, initializing cells, and computing the pressure for the running detonation.

The input for an explosive calculation includes $Q_x$, $X_D$, and $b$ and is read during the first call to EXPLODE from GENRAT. If a constant-volume explosion is desired, only the chemical energy $Q_x$ is provided. $X_D$ is the initiation point for a running detonation and $b$ is the number of cells over which a detonation front is spread. Nominal values of $b$ are 2 to 4.

At the second call to EXPLODE, the energy and density of cells containing explosive are initialized. This call is made from GENRAT during the cell layout process. For a constant-volume explosion, the internal energy is equated to $Q_x$, and $F_B$ (the detonated fraction) is set to 1.0 to show that detonation has taken place. The calculations of pressure during the propagation process are then all treated in a section of EQST.

For a running detonation, only cells near the detonation point are initialized at the second call to EXPLODE. The reacted fraction $F_B$ of a cell is computed based on the distance of the cell midpoint from the initiation point.
\[ F_B = 1 - \frac{|X - X_D|}{bAX} \]  

(B.8)

where \( X \) = the cell midpoint  
\( AX \) = the cell length in the direction of propagation.

From eq. (B.8) it appears that the cell midpoint must be within a distance \( bAX \) of the initiation point for any initiation to occur. For \( F_B > 0 \), the pressure, density, and internal energy are augmented to represent a point along the C-J detonation path in Figure B.1. Hence

\[ P = P_{CJ}^F \]  

(B.9)

\[ \rho = \frac{\rho_o}{1 + F_B (V_{CJ} \rho_o - 1)} \]  

(B.10)

\[ E = Q_x + (E_{CJ} - Q_x)F_B \]  

(B.11)

This energy calculation appears adequate, although it is not justified analytically.

The third call to EXPLODE is made in HSTRESS and only for a running detonation. The purpose of the call is to compute pressure and energy during the reaction process; the pressure of fully detonated material is treated by EQST. First, the time \( t_B \) to begin burning is computed.

\[ t_B = \frac{|X - X_D| - bAX}{D_X} \]  

(B.12)

The fraction detonated is then

\[ F_B = \frac{(t - t_B) D_X}{bAX} \]  

(B.13)

where \( t \) is the current problem time. Because of the absolute value sign in Eq. (B.12), the detonation can proceed in either direction from the initiation point. Given the detonated fraction \( F_B \), the pressure and energy are computed both from the usual polytropic gas relations and as fractions of the C-J values. The pressure and energy values for the cell are taken as the maxima from these two calculations.
Appendix C

DESCRIPTION OF INPUT

This appendix provides some sample input decks and supplements the input description provided in Section 5. The construction and use of a data bank is outlined; the bank can be a permanent or temporary file containing material properties or other data. A procedure for reading special data through a NAMELIST statement is given, and the method for entering variables to obtain historical listings of any array quantity at any cell is described. The meaning of the indicator NVAR is given to aid in incorporating new material models, in calculating with models having large numbers of variables, or in getting data from large models.

Data Banks

A data bank for PUFF is a file containing some portion of the input for a problem. Specifically, the data may be card images representing the general running information of Section 5.1, a complete set of properties for a material, x-ray radiation absorption coefficients for a material, an x-ray spectrum, or a depth-dose profile for a radiation problem. Sample data banks are shown in Figure C.1 and C.2. After describing the banks, we outline their use in setting up problems.

The two banks in Figures C.1 and C.2 were constructed by inserting them like data decks for reading by GENRAT. The first line of the data must read

_DATA or _ABS DATA

where either word starts in column 2. On reading the word "DATA", GENRAT places the next card images up to an end-of-file (the 7/8/9 card) on Tape 4, whereas "ABS DATA" indicates a write to Tape 2. Material properties data, general running information, spectral data, and depth
FIGURE C.1  DATA BANK CONTAINING GENERAL RUNNING INFORMATION, MATERIAL PROPERTY DATA, AND A SPECTRUM (ON TAPE 4)
<table>
<thead>
<tr>
<th>ELEMENT</th>
<th>X-RAY ABS NOE</th>
<th>1 ATWT</th>
<th>2 ATWT</th>
<th>3 ATWT</th>
<th>4 ATWT</th>
<th>5 ATWT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hydrogen</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge1</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coef1</td>
<td>2.44950E 0</td>
<td>0.334932E 0</td>
<td>0.472054E 0</td>
<td>2.71059E 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Liumium</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge1</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coef1</td>
<td>6.06498E 0</td>
<td>0.329055E 0</td>
<td>0.107282E 0</td>
<td>1.14450E 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chromium</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge1</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coef1</td>
<td>9.04503E 0</td>
<td>0.283490E 0</td>
<td>0.209900E 0</td>
<td>2.29488E 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Iron</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge1</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coef1</td>
<td>1.31704E 0</td>
<td>1.253681E 0</td>
<td>3.07696E 0</td>
<td>4.13025E 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vanadium</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge1</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coef1</td>
<td>1.43340E 0</td>
<td>1.264316E 0</td>
<td>3.31403E 0</td>
<td>4.96500E 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nickel</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge1</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coef1</td>
<td>1.34235E 0</td>
<td>1.251606E 0</td>
<td>3.12129E 0</td>
<td>4.87182E 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Copper</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge1</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coef1</td>
<td>1.47599E 0</td>
<td>1.218849E 0</td>
<td>3.21403E 0</td>
<td>5.0120E 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Zinc</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge1</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coef1</td>
<td>1.43235E 0</td>
<td>1.251606E 0</td>
<td>3.12129E 0</td>
<td>4.87182E 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cobalt</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge1</td>
<td>1.00000E 0</td>
<td>1.000E</td>
<td>1.000E</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coef2</td>
<td>1.48015E 0</td>
<td>1.218849E 0</td>
<td>3.21403E 0</td>
<td>5.0120E 0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIGURE C.2 DATA BANK CONTAINING X-RAY ABSORPTION DATA FOR SEVERAL ELEMENTS (ON TAPE 2)**
dose profiles are all contained on Tape 4. Only x-ray absorption data are on Tape 2. The two banks are used so that GENRAT can read properties from Tape 4 and then be referred to Tape 2 to pick up absorption characteristics without losing its position in Tape 4.

The data banks in Figures C.1 and C.2 contain a series of separate data groups. Each group is constructed strictly in accordance with the requirements of GENRAT. However, the groups themselves may be in any order and may be spaced by blanks or comment cards to annotate the bank.

The data banks may be constructed by placing card images on a file before the PUFF calculation or by letting GENRAT write the file during the calculation as in the preceding examples. We have stored large data banks on an UPDATE file and written the data bank from UPDATE as a COMPILE file. Alternatively, the bank may be written by copying cards to the appropriate file using control cards.

The banks are used in the following way. The data deck is constructed in the normal way except that the information in the bank is omitted. Instead, some indicator is provided to show where the data should be found. Figure C.3 shows a data deck for an impact with a hot aluminum target. The "X" in the first column of the IDENT card shows that the remainder of the general running data should come from the data bank and that the NAMELIST routine EXTRA should be called. The letters "SHOCKEY" in columns 72-80 give the title of the set of general running data to be used. These letters correspond to those in column 12-20 following "HEADING" in Figure C.1. After GENRAT reads the general running information, it reads the "EXTRA" card and calls EXTRA to read the "$NLIST..." line. The end-of-file (7/8/9) stops the reading in EXTRA. The "T" in the first column of the material cards for 1145 aluminum show that the properties for these materials must come from the data bank.

In GENRAT the input deck of Figure C.3 is used to construct a complete data deck. The GENRAT output for this case is in Figure C.4. The indicators to the right of the card images help to show the process. The variable IND is the indicator in the first column. IN is the file from which the line is taken: IN = 5 shows the standard input file,
**FIGURE C.3** INPUT DECK FOR HOT ALUMINUM IMPACT CALCULATION

**FIGURE C.4** GENRAT OUTPUT FOR INPUT DECK OF FIGURE C.3
whereas $IN = 4$ indicates Tape 4, the data bank. Hence in this case the first line is from the data deck. Next the data bank is searched (by the subroutine REDR) for a label HEADING SHOCKEY. Then GENRAT reads the next four lines from the data bank. Control then returns to the data bank and the subroutine EXTRA is called to read and print the NAMELIST data which reinitializes the flyer velocity to $1.46 \times 10^4$ cm/sec. Then GENRAT reads the line T1145-AL, which causes REDR to find the appropriate line in the data bank again. GENRAT repeats the reading of the material name card and then reads the remaining properties from the data bank. After the two sets of aluminum data, control returns to the data deck for reading the cell layout and the second NAMELIST record.

A second example of the use of data banks is shown in Figures C.5 and C.6. The data deck in Figure C.5(a) describes a radiation problem using a spectrum labeled SPEC_Z12, which deposits energy into three materials: asbestos phenolic (AP), fused silica, and quartz. The "T" in the first column indicates which data are taken from Tape 4. The material property data for AP are also shown as part of a data bank in Figure C.5(b). The completed input deck constructed by GENRAT and exhibited in Figure C.6 shows the source for each line in the column on the right labeled "IN". $IN = 5$ is the normal input file shown in Figure C.5(a); $IN = 4$ means Tape 4 and $IN = 2$ means Tape 2. The line headed "TAP" in the deck in Figure C.5(a) brings in the properties from Tape 4 shown in the data bank of Figure C.5(a). Included in these properties are the chemical constituent data needed for the x-ray absorption calculation and read in the subroutine DEPOS. The constituent data names the chemical species (e.g., IRON), the source of the absorption data (e.g., ITAPE = 2), and the weight fraction (PBW). The IRON is located by REDR on Tape 2, and DEPOS reads the number of edges (NOE), atomic weight (ATWT), the EDGES, and the COEFS ($A_0$, $A_1$, $A_2$, and $A_3$ referred to in Appendix A). The same process is repeated for silica and quartz, except that PBW is interpreted as the number of atoms of the constituent in the molecule, instead of the weight fraction (because $PBW > 1.0$).
TIDENT AP EXPERIMENT H 3116 X-RAY DEPOSITION INTO A HEAT SHIELD APDS

TAP (ERLICH)

T FUSED SILICA-BARKER

T QUARTZ (GRAHAM)

(a) SAMPLE DATA DECK FOR RADIATION PROBLEM

(b) SAMPLE PROPERTY DATA FOR AP IN DATA BANK ON TAPE 4

FIGURE C.5 DATA DECK AND DATA BANK, ILLUSTRATING USE OF DATA BANKS FOR RADIATION PROBLEMS WITH MULTICONSTITUENT MATERIALS
FIGURE C.6  INPUT CONSTRUCTED BY GENRAT FOR RADIATION PROBLEM IN WHICH GENERAL RUNNING INFORMATION, MATERIAL PROPERTIES, RADIATION ABSORPTION DATA, AND SPECTRUM ARE ON DATA BANKS
<table>
<thead>
<tr>
<th>TABLE 1.1</th>
<th>SPECIFICATION</th>
<th>VALUE 1</th>
<th>VALUE 2</th>
<th>VALUE 3</th>
<th>VALUE 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>X</td>
<td>Y</td>
<td>Z</td>
<td>W</td>
<td>T</td>
</tr>
<tr>
<td>OTHER</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIGURE C.6** Input constructed by GENRAT for radiation problem in which general running information, material properties, radiation absorption data, and spectrum are on data banks (Concluded)
Occasionally it is necessary to insert additional information for which there is no standard reading procedure. In that case the EXTRA routine is called to read the information through a NAMELIST READ statement, or HDATA is called to read variables into the H array. The use of EXTRA is considered first.

In the special NAMELIST READ statement in EXTRA, the variable and its value are given. The sample shown at the end of Figure C.4 is 

$NLIST \ EHL(12) = 20*4.46E9, RHOS(2) = 2.784$. The dollar sign in column 2 and at the end delimit the information and also signal a NAMELIST READ. The list of variables used in EXTRA is called NLIST: it includes most of the material properties, the main cell arrays, indicators, and other variables for which a change might be required. The effect of the READ statement mentioned above is to initialize 20 values of EHL, the internal energy, beginning at EHL(12) with a value of $4.46 \times 10^9$ and then reset the initial density, RHOS, of the second material to 2.784. This case illustrates two uses of EXTRA: the EHL array is being initialized to represent a preheating of the target, and RHOS is being reset. RHOS was initialized to 2.705 in the normal way with the material property data. That value is appropriate for preheated and expanded aluminum and is needed for giving the cells the correct initial mass and density. However, for the equation-of-state calculations, the standard density of 2.784 is required; this resetting is accomplished after the layout by means of the EXTRA routine as shown.

Pressure boundary information may be inserted through the NAMELIST READ as shown in Figure C.5. The parameters P6 and T6 define a pressure history with the form

$$P = P6 \ exp \ (t/T6)$$

Subscripts (1) for P6 and T6 indicate the first boundary, whereas (2) indicates the final boundary.
CALLS to EXTRA may occur at two points in GENRAT: immediately following the general running information and at the end of the deck. As mentioned above, the first of these CALLS is triggered by an "X" in the first column of the IDENT line. The second CALL is caused by a line with the letters "_EXTRA" preceding the lines containing the NAMELIST data.

The NAMELIST statement does not permit the use of alpha or octal data. Therefore, to initialize the H parameter array, it was necessary to construct a special reading subroutine, HDATA. HDATA is called only at the end of the data deck. If both EXTRA and HDATA are used, HDATA must precede. The data line for HDATA is preceded by a line containing the label "_H-DATA". HDATA reads only 1 or 2 H values for each call, but multiple calls are possible by providing additional "_H-DATA" and data lines. The data are in a single line containing, J, I, and K for the equation H(J,I) = K, and K is read in an R5 format. In our work the only H values reset with HDATA have been at first or last coordinates to change boundary conditions; therefore, not more than two values were required.

Input Description for Historical Prints

Historical listings can be obtained for any variable in the cell or coordinate arrays and for several other variables. This section describes the input data required to obtain the histories, and the subroutines used.

Input Directives. Each input directive for a historical listing consists of two groups of symbols: one part is for the type of data and one part is for the location in the material. The directives are provided in free-field format in columns 11 to 80 of a data line. Samples of these directives are

\[ S1,26 \quad D,18 \quad COM1,3.25 \]

In each of these three pairs, the characters before the comma are a directive group that designates a data type: S1 is thermodynamic stress in the direction of propagation, D is density, and COM1 is the first variable assigned to the COM array, a large array available for use with constitutive relations that require extra storage. All these type
designators are defined in Table C.1. The number after the comma are a directive group that designates a location within the material. For example, 26 and 18 are cell numbers. The decimal 3.25 means layer 3, 25% of the distance from the front of the layer. The groups of characters forming a directive group are separated by either commas or blanks.

More samples of the directives are given in Table C.2. The first 10 spaces of each line may be used to identify the line or may be left blank. The next 70 characters contain the designators that are processed to determine which stress histories are required. Table C.2 shows several sets of directive groups. Each set begins with one or more type designator groups (beginning with a letter) and ends with one or more numerical groups. A set constitutes a request for histories of all the types given by type designators at each of the locations in the numerical groups.

The first line of the table contains five numbers that constitute a set requesting stress histories in the direction of propagation at those cell locations. In this case a type designator was omitted: S1 is assumed to be the type if the first character on the first line is an integer. The next type designator is D for density, followed by three cell locations for which density histories are required. On the second line is a large set containing five type designators: S1, S2, S3, E, and Y. Hence first, second, and third principal stresses, internal energy, and yield strength are requested at cells 6, 7, and 8. Next the coordinate position X is requested at coordinate points 6 and 9. The third line shows a request similar to that on the second line, except that the second request set (C, U, SD1, 24, 30, 35) is continued on the fourth line. The fourth line also contains a set requesting histories of the 24th variable in the COM array for cells 5, 10, 15, and 20.

In addition to the requested histories is a group of histories that are automatically obtained. The time increment (DTNH), the calculation time for each time step (DELTIM), and the cell controlling the time step (JTS) are always given. In a multilayer problem, interface stress
Table C.1
DEFINITIONS OF DIRECTIVE GROUPS

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>Sound speed, cm/sec</td>
</tr>
<tr>
<td>COM,</td>
<td>An array containing special variables used by constitutive relations that require more than the standard arrays. A number immediately following COM indicates the particular one of these special variables requested.</td>
</tr>
<tr>
<td>COM2,</td>
<td></td>
</tr>
<tr>
<td>COM12</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>Density, g/cm$^3$</td>
</tr>
<tr>
<td>DPDD</td>
<td>$\partial P/\partial \rho$, dyn/cm$^2$/(g/cm$^3$)</td>
</tr>
<tr>
<td>DPDE</td>
<td>$\partial P/\partial E$, dyn/cm$^2$/(erg/g)</td>
</tr>
<tr>
<td>E</td>
<td>Internal energy, erg/g</td>
</tr>
<tr>
<td>H1</td>
<td>$H(J,1)$, cell state indicator</td>
</tr>
<tr>
<td>H2</td>
<td>$H(J,2)$, cell or coordinate type indicator</td>
</tr>
<tr>
<td>H3</td>
<td>$H(J,3)$, cell state indicator</td>
</tr>
<tr>
<td>IMP</td>
<td>Impulse = $\int R dt$, dyn·sec/cm$^2$</td>
</tr>
<tr>
<td>NEM,</td>
<td>Special arrays; meaning depends on the material model</td>
</tr>
<tr>
<td>NET</td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>Pressure, dyn/cm$^2$</td>
</tr>
<tr>
<td>R</td>
<td>Mechanical stress in direction of propagation, dyn/cm$^2$</td>
</tr>
<tr>
<td>SDT</td>
<td>Deviator stress in the circumferential direction in cylindrical problems, dyn/cm$^2$</td>
</tr>
<tr>
<td>SD1,</td>
<td>Deviator stresses in the direction of propagation, and in two orthogonal directions. For cylindrical geometry, the second direction is circumferential and the third is axial, dyn/cm$^2$</td>
</tr>
<tr>
<td>SD2,</td>
<td></td>
</tr>
<tr>
<td>SD3</td>
<td></td>
</tr>
<tr>
<td>S1,</td>
<td>Principal stress in the direction of propagation and in two orthogonal directions. For cylindrical geometry, the second direction is circumferential and the third is axial, dyn/cm$^2$</td>
</tr>
<tr>
<td>S2,</td>
<td></td>
</tr>
<tr>
<td>S3</td>
<td></td>
</tr>
<tr>
<td>S-INT</td>
<td>Interface stress—average of stresses in cells on either side of interface, dyn/cm$^2$</td>
</tr>
</tbody>
</table>
Table C.1 (concluded)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>Spall strength, dyn/cm²</td>
</tr>
<tr>
<td>U</td>
<td>Coordinate velocity, cm/sec</td>
</tr>
<tr>
<td>V</td>
<td>Specific volume, cm³/g</td>
</tr>
<tr>
<td>X</td>
<td>Coordinate location, cm</td>
</tr>
<tr>
<td>XO</td>
<td>Initial coordinate location, cm</td>
</tr>
<tr>
<td>Y</td>
<td>Yield strength, dyn/cm²</td>
</tr>
<tr>
<td>Z</td>
<td>Cell mass, g/cm², g/cm, or g for planar, cylindrical and spherical geometries, respectively</td>
</tr>
<tr>
<td>1,2,any</td>
<td>Cell or coordinate number</td>
</tr>
<tr>
<td>3.25</td>
<td>Location designator. Integer before the decimal indicates the layer number (not counting void layers). The following number, including the decimal, is the fractional distance into the layer</td>
</tr>
</tbody>
</table>
Table C.2

SAMPLE INPUT DIRECTIVES

<table>
<thead>
<tr>
<th>1*</th>
<th>10</th>
<th>11</th>
<th>80</th>
</tr>
</thead>
<tbody>
<tr>
<td>JEDIT =</td>
<td>16</td>
<td>23</td>
<td>4</td>
</tr>
<tr>
<td>JEDIT 2</td>
<td>S1,S2,S3,E,Y,</td>
<td>6,7,8</td>
<td>X,6,9</td>
</tr>
<tr>
<td>HIST 3</td>
<td>U, H2,NEM,16</td>
<td>C,U, SD1</td>
<td>24</td>
</tr>
<tr>
<td>4th CARD</td>
<td>30,35,COM24</td>
<td>5</td>
<td>10</td>
</tr>
</tbody>
</table>

*Column numbers on an input card; first column should be left blank.
histories are listed between each layer. With the current dimensions, a total of 100 histories may be printed.

Subroutine Description. Three subroutines, PRESCR, STORR and Scribe, process the input directives, store the required cell information during the wave propagation calculation, and print the histories at the end of the calculation. Here only an outline of the procedure is given.

During the initialization stage of a computation, the input directives are read by GENRAT. At the end of GENRAT, PRESCR (meaning PRE-SCRIBE) is called. PRESCR examines the input directives character by character and constructs three arrays: JTYP, JEDIT, and JNUM. JTYP contains the title of the history, including the data type and cell location. JEDIT is the j value of the cell, and JNUM is the location of the specific variable in the coordinate arrays.

At each time step during a wave propagation calculation, STORR is called to store all the requested variable values from that time step. The JNUM and JEDIT arrays are used to select the correct values for storage. Temporarily these values are stored in the A array. When part of the A array is filled, the values are buffered out to a disk file (called Tape 3) while the second part of the array is being filled. When the second part is full, storage begins again in the first part and the second part is buffered out. This process is repeated throughout the calculation.

At the end of the wave propagation calculation, STORR is called to complete buffering of information to the disk file. Then Scribe is called to print the histories. Scribe reads the disk file and prints 10 histories at a time. When one set of histories is complete, Scribe rewinds and rereads the file and prints another set until all the histories have been listed.
Additional Variables for Material Models: COM, LVAR, NVAR

An array of additional variables is provided for use with material models requiring more variables per cell than normally available. These extra variables are in the COM array. This section describes the use of the array, when it is needed, how to use it when adding new material models, and how to obtain historical listings of values in the array.

The usual variables available at each cell for each material model are those in the COMMON labeled COORD. Included are the yield array YHL, the quantities NEM and NET, and an indicator H(J,I). For material models where these variables are insufficient, the COM array is provided. So far, the following subroutines have required this extra storage: BFRACT2 (11 variables), BFRACT3 (20, HYPO (3), PEST (5), REBAR (7), and SHEAR2 (indefinite number). The number required for SHEAR2 is $4 + \text{NANG} + 2 \sum \text{NSIZE}$, where NANG and NSIZE are input data for SHEAR2.

Locations within COM are assigned with the aid of a second array LVAR(J). LVAR(J) is the location in COM at which the storage for the jth cell begins. Then, for example, the fifth value in COM for the jth cell is COM(L+4) where L = LVAR(J). NVAR(M) (an input quantity) is the number of additional variables assigned to each cell. The location quantities LVAR may be assigned during the initialization of the problem or during the running. For the fracture routines BFRACT2 and BFRACT3, the assignment is made for the jth cell during the computation at the time fracture begins at that cell. Hence, if the cell never undergoes fracture, it does not require the added storage.

The COM array is especially convenient for providing variables to new models because the formal parameters of the model subroutine may be either scalars or arrays. For example, BFRACT2 has the formal parameters FU2D, CL, and CN, where FU2D is a scalar and CL and CN are each arrays of five quantities. In the CALL statement these same parameters are listed as COM(L), COM(L+1), and COM(L+6).
Historical listings can be obtained of all array quantities, including COM array quantities. The form of the request for the listing is "COM2 or "COM11". In the sample of the preceding paragraph, COM2 would indicate the second value in the COM array for the particular cell, and that corresponds to CL(1) in the CALL to BFRACT2. Similarly, COM11 refers to CN(5). Usually the CALL statements in HSTRESS must be compared with the formal parameters of the material model to relate the COM quantities to the variables of interest.

Sample Data Decks

A number of sample data decks are provided to illustrate the main features of PUFF and the range of problems that can be treated. General guidelines for constructing the decks are listed below.

- The data fields are usually in multiples of 5 or 10 characters.
- The first column is reserved for indicators.
- Columns 2 through 10 are usually labels only.
- Any number of decks can be run, one following the next with only an end-of-file (7/8/9) between decks.

These features are illustrated in the following sample decks.

The data decks are grouped according to problem type, but each also illustrates many other features. Figures C.7 through C.10 (and Figure C.3) show impacts in planar geometry, Figures C.11 through C.13 are for cylindrical geometry, and Figure C.14 is for spherical geometry. Explosives are featured in Figures C.15 and C.16 and radiation in Figures C.17 through C.20 (and Figure C.5). A pressure boundary provides the loadings in Figures C.21 and C.22.

The JEDITS are listed in several ways. Many are integers without TYPE designation, indicating that only $\sigma_1$ is required. In Figure C.11, all three principal stresses and COM(3) are required at positions given by decimals such as 2.1 (2.1 means a location in layer 2, 0.1 times the thickness through the layer).
FIGURE C.7 INPUT DECK FOR IMPACT IN 1145 ALUMINUM, ILLUSTRATING DUCTILE FRACTURE DATA AND JEDITS WITH NO TYPE INDICATOR
IDENT = S25 E SHOT 8b78-1-S25 IN ARMCO IRON

C   STANDARD IMPACT SIMULATION USED TO CALIBRATE OR CHECK BRITTLE FRACTURE
C   MODELS

NTEIDT = 0  NJEDIT= 2  NREZON= 0
JEDIT = 27  28  29  30  31  32  33  34  38  39  16  18  20  22
        23  24  25  26
NEDIT = 10  JCYCS = 150  CKS = 3.0  TS = 1.00E-05
VMTHLS = 2  MATFL = 1  UZERO = 1.960E+04

ARMCO IRON
RHOS = 7.85  CFP = 020  DPY = 001
EQST = 1.589E+12  5.170E+12  7.360E+10  1.690E+00  2.500E-C1
TSR1 = -5.500E-04-1  .000E+08  5.000E-05  4.000E+12-3.000E+09-5.270E+09
TSR2 = 0.  0.  2.500E-01  5.000E-01  4.000E-01  3.000E+00
YO = 2.000E+09  8.190E+11

PMMA-BKB (BARKER)
RHOS = 1.184  CFP = 000  DPY = 001
EQST = 7.000E+10  4.050E+11  1.000E+10  1.000E+06  2.500E+01  3.640E+11
YIELD = 1.000E+06  1.950E+10  2.350E+09

NLAYER = 3  JMAT = 1  I  1  2
NZONES= 1  10 CELLS IN 1.133E-01 CM
NZONES= 1  25 CELLS IN 3.156E-01 CM
NZONES= 1  22 CELLS IN 4.800E-01 CM  DX = 1.250E-02  RATIO = 1.05

7/8/9

FIGURE C.8 INPUT DECK FOR IMPACT IN ARMCO IRON, ILLUSTRATING BRITTLE FRACTURE AND A GEOMETRIC CELL LAYOUT
FIGURE C.9 INPUT DECK FOR ASYMMETRIC IMPACT OF TONALITE, SHOWING COULOMB FRICTION WITH \( \tan \phi = 0.056 \), MULTIPLE ZONES IN LAYER, AND USE OF THE POREQST MODEL
IDENT PROJECTION IMPACT ON CONCRETE AT 22.34M/SEC - TEST REBAR
NEDIT = 0 NJEDIT = 1
JEDIT = 5 8 11 14 16 18 20 22 24 26 27 29
INDEX = 10 JCYCS = 150 CKS = 1.000E+02 TS = 1.000E-04
NMRLS= 4 MATFL = 1 UZERO = 2.234E+03
IMPACTOR STEEL PHOS = 7.85E0 CFP = 0.00DPY = 01
EQST = 1.588E12 5.17E12 7.36E10 1.69E0 0.25E0 5.17E13
YIELD = 1.222E10 8.188E11
REBAR STEEL PHOS = 7.85E0 CFP = 0.00DPY = 01
EQST = 1.588E12 5.17E12 7.36E10 1.69E0 0.25E0 5.17E13
YIELD = 1.030E10 8.188E11
CONCRETE PHOS = 2.85E0 CFP = 0.004 UPY = 00
EQST = 2.83E0+11 0 1.000E+11 2.000E+00 .25 0
RHO = 2.02E0 AMU = 2.033E+11
AK = 7.005E+10 AK2 = -0.550E+02 MUP = 5.250E+10 MUP2 = 0.125E+03
MC = 1.040E+09-8.300E+08 2.70E+0 2.50E+08 1.555
SCTIV[m] = 2.30E+03 DAMG = 0.100E-02
EV = 0.
-1.20E-02-3.500E-02-5.000E-02-2.230E-01
NREG = 4 NPRCAP = 0 P1 = -3.50E+03 W2 = 1.25E0
P2 = -1.000E+09 DELP = C
P2 = -2.50E0+09 DELP = 0
P2 = -3.400E+09 DELP = 0
P2 = -3.533E+10 DELP = 0
REBAR PHOS = 2.5015E0 CFP = 0.00DPY = 000 VAR = 7
FS = 0.05EO THEFT = 5 .01INC = 31 MS = 2
NAYER = 6 JMAT = 1 3 4 3 4 3
NZONES = 1 14 CELLS IN 10.16E0 CM
NZONES = 1 1 CELLS IN 0.508E CM
NZONES = 1 1 CELLS IN 0.508E CM
NZONES = 1 6 CELLS IN 3.048E CM
NZONES = 1 1 CELLS IN 0.508E CM
NZONES = 1 1 CELLS IN 0.508E CM
7/8/9

FIGURE C.10 INPUT DECK FOR IMPACT OF A STEEL PLATE ONTO REINFORCED CONCRETE, SHOWING THE USE OF CAP AND REBAR SUBROUTINES AND MULTIPLE LAYERS OF A SINGLE MATERIAL
IDENT FR 5 FRAG ROUND OF ARMCO IRON TO SIMULATE CROWE'S TESTS 3 AND 4
C. THE COMP B EXPLOSIVE IS TREATED BY A SIMULTANEOUS DETONATION

NTEOT = 0 NJEOT = 1 NREZON = 0 NALPHA = 2
NJEDIS = 5, 2, 1, 2, 3, 4, 5, 6, 7, 8, 9
NEDIS = 10 JCYS = 150 CKS = 3.000E+01 TS = 8.0005005
NMTRLS = 2 MATFL = 1 UZERO = 0.

COMPB

RHOS = 1.72 CFP = 000 DPY = 012
EQST = 1.0 0.0 1.0 1.0 1.0 1.0
TENS = -1.000E+09 0.0 -1.0
MELT = -1.0

ARMCO SHEET

RHOS = 7.85E0 CFP = 030 DPY = 002 NVAR = 58 NCON = 0
EQRST = 1.589E+12 5.170E+12 7.360E+10 1.69E0 0.25E0 5.170E+12
SH2 = 0.000E+01 -2000E+00 1.100E-02 0.000E-04 0.17 0.070E+00 0.070E+00 0.070E+00
0.070E+00 1.4 3.000E-08 3.000E+08 6.0 2.0 1.7 7.000E+09
NSIZE = 0 0 0 0 8 8 8 0 0 0
MELT = 1.085E+10 6.460E+09 1.500E-01 2.500E-01 6.000E-02
YIELD = 2.000E+09 8.190E+11

NLAYERS = 2 JMATH = 1 2
NZONES = 1 10 CELLS IN 1.5 INCH
NZONES = 1 10 CELLS IN 0.75 INCH

FIGURE C.11 INPUT DECK FOR THE CYLINDRICAL CALCULATION OF A FRAGMENTING ROUN, SHOWING DETONATION OF AN EXPLOSIVE, SHEAR BAND MODEL, AND ENGLISH UNITS IN THE LAYOUT
IDENT FR 4 FRAGMTG RND OF HF-1 TO SIMULATE CROWES TESTS 1 AND 2.
NTEOT = 0 NJEDIT = 3 NALPHA = 2
JEDIT = S1 2 1 2 2 3 2 4 2 5 2 6 2 7 2 8 2 9
JEDIT = COM2 COM3 COM4 12 13 14 15 16 17 18 19
JEDIT 3 S2 S3 U0 17 18 19
NEDIT = 10 JCYCS = 300 CKS = 30* TS = 8.000E-05
NMTRLS = 2 MATFL = 1 UZERO = 0.

COMP0
RHOS = 1.72 CFP = 000 DPY = 012
EQST = 1.0 0*
QEXPL = 4.469E+10
TENS = -1.000E+09 0*
MELT = -1.

HF-1
RHOS = 7.85E0 CFP = 030 DPY = 002 NVAR = 58 NCON = 0
EQST = 1.589E+12 5.170E+12 7.360E+10 1.69E0 0.25E0 5.170E+13
SH 2 3.000E+01 2E0 1.100E-02 1.000E-04 1.700E-01 7.000E-02 7.000E-02
1.4 3.000E-08 3.000E+08 6* 2* 1 17 7.000E+09
NSIZE = 0 0 0 0 8 8 8 0 0
YIELD = 1.030E+10 8.190E+11
MELT = 1.085E+10 1.000E+08 1.500E-01 2.500E-01 6.000E-02
NLAYERS = 2 JMAT = 1 2
NZONES = 1 10 CELLS IN 1.5E0 INCH
NZONES = 1 10 CELLS IN 8.750E-01 INCH

FIGURE C.12 INPUT DECK FOR THE CYLINDRICAL CALCULATION OF A FRAGMENTING ROUND OF HF1 STEEL, SHOWING DETONATION OF AN EXPLOSIVE
CYLINDRICAL PUFF CALC OF CONCRETE/GROUT FOR KEDUGH ON 6172-10
NTEDIT = 0 NJEDT = 1 NREZUN = 0 NALPHA = 2
JEDIT
JEDIT
JEDIT
JEDIT
JEDIT
JEDIT
S = 1.1.1.1.5.1.75.1.2.2.2.3.2.4.2.5.2.6.2.8 S=INT.0

NEDIT = 10 JCYCS = 100 CKS = 100. TS = 4.000E-04
NMTRLS = 2 MATFL = 2 UZERU = 0.

CONCRETE
RHOS = 2.630E+00 CFP = 001 DPY = 001
EQST = 2.830E+11 0. 1.000E+11 2. 0.25 0.
RHO = 2.22
AK = 7.000E+10 MUP = 5.250E+10 YO = 2.420E+08
NREG = 4
RHOP = 2.22 2.247 2.299 2.334 2.775 2.8
P1 = 3.500E+08
P2 = 1.000E+09 DELP = 0. YADD = 2.970E+08
P2 = 2.400E+09 DELP = 0. YADD = 2.660E+08
P2 = 3.400E+09 DELP = 0. YADD = 1.220E+08
P2 = 1.533E+10 DELP = 0. YADD = 2.630E+08
YIELD = 0. 5.250E+10 0. 0.

GROUT (DSRM-2)
RHOS = 2.2204 CFP = 001 DPY = 001
EQST = 8.839E+10 0. 2.000E+11 2. 0.25 0.
RHO = 2.0668
AK = 8.035E+00 MUP = 3.887E+10 YO = 1.000E+08
NREG = 3
RHOP = 2.0668 2.142 2.245 2.353 2.353 2.4
P1 = 1.000E+08
P2 = 3.000E+08 DELP = -2.000E+07
P2 = 1.200E+09 DELP = -1.200E+08
P2 = 4.000E+09 DELP = -2.400E+08
YIELD = 1.000E+08 3.887E+10

NLAYERS = 3 JMAT = 0 1 2
NZONES = 1 0 CELLS IN 45. CM
NZONES = 1 5 CELLS IN 10. CM
NZONES = 1 25 CELLS IN 50. CM

FUNCTION SIGMAT(LST)
DIMENSION PS(10), TS(10)
DATA PS/0.0,4.E9,4.E9,7*0./
DATA TS/0.0,6.E-5,5.0E-4,1.E-3,6*0./
DATA NM/4/
N=1
SIGMAT=0.
20
N=N+1
IF (N .GT. NM) RETURN
IF (T .GT. TS(N)) GO TO 20
SIGMAT = PS(N-1)+(PS(N)-PS(N-1))/(TS(N)-TS(N-1))*(T-TS(N-1))
RETURN
END

FIGURE C.13 SIGMAT AND INPUT DECK FOR A CYLINDRICAL CALCULATION, ILLUSTRATING A HOLLOW OR EMPTY FIRST LAYER, PRESSURE BOUNDARY, AND USE OF POREQST
FIGURE C.14  INPUT DECK FOR SPHERICAL EXPLOSION OF PETN IN ROCK MATCHING GROUT, SHOWING USE OF A TABULAR EQUATION OF STATE, CAP MODEL, AND A NAMELIST STATEMENT
FIGURE C.15  INPUT DECK FOR EXPLOSIVELY THROWN FLYER PLATE IMPACTING OTWR AND ILLUSTRATING THE USE OF EXPLOSIVE, NAMELIST, COMMENTS, AND GAPS IN THE LAYERS
IDENT = 310 I RUNNING DETONATION
C TREAT A RUNNING DETONATION THROUGH HMX, CDMP B, AND TNT TO STUDY THE
C EFFECT OF OVERDRIVING OF A LOW C-J EXPLOSIVE BY A HIGH C-J EXPLOSIVE.
NTEDT = 0 NJEDIT = 2
JEDIT = 1 16 31 46 61 77 80 84 93 102 105 109 118 127
130 134 137 140
NEDIT = 10 JCYCS = 200 CKS = 30.0 T5 = 1.000E-04
NMATLS = 4 MATFL = 1 UZERO = 0.

HMX
RHO S = 1.84 CFP = 0.0 DPY = 012
EQST = 1.0 0.1.0 1.0 1.0 1.89 1.89 0.
QEXPL = 5.690E+10 0.0625 2.0
TENS = -1.0 0.0 0.0 0.0 0.0 0.0
MELT = -1.0 0.0 0.0 0.0 0.0 0.0

CDMP B
RHO S = 1.68E0 CFP = 0.0 DPY = 012
EQST = 1.0 0.1.0 1.0 1.63E0 1.63E0 0.
QEXPL = 5.190E+10 0.0625E0 2.0E0
TENS = -1.0 0.0 0.0 0.0 0.0 0.0
MELT = -1.0 0.0 0.0 0.0 0.0 0.0

TNT
RHO S = 1.56E0 CFP = 0.0 DPY = 012
EQST = 1.0 0.1.0 1.0 1.44E0 1.44E0 0.
QEXPL = 4.520E+10 0.0625E0 2.0
TENS = -1.0 0.0 0.0 0.0 0.0 0.0
MELT = -1.0 0.0 0.0 0.0 0.0 0.0

AL6061-T6
RHO S = 2.707E0 CFP = 0.0 DPY = 001
EQST = 6.670E+11 1.000E+12 1.220E+11 2.04E0 0.25E0 0.
YIELD = 3.210E+09 2.670E+11 3.790E+10

NLAYER = 4 JMAT = 1 2 3 4
NZONES = 1 1 1 1
NZONES = 4 24 24 50 CELLS IN 10.0 3.0 6.0 CM

FIGURE C.16 INPUT DECK FOR A RUNNING DETONATION THROUGH THREE EXPLOSIVES,
ILLUSTRATING THAT PUFF PERMITS OVERDRIVING OF EXPLOSION
IDENT = DCS5-A-1 SINTERED AL2O3 WITH BE AT FRONT
C AUTOMATIC REZONING EVERY 30 CYCLES
C X-RAY DEPOSITION HAS BEEN CALCULATED BY -FSCATT- AND PROVIDED AS A
C DEPTH-DOSE PROFILE FOR THE PUFF CALCULATION.
NEDIT = 1 NJEDIT = 1 NREZONE = -30
TEDIT = 5.000E-09 JEDIT = 5 17 26 40 48 61 66 68 70 72 74 76 78 80
DTMAX = 5.000E-09 TREZON = 5.000E-09
NEGIT = 30 JCYCS = 200 CKS = 2.0 TS = 1.500E-06
NTRLS = 3 MATFL = 0 UZERO = 0.

BERYLLIUM RHOS = 1.050E0 CFP = 000 DPY = 001
EQST = 1.203E+12 1.524E+12 3.550E+11 1.450E0 2.500E-01 5.310E+11 0.
MELT = 3.955E+10 1.978E+10 1.600E-01 1.600E-08 1.200E-09 4.500E-01

ALUMINA SINTERED RHOS = 3.969 CFP = 000 DPY = 002
EQST = 2.655E+12 4.200E+12 3.080E+11 1.320E+00 2.500E-01 2.090E+11
RHO = 3.160 E0 AK = 1.700E+12 MUP = 1.000E+12 YO = 3.000E+09
NREG = 3 RHDP = 2.8 E0 C1 = 0.050
P1 = 3.000E+10 1P2 = 5.000E+10 DELP = 0.
2P2 = 1.400E+11 DELP = -2.500E+10 YADDP = 1.000E+10
3P2 = 3.350E+11 DELP = -2.200E+10 YADDP = 1.000E+10
YIELD = 6.600E+10 1.600E+12
TENS = -1.000E+09 -1.000E+09 1.000E+11
VISC = 2.0 .02

C-7 RHOS = 1.190 CFP = 000 DPY = 001
EQST = 7.816E+10 1.956E+11 8.000E+09 2.79 2.213E+11
EMELT = 6.000E+09 4.000E+09 .1 3.0 1.5
MLAYERS= 4 JMAT = 1 2 3 3
NZONES= 1 31 CELLS IN 1 CM DX = 0.0076 RATIO = 9.35
NZONES= 1 31 CELLS IN 1 CM DX = 0.001 RATIO = 1.07
NZONES= 1 17 CELLS IN 15 CM DX = .01 RATIO = 1.07
NZONES= 1 30 CELLS IN .50 CM
NSPEC = 1 ANGLE = 0.
SPEC DC NARB ECAL = 200. START = 0. SSTOP = 3.000E-09

IDENT = DCS AL2O3 TH = NP= 7 (8E10.3)
0. 1.073E+01 9.333E-03 9.178E-01 1.667E-02 6.342E-01 2.500E-02 4.954E-01
5.000E-02 3.909E+01 7.500E-02 2.317E-01 1.000E-01 1.866E-01
IDENT = DCS AL2O3 TH = NP= 13 (8E10.3)
1.000E-01 6.638E+00 1.008E-01 5.319E+00 1.017E-01 4.480E+00 1.025E-01 3.895E+00
1.050E-01 2.871E+00 1.075E-01 2.334E+00 1.100E-01 2.002E+00 1.175E-01 1.473E+00
1.250E-01 1.208E+00 1.325E-01 1.041E+00 1.550E-01 7.652E-01 1.775E-01 6.230E-01
2.000E-01 5.411E-01
IDENT = DCS AL2O3 TH = NP= 7 (8E10.3)
2.000E-01 7.378E-02 2.250E-01 7.309E-02 2.500E-01 7.238E-02 2.750E-01 7.165E-02
3.000E-02 7.091E+00 2.500E-01 7.018E-02 3.500E-01 6.940E-02
IDENT = DCS AL2O3 TH = NP= 7 (8E10.3)
6.833E-01 5.861E-02 7.667E-01 5.572E-02 8.500E-01 5.270E-02

FIGURE C.17 INPUT DECK FOR RADIATION INTO BERYLLIUM AND ALUMINA, SHOWING THE USE OF A DEPTH-DOSE PROFILE, GEOMETRIC LAYOUT, AND AUTOMATIC REZONING.
FIGURE C.18  INPUT DECK FOR RADIATION INTO ALUMINUM, SHOWING A DEPTH-DOSE PROFILE AND MULTIPLE ZONES IN ONE LAYER
IDENT = 610 20C CAL RADIATION WITH SPECTRA IN 3 TIME INCREMENTS.
C THE RADIATION HAS BEEN SEPARATED INTO A SERIES OF BLACK BODY RADIATORS
C EACH WITH ITS OWN TEMPERATURE AND TIME OF DEPOSITION.

\[ NEDIT = 9 \quad JEDIT = 1 \quad NREZON = 4 \quad NALPHA = 1 \]
\[ TEDIT = 1.000E-08 1.000E-07 2.000E-07 4.000E-07 7.000E-07 1.000E-06 1.400E-06 \]
\[ JEDIT = 25 42 54 100 128 142 \]
\[ NTR = 2 3 4 5 \]
\[ JREZON = 45 60 60 110 \]
\[ NEDIT = 50 \quad JCYCS = 200 \quad CKS = 6.0E-06 \quad TS = 6.000E-06 \]
\[ NMTRLS = 1 \quad MATFL = 0 \quad UZERO = 0. \]

ALUMINUM
- \( RHOS = 2.785E0 \)
- \( CFP = 0.000 \)
- \( DPY = 0.004 \)
- \( NCUN = 1 \)

\[ EQST = 7.556E+11 \quad 1.290E+12 \quad 1.220E+11 \quad 2.40E0 \quad 2.500E-01 \quad 1.197E+12 \quad 3.110E+10 \]
\[ TENS = -2.500E+10 \quad 0. \quad -2.000E+10 \]
\[ CQSO = 3.24E0 \quad 2.500E-01 \quad 0. \]
\[ CO = 2.500E+09 \quad 2.870E+11 \]
\[ EMELT = 1.060E+10 \quad 2.500E-01 \quad 2.500E-01-6.000E-02 \]

ALUMINUM ITAPE = 5 PBW = 1.

ALUMINUM X-RAY ABS NDE = 2 ATWT = 26.98

\[ EDGE1 = 1.00000E 0 \quad 1.56000E 0 \]
\[ COEF1 = 1.08710E 1 \quad -2.78415E 0 \quad 1.58984E -1 \quad 0.00000E 0 \]
\[ COEF2 = 1.31739E 1 \quad -2.18214E 0 \quad 0.58940E -1 \quad 2.22834E -2 \]

NLAYERS = 1
ZONES = 3
- CELLS IN 40 CM DX = 1.224E-02 RATIO = 1.051E0
- CELLS IN 33 CM DX = 5.776E-02 RATIO = 1.05E0
- CELLS IN 76 CM DX = 2.93E0 RATIO = 1.05E0

NSPEC = 3

\[ NHNU NBB = 1 \quad ECAL = 4.2E0 \quad START = 0. \quad SSTOP = 3.500E-09 \]
\[ 200 CAL TEMP = 3.7E0 \quad ECAL = 4.2E0 \]
\[ NHNU NBB = 1 \quad ECAL = 8.7E0 \quad START = 3.500E-09 \quad SSTOP = 4.500E-09 \]
\[ 200 CAL TEMP = 2.37E0 \quad ECAL = 8.7E0 \]
\[ NHNU NBB = 1 \quad ECAL = 1.380E+01 \quad START = 4.500E-09 \quad SSTOP = 5.500E-09 \]
\[ 200 CAL TEMP = 2.08E0 \quad ECAL = 1.380E+01 \]

FIGURE C.19 INPUT DECK FOR RADIATION FROM THREE BLACK BODIES INTO ALUMINUM, SHOWING FOUR REZONES AND MULTIPLE ZONES IN ON ONE LAYER.
FIGURE C.20 INPUT DECK FOR RADIATION BY A DEPTH-DOSE PROFILE INTO SEVERAL LAYERS AND ILLUSTRATING USE OF HDATA AND TEDITs
IDENT = 261A EXPONENTIAL LOADING ON A HEAT SHIELD
C THE LOADING IS APPLIED AS AN EXPONENTIAL PRESSURE ON THE FIRST BOUNDARY.
C PARAMETERS OF THE PRESSURE LOADING ARE READ IN THROUGH -EXTRA-.
NTEDT = 0 NJEDIT = 2 NZELON = 0
JEDITS = 1 11 22 33 43 54 65 75 86 97 107 118 129 139
  160 174 179 183
NEEDIT = 10 JCYCS = 120 CKS = 3.00 TS = 7.000E-06
NWTRLS = 3 MATFL = -2 UZERO = 0*

JWR
RHOS = 1.63E0 CFP = 000 DPY = 001 NCON = 0
EOST = 7.490E+10 1.500E+11 1.600E+10 0.174E0 0.25E0 1.310E+11
YIELD = 2.400E+08 3.000E+10 0.0*

BDON
RHOS = 1.1E0 CFP = 000 DPY = 000 NCON = 0
EOST = 2.990E+10 1.588E+11 1.000E+10 0.5E0 0.25E0 -1.450E+11
YIELD = 3.210E+09 2.670E+11 3.790E+10

NLAYER = 3 JMAT = 1 2 3
NZONES = 1 17 CELLS IN 0.076 CM
NZONES = 1 17 CELLS IN 0.254 CM
EXTRA
%NLIST P6(1)=7.1E10, T6(1)=0.385E-6 $
7/8/9

FIGURE C.21 INPUT DECK FOR PRESSURE LOADING ON A THREE-LAYERED PLATE,
SHOWING USE OF THE NAMELIST STATEMENT
FIGURE C.22  INPUT DECK FOR SIMULATING AN AIR SHOCK BY APPLYING A PRESSURE BOUNDARY THROUGH NAMELIST

| IDENT JIM GPANS SHOCK TUBE | NALPHA
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NTEDT = 10, 15, 20, 23, 30</td>
<td></td>
</tr>
<tr>
<td>NJEDT = 10 JCYCS</td>
<td>150 CKS</td>
</tr>
<tr>
<td>NMTRLS = 2 MATFL</td>
<td>-2 UZEPO</td>
</tr>
<tr>
<td>AIR</td>
<td>RHOS = 1.169E-03</td>
</tr>
<tr>
<td>EUST = 0.</td>
<td>0.</td>
</tr>
<tr>
<td>G = 2.17E+09</td>
<td></td>
</tr>
<tr>
<td>TENS = -1.000E+10-1.000E+10-1.000E+10</td>
<td></td>
</tr>
<tr>
<td>VISC = 4.</td>
<td></td>
</tr>
<tr>
<td>STEEL</td>
<td>RHOS = 7.903</td>
</tr>
<tr>
<td>EQST = 1.048E+12 2.432E+12 7.360E+10 1.17</td>
<td></td>
</tr>
<tr>
<td>YIELD = 7.013E+09 7.892E+11</td>
<td></td>
</tr>
<tr>
<td>NLAVER</td>
<td>NJONES = 1 30 CELLS IN 15.5 INCH</td>
</tr>
<tr>
<td>NZONES = 1 5 CELLS IN 25. CM</td>
<td></td>
</tr>
<tr>
<td>EXTRA</td>
<td></td>
</tr>
</tbody>
</table>

%NLIST Po(1)=1.327E7, T6(1)=1.8, RHOS(2)=7.902995137$
The material properties give samples for ductile fracture (DFRACT in Figures C.3, C.7, and C.18), brittle fracture (BFRACT in Figure C.8), and shear banding (SHEAR2 in Figures C.11 and C.12). Porous materials are modelled by POREQST in Figures C.6, C.9, C.13, and C.17 and by CAP1 in Figures C.10 and C.14. The composite model REBAR is used in Figure C.10. The tabular equation of state EOSTAB is used for PETN in the data deck in Figure C.14. Explosives are treated in various ways in Figures C.11, C.12, C.14, C.15, and C.16.

In the layout, most materials are treated with uniform size cells. However, multiple zones within a layer are used in Figures C.15, C.18, and C.19. The geometric cell layout is featured in Figures C.5, C.8, C.14, C.15, and C.17 through C.20. Gaps between layers occur in Figures C.13 and C.15. A large number of layers (up to 30) are permitted as shown in Figure C.10. For planar geometry, an infinite boundary may occur at the first or last coordinate by making the first or last JMAT value negative as shown in Figure C.22. For convenience, the thickness dimension may be inserted in English units if columns 41 to 45 contain the letters "_INCH" (See Figures C.11 and C.22). GENRAT changes the dimensions to centimeters for internal use and for printing later. Depth-dose profiles are shown in Figures C.17, C.18, and C.20; black body x-ray spectra appear in Figure C.19 and an arbitrary spectrum in Figure C.5.

The EXTRA and HDATA lines following the normal data deck permit many special features. In Figure C.3, the EXTRA line provides the internal energy for the hot aluminum and resets the density to its normal value for the equation-of-state calculations. A similar effect is illustrated in Figure C.20. In Figures C.14, C.21, and C.22 a pressure boundary is provided by specifying P6 and T6. A preload is given in Figure C.14. A simultaneous detonation of EL-506D is provided in Figure C.15 by the insertion of internal energy through the EXTRA line. In Figure C.22, the air is initialized at a moderate pressure by providing it with some internal energy (treating it as an explosive undergoing a simultaneous detonation), and the steel is preloaded by decreasing the density in the EXTRA line. Figure C.20 contains a data deck with both
EXTRA and HDATA lines. The HDATA line sets the boundary indicators to the MIRROR case to simulate a fixed or reflecting boundary on both sides.
Appendix D

FMELT: THERMAL REDUCTION FUNCTION

The subroutine FMELT is used to reduce the strength and shear moduli as a function of the internal energy. FMELT contains two functions. The first (F) normally affects the yield strength, spall strength, and the amplitude of the compaction surface in porous materials. The second function (FG) reduces the shear modulus.

FMELT is called in GENRAT for initialization, and in HSTRESS to compute the nondimensional reduction factors.

Formulation of the Model

The strength reduction and modulus reduction factors are presumed to have the form shown in Figure D.1 for several grades of aluminum. The reduction factor is described by a series of parabolas as illustrated in Figure D.2. Up to three parabolas are used. Each parabola is defined by the coordinates of its end points plus the amplitude at its midpoint. These input quantities are transformed to coefficients of the series for F. In the \( i \)th interval, the coefficients are

\[
F = F_{ai+1} + F_{bi+1}E + F_{ci+1}E^2
\]

where

\[
F_{ai+1} = F_{Li} - E_{Li} \left( F_{Ri} - F_{Li} + 4AF \cdot E_{Ri} / \Delta E \right) / \Delta E
\]

\[
F_{bi+1} = (F_{Ri} - F_{Li}) / \Delta E + 4AF(E_{Li} + E_{Ri}) / \Delta E^2
\]

\[
F_{ci+1} = -4AF / \Delta E^2
\]

\[
\Delta E = E_{Ri} - E_{Li}
\]
Figure D.1  Variation of Strength with Temperature for Aluminum 1100

1. MELT = E
   NO PARABOLIC REGIONS

2. MELT = E
   ONE PARABOLIC REGION

3. MELT = E
   TWO PARABOLIC REGIONS

4. MELT = E
   THREE PARABOLIC REGIONS

STANDARD PARABOLA AND DEFINITION OF TERMS IN DERIVATION

FIGURE D.2 DEFINITION OF INPUT FOR THERMAL STRENGTH AND MODULUS REDUCTION FUNCTION


$E_{Ri}, E_{Li}, F_{Ri}, F_{Li} = $ energies and amplitudes on the right and left sides of the interval

$\Delta F = $ the offset of the midpoint of the parabola from a straight line, as shown in Figure D.2.

Sample inputs for the strength and modulus reduction factors are listed in Appendix C.

Several options are available to the user with the FMELT function. From zero to three parabolas may be used to define the function. The number is determined automatically by the number of input values used. Both strength and modulus reduction functions may be used or only the strength reduction function. If only the strength reduction function is supplied, the same function is used for modulus reduction.

The data are supplied as a series of numbers designated $E_1, E_2, \ldots E_8$ in Figure D.2. The first parameter $E_1$ is always the melt energy in erg/g. The other parameters vary in significance according to the number of parabolas as shown in Figure D.2. The sign convention for $\Delta F$ and the slopes at the end of the parabolic segment are shown in the last diagram of D.2. The slopes of the parabola at its ends are determined graphically by passing straight lines through the end points and a point $2\Delta F$ from the midpoint of the straight line segment as shown. It is advisable to examine the slopes to verify that the chosen parabola matches the experimental data adequately and does not contain a local minimum or maximum.
Appendix E

RESIZING THE CELLS: REZONE

The purpose of rezoning is to give the cells an optimum size distribution for the hydrodynamic calculations. During the radiation deposition or shortly after impact (first part of the calculation), the cells near the radiated face or near the impact interfaces should be small to correctly depict the wave motion at those points. Later on, as the waves spread out, the presence of the small cells merely slows down the hydrodynamic computations. Therefore, REZONE is called to gradually increase cell size (the current REZONE does not decrease size). As outlined in Section 5.1, rezoning begins either at the right boundary (negative NREZON) or at JREZON (positive NREZON) and sweeps to the left, resizing groups of cells to obtain the desired size. If cells are already larger, they are unaffected. Because there are fewer cells following each rezoning, the initial coordinate, JINIT, is increased by each call to REZONE.

The following guidelines were used in calculating the redistribution of coordinates:

- Boundaries must remain as coordinate points
- JEDIT locations (Lagrangian coordinates at which printouts are requested) should not be disturbed.
- Cell thicknesses should not be allowed to vary rapidly in a material.
- Across boundaries, the cell thicknesses should vary so that the crossing time of a wavelet is the same across any cell; that is,

\[
\frac{\Delta X_1}{C_1} = \frac{\Delta X_2}{C_2}
\]

(E.1)

where

\[
\Delta X_1, \Delta X_2 = \text{cell thicknesses}
\]

\[
C_1, C_2 = \text{sound speeds.}
\]
- Smoothing of the wave should be minimized. For cell-centered quantities (SHL, PHL, EHL, etc.) this is accomplished by weighting the old cell quantities according to their contribution of mass to the new cell. For example, the new internal energy is computed from

\[
E_{\text{NEW}} = \frac{\sum E_{\text{OLD}}^\alpha \Delta x}{\sum \rho \Delta x}
\]

(E.2)

- As illustrated in Figure E.1, the summations are carried out from XSTART to XFIN, the boundaries of JNEW. For coordinate-centered quantities, such as U, a more complicated technique is required, as explained later.

- Neglect conservation of kinetic energy. Because cells are usually larger when rezoned, this neglect will lead to some loss of total energy.

![Figure E.1 Layout for Computing Properties at Rezoned Coordinates](image)

FIGURE E.1 LAYOUT FOR COMPUTING PROPERTIES AT REZONED COORDINATES

The subroutine that was constructed to perform the rezoning is naturally separable into three parts: one to locate the initiation point of rezoning, one to select rezonable sets of cells, and one to compute the new cell properties. In the first part of the subroutine, the control variable (JTS for NREZON < 0 and JREZON for NREZON > 0) is located with respect to material boundaries. A possible layout of the coordinates is shown in Figure E.2. (JEDITS need not be in numerical order.)

152
Coordinates are not all rezoned at once, but in groups between JEDIT, material boundaries, and spall planes. The second part of the sub-routine searches for these rezonable groups of coordinates. Figure E.3 defines some nomenclature used in the searching process.

**FIGURE E.2** REPRESENTATIVE LAYOUT OF COORDINATES BEFORE REZONING

**FIGURE E.3** REZONABLE SET OF CELLS TERMINATED ON LEFT BY A JEDIT
In the third part of the program, the rezonable set of cells is tested before rezoning. If the number of cells ahead is less than the number that would be obtained in the rezoning, a check is made to determine whether a region of small cells is followed by a region of large cells in the rezonable set. (This is likely in a radiation problem because the surface material expands.) If there is a region of small cells, the rezonable set is truncated to include only those small cells, and rezoning is performed. If the numbers of new cells and old cells are equal and the old cells have a fairly uniform thickness, then the coordinates are simply renumbered. If computation of new properties is called for, the calculations are performed as described in the guidelines above. If the rezonable set of cells is terminated at the left by a boundary or spall surface, then the new coordinate at JFIRST is included in the computation of the current set of cells. For other termination conditions of the rezonable set, new properties are computed up to, but not including, the new coordinate at JFIRST. Those properties will be computed with the next rezonable set.

Conservation of Momentum: Velocity Computation

Several approaches are available for conserving total momentum in computing the new particle velocity array. Because the velocity array is associated with the coordinate points, the approach used was to compute a momentum associated with each coordinate. The requirements for the computation were to:

- Preserve momentum exactly.
- Leave the velocity unaltered if the cell dimensions on both sides of the coordinate are unchanged.

The momentum associated with a coordinate is computed by weighting the momenta near the new coordinate in proportion to the distance from the coordinate.

The momentum is separated into two components: a term proportional to the average momentum (the usual momentum term) and a term related to the variation of momentum across the cell:

154
\[ M_{12} = M_{a12} + M_{b12} \] (E.3)

\[ M_{a12} = \frac{1}{2} \int_{X_1}^{X_1 + \Delta X} \rho u d\xi \] (E.4)

\[ M_{b12} = -\frac{3}{\Delta X} \int_{X_1}^{X_1 + \Delta X} \rho u (\xi - X_1 - \frac{\Delta X}{2}) d\xi \] (E.5)

where
\[ M_{a12} = \text{one-half the momentum of the cell between coordinates 1 and 2} \]
\[ M_{b12} = \text{the contribution to coordinate 1 of the variation of momentum in the cell 1-2} \]
\[ X_1 = \text{the location of coordinate 1} \]
\[ \Delta X = \text{the dimension of cell 1-2}. \]

The coefficient \((-3/\Delta X)\) and weight factor \((\xi - X_1 - \frac{\Delta X}{2})\) in \(M_{b12}\) were determined by requiring that \(M_{b12} = 0\) if \(\rho u\) is uniform in the cell and that the velocity \(U_1\) be unchanged if the cell size is unchanged. The new velocity will be computed from

\[ U_1 = 2 \frac{M_{12} + M_{01}}{Z_{12} + Z_{01}} \] (E.6)

where \(Z_{12}\) is the mass of the cell between coordinates 1 and 2.

To keep \(U_1\) unchanged, \(M_{12}\) must be a function of \(U_1\) only. The momentum at coordinate 2 from cell 1-2 is

\[ M_{21} = M_{a12} - M_{b12} \]

As an example, consider a cell bounded by coordinates with velocities \(U_1\) and \(U_2\). Then the velocity at any point is

\[ U = U_1 + (U_2 - U_1) \frac{\xi - X_1}{\Delta X} \] (E.7)

and
\[ M_{a12} = \frac{1}{4} \rho \Delta X (U_1 + U_2) \]
\[ M_{b12} = \frac{1}{2} \rho \Delta X (U_1 - U_2) \]

Hence,
\[ M_{12} = \frac{1}{2} \rho \Delta X U_1 \]
\[ M_{21} = \frac{1}{2} \rho \Delta X U_2 \]
The more general problem is one in which a portion (from $X'_1$ to $X'_2$) of an old cell contributes to a new cell. The velocities at the boundaries of this portion are computed from

$$U'_1 = U_j + (U_{j+1} - U_j) \frac{X'_1 - X_j}{X_{j+1} - X_j} \quad (E.8)$$

$$U'_2 = U_j + (U_{j+1} - U_j) \frac{X'_2 - X_j}{X_{j+1} - X_j} \quad (E.9)$$

where the $U$ and $X$ quantities with $j$ subscripts refer to the old cell velocities and locations. Let

$$\xi_1 = \frac{X'_1 + X'_2}{2} - X_j - \frac{\Delta X}{2}$$

the distance between centroids of the contributing portion of the old cell and of the new cell; $\xi_2 = X'_2 - X'_1$, the contributing portion of the old cell. Then the momentum contributions of the portion are

$$M_{a12} = \frac{1}{4} \rho \xi_2 (U'_1 + U'_2) \quad (E.11)$$

$$M_{b12} = -\frac{1}{4} \rho \frac{\xi_2}{\Delta X} [6\xi_1 (U'_1 + U'_2) + \xi_2 (U'_2 - U'_1)] \quad (E.12)$$

In the code these two momentum quantities are AMAVG and AMSLP. The sums and differences are stored in the MOM array.

**Detailed Treatment of Coordinate Arrays**

The coordinate arrays may be divided into four groups according to their reference point (cell or coordinate) and numerical or nonnumerical character. The cell quantities are sound speed, density, internal energy, pressure, stresses, yield strength, mass, $H(J,1)$, $H(J,3)$, and other variables associated with the material model. The $H$ quantities are
integers used as indicators; consequently, they cannot be handled by the weighting procedures otherwise appropriate. Density DHL is computed from the mass ZHL, rather than directly by averaging.

The coordinate quantities are X, T, U, and H(J,2). U is computed as described in the previous subsection. T, the spall strength, is set to the initial value TENS(M,1) except at interfaces and spall planes. There it is set to the corresponding T value in the unrezoned array. H(J,2) indicates spall or interface conditions at a coordinate. It is reset in the second (searching) portion of REZONE following computation of new cell quantities.

Printout

Some printout is obtained from each major step in REZONE. Therefore, if problems arise because of rezoning, they can usually be quickly traced.
Appendix F

ONE-DIMENSIONAL CYLINDRICAL AND SPHERICAL FLOW

The basic wave-propagation relations for one-dimensional geometry are derived here for cylindrical and spherical flow. Included are the mass and momentum conservation equations, expressions for the internal energy, elastic-plastic stress-strain relations, and spall equations.

Kinematic Calculations

The equations for mass and momentum conservation and the expressions for internal energy are derived here.

For spherical flow, consider two finite-difference cells bounded by radii \( r_1 \), \( r_2 \), and \( r_3 \) and subtending an arc of \( d\theta \) in orthogonal circumferential directions as shown in Figure F.1. The mass of cell 1 is

\[
M_1 = \frac{\rho_0}{3} d\theta^2 (r_2^3 - r_1^3)
\]  

(F.1)

Mass conservation is provided by storing \( Z_1 = M_1/d\theta^2 \) as a constant for cell 1. Then the density at any time is

\[
\rho = \frac{3Z_1}{r_2^3 - r_1^3}
\]  

(F.2)

Conservation of momentum is the basis for determining the velocities of cell boundaries. The mass associated with each boundary point is half the mass in the two adjacent cells. The forces acting on this mass are computed from the stresses in the adjacent cells and the areas they act on. The stress in the cell between \( r_1 \) and \( r_2 \) acts at a mid-mass radius given by

\[
\frac{r_1^3}{r_1^3} = \frac{r_2^3 + r_1^3}{2}
\]  

(F.3)
FIGURE F.1  CELL GEOMETRY CONSIDERED FOR ONE-DIMENSIONAL SPHERICAL MOMENTUM CALCULATIONS
and thus on the area $A_1$ shown in Figure F.1

$$A_1 = d\theta^2 \bar{r}_1^2$$  \hspace{1cm} (F.4)

The radial component of the tangential stress is $\sigma_{\theta 1} d\theta/2$, which acts on the area $A_{12}$ on each side of the cell in Figure F.1:

$$A_{12} = \frac{\bar{r}_1 + \bar{r}_2}{2} (r_2 - \bar{r}_1)d\theta$$  \hspace{1cm} (F.5)

Assembling all the radial forces on the mass centered at $r_2$ and extending from $\bar{r}_1$ to $\bar{r}_2$ gives

$$\sigma_{r1} d\theta^2 \bar{r}_1^2 - \sigma_{r2} d\theta^2 \bar{r}_2^2 + 4 \left[ \sigma_{\theta 1} \frac{d\theta}{2} \left( \frac{\bar{r}_1 + \bar{r}_2}{2} \right) (r_2 - \bar{r}_1) d\theta \right]$$

$$+ \sigma_{\theta 2} \frac{d\theta}{2} \left( \frac{\bar{r}_2 + \bar{r}_2}{2} \right) (\bar{r}_2 - r_2) d\theta$$

$$= \frac{\rho d\theta^2}{3} \frac{r_3^3 - r_1^3}{2} \frac{\Delta U_2}{\Delta t}$$  \hspace{1cm} (F.6)

Here $\Delta U_2$ is the change in velocity of the coordinate $r_2$. Elimination of $d\theta^2$ and use of Eq. (F.1) for the definition of the initial cell mass leads to

$$\Delta U_2 = 6 \Delta t \left[ \frac{\sigma_{r1} \bar{r}_1^2 - \sigma_{r2} \bar{r}_2^2 + \sigma_{\theta 1} (\bar{r}_1 + r_2) (r_2 - \bar{r}_1) + \sigma_{\theta 2} (r_2 + r_2) (\bar{r}_2 - r_2) }{Z_1 + Z_2} \right]$$  \hspace{1cm} (F.7)

In Eq. (F.7), the radial stresses are augmented by the artificial viscosity stresses for the momentum calculations. No artificial viscosity is added to the tangential stresses.

The change in internal energy that arises from the work done is

$$\Delta E = V \int \sigma d\varepsilon = - \int P dV + V \sum_i \int \sigma_{i1} d\varepsilon_i$$  \hspace{1cm} (F.8)
Through use of the stress and strain definitions in Section 4.1, the energy change reduces to

\[ \Delta E = - \int \sigma_1 dV + \frac{3}{2} \int \sigma_{1}^{-1} (dV + V \varepsilon_1) \]  \hspace{1cm} (F.9)

This expanded form is convenient for computations because the first term is the expression for planar flow and the second is added only for spherical flow.

For cylindrical flow, the finite-difference cell is bounded by an inner radius \( r_1 \) and an outer radius \( r_2 \), subtends an arc of \( d\theta \), and has indefinite extent in the Z direction. Motion occurs only in the radial direction. The mass of the cylindrical cell is

\[ M_1 = \frac{\rho_o}{2} d\theta \left( r_2^2 - r_1^2 \right) \]  \hspace{1cm} (F.10)

where the cell is assumed to have unit length in the Z direction. Mass is conserved at each cell by storing the mass \( Z = M/d\theta \) for each cell and computing the density \( \rho \) at any time from the geometry and \( Z \) as in Eq. (F.10):

\[ \rho = \frac{2Z_1}{r_2^2 - r_1^2} \]  \hspace{1cm} (F.11)

Momentum conservation follows the same plan as for spherical flow. First, a mid-mass radius is defined:

\[ r_1^2 = \frac{r_1^2 + r_2^2}{2} \]  \hspace{1cm} (F.12)

The area of action of the radial stress at mid-cell is

\[ A_1 = r_1 d\theta \]  \hspace{1cm} (F.13)
and the area for the circumferential stress $\sigma_1$ is

$$A_{12} = d\theta (r_2 - r_1)$$  \hspace{1cm} (F.14)

Assembling all the radial forces on the boundary-centered mass at $r_2$ leads to

$$\sigma_{r1} d\theta \cdot \vec{r}_1 - \sigma_{r2} d\theta \cdot \vec{r}_2 + \sigma_{\theta1} d\theta (r_2 - \vec{r}_1)$$

$$+ \sigma_{\theta2} d\theta (\vec{r}_2 - r_2) = \rho d\theta \left( \frac{r_2^2 - r_1^2}{2} \right) \frac{\Delta U_2}{\Delta t}$$  \hspace{1cm} (F.15)

Elimination of $d\theta$ and use of the definition of $Z$ leads to

$$\Delta U_2 = 4\Delta t \left( \frac{\sigma_{r1} \vec{r}_1 - \sigma_{r2} \vec{r}_2 + \sigma_{\theta1} (r_2 - \vec{r}_1) + \sigma_{\theta2} (\vec{r}_2 - r_2)}{Z_1 + Z_2} \right)$$  \hspace{1cm} (F.16)

As in spherical flow, the radial stresses in Eq. (F.16) are augmented by the artificial viscosity stresses; the tangential stresses are not.

The change in internal energy in cylindrical flow is computed from Eq. (F.8) with the aid of the stress and strain definitions in Section 4.1. The energy change is

$$\Delta E = - \int \sigma_1 dV + \int (\sigma'_1 - \sigma'_2)(dV + V d\epsilon_1)$$  \hspace{1cm} (F.17)

The first term is the expression used for planar flow. The second term is simply added for cylindrical flow. This term is similar to the second term in Eq. (F.9) because $\sigma'_2 = -\sigma'_1/2$ in spherical flow.

The foregoing analyses have been implemented into the SRI PUFF code for one-dimensional wave propagation.
Elastic-Plastic Calculations for Planar, Cylindrical, and Spherical Geometries

In this section, the general elastic and plastic calculations of Appendix G are applied to one-dimensional flows with linear work-hardening.

In planar flow, strain occurs only in the direction of propagation, and the transverse strains $\varepsilon_2$ and $\varepsilon_3$ are zero. Such planar flow occurs during impact of flat plates and in response to a simultaneous detonation of an explosive over a plane. In cylindrical flow, only radial motion occurs. Thus radial and circumferential strains are nonzero but axial strain is zero. Cylindrical flow occurs in the response of long buried tunnels, pipe lines, and in fragmenting rounds or bombs. In spherical flow, the flow is all radial and the transverse strains are equal and nonzero.

The equations for one-dimensional flow are summarized in Table F.1. The deviator strain is defined as

$$d\varepsilon_i' = d\varepsilon_i - \frac{1}{3}(d\varepsilon_1 + d\varepsilon_2 + d\varepsilon_3) \quad (F.18)$$

The equivalent shear strain quantities are derived from Eq. G.5. The expressions for the equivalent stress $\bar{\sigma}$ are from Eq. G.4.

For planar flow, stresses are found by first computing $\sigma_{1}^{N}$ from

$$\sigma_{1}' = \sigma_{1}^{N} = \sigma_{10}^{N} + \frac{4}{3} \mu \Delta \varepsilon_{1} \quad (F.19)$$

which can be obtained from Eq. (G.11) because $\Delta \varepsilon_{1}' = \frac{2}{3} \Delta \varepsilon_{1}$. If $\sigma_{1}^{N}$ exceeds $2Y_o/3$, then from Eq. (G.27)

$$\sigma_{1}' = \frac{M\sigma_{1}^{N} + 2G Y^*}{M + 3G} \quad (F.20)$$

where $Y^*$ has the same sign as $\sigma_1'$ and $|Y^*| = Y_o$, the yield strength
## Table F.1

STRESS AND STRAIN QUANTITIES IN ONE-DIMENSIONAL FLOW

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Planar</th>
<th>Cylindrical</th>
<th>Spherical</th>
</tr>
</thead>
<tbody>
<tr>
<td>( d\varepsilon_1 )</td>
<td>( \frac{\partial u}{\partial x} )</td>
<td>( \frac{\partial u}{\partial r} )</td>
<td>( \frac{\partial u}{\partial r} )</td>
</tr>
<tr>
<td>( d\varepsilon_2 )</td>
<td>0</td>
<td>( \frac{u}{r} )</td>
<td>( \frac{u}{r} )</td>
</tr>
<tr>
<td>( d\varepsilon_3 )</td>
<td>0</td>
<td>0</td>
<td>( \frac{u}{r} )</td>
</tr>
<tr>
<td>( d\varepsilon'_1 )</td>
<td>( \frac{2}{3} d\varepsilon_1 )</td>
<td>( d\varepsilon_1 - \frac{1}{3} \frac{\partial \rho}{\partial r} )</td>
<td>( d\varepsilon_1 - \frac{1}{3} \frac{\partial \rho}{\partial r} )</td>
</tr>
<tr>
<td>( d\varepsilon'_2 )</td>
<td>( -\frac{1}{3} d\varepsilon_1 )</td>
<td>( -d\varepsilon_1 + \frac{2}{3} \frac{\partial \rho}{\partial r} )</td>
<td>( -\frac{1}{2} d\varepsilon_1 + \frac{1}{6} \frac{\partial \rho}{\partial r} )</td>
</tr>
<tr>
<td>( d\varepsilon'_3 )</td>
<td>( -\frac{1}{3} d\varepsilon_1 )</td>
<td>( -\frac{1}{3} \frac{\partial \rho}{\partial r} )</td>
<td>( -\frac{1}{2} d\varepsilon_1 + \frac{1}{6} \frac{\partial \rho}{\partial r} )</td>
</tr>
<tr>
<td>( d\varepsilon )</td>
<td>( \frac{2}{3}</td>
<td>d\varepsilon_1</td>
<td>)</td>
</tr>
<tr>
<td>( d\varepsilon^p )</td>
<td>(</td>
<td>d\varepsilon^p_1</td>
<td>)</td>
</tr>
<tr>
<td>( \sigma )</td>
<td>( \frac{3}{2}</td>
<td>\sigma'_1</td>
<td>)</td>
</tr>
<tr>
<td>( \sigma'_2 )</td>
<td>( -\frac{1}{2} \sigma'_1 )</td>
<td>-</td>
<td>( -\frac{1}{2} \sigma'_1 )</td>
</tr>
</tbody>
</table>

**Notes:**
- Subscript 1 is in direction of propagation
- 2 is in \( \theta \) direction in cylindrical flow or any transverse direction for the other two flows
- 3 is in third orthogonal direction
- \( \varepsilon, \varepsilon', \varepsilon^p \) are strain, deviator strain, plastic strain, positive in tension
- \( \sigma, \sigma', \sigma^p \) are stress, deviator stress, equivalent or Mises stress
- \( \gamma, \gamma^p \) are equivalent shear strain, equivalent plastic shear strain
- \( u \) is displacement in the direction of motion
- \( x, r \) is coordinate in the direction of motion
- \( \rho \) is density.

165
at the previous time step. The plastic strain is found from Eq. (G.22), accounting in addition for the possibility that the increment includes the beginning of yielding.

\[
\Delta \varepsilon^p_1 = \frac{2G\Delta \varepsilon - Y^*}{M + 3G} + \frac{3}{2} \sigma^1 \varepsilon^p_1
\]  

(F.21)

and

\[
Y = Y_o + M\Delta \varepsilon^p = Y_o + \frac{3}{2}M|\Delta \varepsilon^p_1|.
\]

This result agrees with the fact that, for perfect plasticity (M = 0), the plastic strain is

\[
\Delta \varepsilon^p_1 = \Delta \varepsilon^+ = \frac{2}{3} \Delta \varepsilon_1
\]  

(F.22)

and there is no change in the elastic deviator strain.

For cylindrical flow, two deviator stresses \( \sigma^N_1 \) and \( \sigma^N_2 \) are calculated from Eq. (G.26) and then \( \sigma^N_1 \) is evaluated from Eq. (G.4). If \( \sigma^N_1 \) exceeds \( Y \), then \( \bar{\sigma} \) is reduced from \( \sigma^N_1 \) as follows:

\[
\bar{\sigma} = \frac{M\sigma^N_1 + 3GY}{M + 3G}
\]  

(F.23)

The individual deviator stresses are then calculated from Eq. (G.25):

\[
\sigma^*_1 = \frac{\sigma^N_1 Y}{\sigma^N_1}
\]  

(F.24)

\[
\sigma^*_2 = \frac{\sigma^N_2 Y}{\sigma^N_2}
\]  

(F.25)

where \( Y = Y_o + M\Delta \varepsilon^p \).

The plastic shear strain is obtained as in Eq. (F.21):
and from Eqs. (G.24) and (G.25):

\[
\Delta \varepsilon_1^P = \Delta \varepsilon_1^E - \Delta \varepsilon_1^E = \Delta \varepsilon_1^E (1 - \frac{Y}{\sigma N})
\]  

(F.27)

For spherical flow, \( \sigma_1^N \) is first computed elastically as usual and compared with \( 2Y_o/3 \). If yield has occurred,

\[
\sigma_1^E = \frac{M \sigma_1^N + 2GY_o^*}{M + 3G}
\]  

(F.28)

and the plastic strain is

\[
\Delta \varepsilon_1^P = \frac{3G \Delta \varepsilon_1^E - Y_o^* + \sigma_o}{M + 3G}
\]  

(F.29)

Note that in spherical flow, the relations for \( \sigma_1^E \) and \( \Delta \varepsilon_1^P \) are almost identical to those in planar flow.

The plastic strain energy is associated with work hardening, temperature rise, and thermal softening, and is used in some dislocation models. The plastic energy is defined as

\[
\Delta E^P = V \sum_1 \sigma_1^E \Delta \varepsilon_1^P
\]  

(F.30)

where \( V \) is specific volume and \( \Delta E^P \) is the increase in specific internal energy. For planar and spherical geometries, the energy change is

\[
\Delta E^P = \frac{3}{2} V \sigma_1^E \Delta \varepsilon_1^P
\]  

(F.31)

A convenient form for the energy change in the cylindrical case is

\[
\Delta E^P = V \left[ \sigma_1^E (2\Delta \varepsilon_1^P + \Delta \varepsilon_2^P) + \sigma_2^E (2\Delta \varepsilon_2^P + \Delta \varepsilon_1^P) \right]
\]  

(F.32)
Appendix G

DEVIATOR STRESS MODELS

This appendix gives a derivation for a three-dimensional deviator stress model including elastic, plastic, and work-hardening behavior. The plasticity model is then expanded to encompass strain rate effects.

Plasticity Relations for Mises-Type Models

A three-dimensional computational model was developed for yielding based on Reuss (incremental or flow) plasticity, Von Mises yield behavior, and work hardening (See Hill for general background). The following four assumptions form the basis of the model:

1. The strain can be separated into an elastic and a plastic component at each step. As in elasticity, the stress is proportional to the elastic strain component

\[ \text{de} = \text{de}^E + \text{de}^P \]  

2. According to Reuss, the shear (or deviator) stress in any direction is proportional to the increment of plastic strain in that direction. The mathematical formulation of the condition is

\[ \frac{d\varepsilon_{12}^P}{\sigma_{12}} = \frac{d\varepsilon_{23}^P}{\sigma_{23}} = \frac{d\varepsilon_{13}^P}{\sigma_{13}} = \frac{d\varepsilon_{11}^P}{\sigma_{11}} = \ldots = d\lambda \]  

These relations provide for changes in the directions of the principal stresses. Inherent in Eq. (G.2) is the assumption that there is no volume change in plastic strain, i.e.,

\[ d\varepsilon_{1}^P + d\varepsilon_{2}^P + d\varepsilon_{3}^P = 0 \]  

where the singly subscripted strains are principal.
3. The behavior is homogeneous and isotropic even with work-hardening. Because there is no directionality, the state can be defined completely by scalars. The chosen scalars are an effective stress $\bar{\sigma}$ and an effective strain $\bar{\varepsilon}^P$, which are invariant under rotation and do not distinguish between the three principal directions. For convenience, the effective stress is chosen so that $\bar{\sigma} = \sigma$ at yield. The usual definition for the effective stress has the following forms:

$$
\bar{\sigma} = \sqrt{\frac{1}{2}[(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2]} \quad (G.4a)
$$

$$
= \sqrt{\frac{3}{2}[(\sigma_1')^2 + (\sigma_2')^2 + (\sigma_3')^2]} \quad (G.4b)
$$

$$
= \sqrt{\frac{3}{2}[(\sigma_x')^2 + (\sigma_y')^2 + (\sigma_z')^2 + 2(\tau_{yx}^2 + \tau_{zx}^2 + \tau_{xy}^2)]} \quad (G.4c)
$$

$$
= \sqrt{3[(\sigma_1')^2 + (\sigma_2')^2 + \sigma_1'\sigma_2']} \quad (G.4d)
$$

where

$$
\sigma_1', \sigma_2', \sigma_3' = \text{principal stresses}
$$

$$
\sigma_1', \sigma_2', \sigma_3' = \text{deviator stresses in the principal directions}
$$

$$
\sigma_x', \sigma_y', \sigma_z' = \text{deviator stresses in the coordinate directions}
$$

$$
\tau = \text{shear stress}
$$

A similar definition is given to the effective plastic strain, $\bar{\varepsilon}^P$. The amplitude is fixed by requiring that $\bar{\varepsilon}^P = \varepsilon_1^P$ for any case where $\varepsilon_2^P = \varepsilon_3^P$. Then $\bar{\varepsilon}^P$ has the forms

$$
\bar{\varepsilon}^P = \varepsilon_1^P - \frac{2}{3}[\varepsilon_1^P - \varepsilon_2^P]^2 + (\varepsilon_2^P - \varepsilon_3^P)^2 + (\varepsilon_3^P - \varepsilon_1^P)^2] \quad (G.5a)
$$

$$
= \sqrt{\frac{2}{3}[\varepsilon_1^P]^2 + (\varepsilon_2^P)^2 + (\varepsilon_3^P)^2]} \quad (G.5b)
$$
$= \sqrt[3]{\frac{2}{3} \left[ (d\varepsilon_1^p)^2 + (d\varepsilon_2^p)^2 + (d\varepsilon_3^p)^2 + 2(d\varepsilon_{xy}^p)^2 + 2(d\varepsilon_{yz}^p)^2 + 2(d\varepsilon_{zx}^p)^2 \right]}$ (G.5c)

$= \sqrt[3]{\frac{4}{3} \left[ (d\varepsilon_1^p)^2 + (d\varepsilon_2^p)^2 + d\varepsilon_1^p d\varepsilon_2^p \right]}$ (G.5d)

$= \sqrt[3]{\frac{2}{3} d\varepsilon_{ij}^p d\varepsilon_{ij}^p}$

where

$\begin{align*}
d\varepsilon_1^p, d\varepsilon_2^p, d\varepsilon_3^p &= \text{plastic strains in the principal directions} \\
d\varepsilon_x^p, d\varepsilon_y^p, d\varepsilon_z^p &= \text{plastic strains in the coordinate directions} \\
d\varepsilon_{xy}^p, d\varepsilon_{yz}^p, d\varepsilon_{zx}^p &= \text{plastic shear strains (tensor components)}.
\end{align*}$

4. The yield condition describes yielding as a function $\phi$ of the second invariant $J'_2$ of the deviator stress tensor

$\phi = J'_2 - \kappa^2 = 0$ (G.6)

where

$J'_2 = \frac{-2}{3} = \frac{1}{6} \left[ (\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2 \right]$ 

and $\kappa$ is a constant (yield strength in pure shear). Yielding occurs when Eq. (G.6) is satisfied. Alternatively, the yield criterion can be expressed in terms of the equivalent stress $\bar{\sigma}$ and the yield strength $Y$ in simple tension.

The preceding assumptions form the basis of a plasticity model with an "associated flow" rule. For such a model, both the stress-strain relations, Eq. (G.2), and the yield function, Eq. (G.6), employ the same function $\phi$. That is, Eq. (G.2) can be put into the form

$\frac{d\varepsilon_{ij}^p}{d\lambda} = \frac{\partial \phi}{\partial \sigma_{ij}} d\lambda.$ (G.8)
In a model with an associated flow rule, the plastic strain vector in principal strain space is always normal to the yield surface in stress space; this condition introduces simplifications that will be used later.

Next we introduce the elastic stress-strain relations:

\[ \sigma'_{ij} = 2G(\varepsilon^E_{ij} - \frac{\delta_{ij}}{3} \Sigma \varepsilon_{ii}) \]  

(G.9)

For convenience, we can simplify Eq. (G.9) by defining a deviator strain similar to the deviator stresses:

\[ \varepsilon'_{ij} = \varepsilon_{ij} - \frac{\delta_{ij}}{3} \Sigma \varepsilon_{ii} \]  

(G.10)

Then Eq. (G.9) takes the form

\[ \sigma'_{ij} = 2G\varepsilon'_{ij}^E \]  

(G.11)

The plastic flow relations, Eq. (G.2), are now rewritten into a convenient form. If each term in Eq. (G.2) is put in the form

\[ \frac{9}{4}(d\varepsilon^P)^2 = \sigma^2(d\lambda)^2 \]  

(G.12)

Replacing this value for \( d\lambda \) in Eq. (G.2) provides the convenient form

\[ \sigma'_{ij} = \frac{2G}{3} \frac{d\varepsilon^P}{d\varepsilon^P} \]  

(G.13)

To complete the model, we will assume that work hardening, if it occurs, is a function only of the equivalent plastic strain. The increase in the yield strength is

\[ dY = M\varepsilon^P \]  

(G.14)
where $M$ is the work-hardening modulus. Hence, the work hardening assumed is independent of the direction of straining so that material remains isotropic during plastic flow.

The problem we face can now be formulated as: Given the total strain increments, the stress components at the previous time, and the yield strength, solve Eqs. (G.1), (G.3), (G.11), and (G.13) simultaneously for the stresses $\sigma'_{ij}$. To aid in visualizing this problem, we introduce a vector notation for both principal deviator stress and principal deviator strain:

$$\vec{\sigma} = \sigma^i \hat{i} + \sigma^j \hat{j} + \sigma^k \hat{k}$$

(G.15)

and similarly for strain $\vec{\epsilon}$. For elastic behavior, Eq. (G.11) shows that

$$\vec{\sigma} = 2G\vec{\epsilon}$$

(G.16)

so that the two vectors are coaxial. The strain vectors are illustrated in Figure G.1 and we can imagine a corresponding stress diagram with the same directions, but magnified by $2G$. An initial yield surface is shown as the ellipse defined by the elastic strain corresponding to $\vec{\sigma}_o = Y_o$. The equation of the ellipse is given by Eq. (G.4d) and (G.11).

$$\left(\epsilon_1^E\right)^2 + \left(\epsilon_2^E\right)^2 + \epsilon_1^E \epsilon_2^E = \frac{Y_o^2}{12G^2}$$

(G.17)

Now strain increments are added to the components of the elastic strain deviator tensor defining point A to obtain a new tensor with components $\epsilon_{ij}$

$$\epsilon_{ij} = \epsilon_{ij}^E + \Delta\epsilon_{ij}$$

(G.18)

* In three-dimensional principal stress space the yield surface is a cylinder with its axis equiangular to the three principal directions and with radius $\sqrt{2/3} Y$. In principal deviator stress space, the yield locus is the circle on this cylinder with the center at the origin. When viewed parallel to the third axis, the circle appears as an ellipse in the 1, 2 plane.
FIGURE G.1  VECTORIAL REPRESENTATION OF PRINCIPAL STRAINS DURING AN INTERVAL OF PLASTIC FLOW WITH WORK HARDENING
where none of these tensors is necessarily oriented in such a way that the components are principal. When the new total strain tensor is diagonalized to obtain the principal deviator strains, they define a new point C. (We take C to be outside the yield surface to illustrate yielding.) Because of the diagonalizations involved in proceeding from point A to C, a vector from point A to C does not have a simple relation to the strain increment tensor.

The new elastic strain state (and stress state) is given by the vector \( \vec{OB} \) which terminates on a yield surface which has expanded because of the work hardening. We can determine the coordinates of the point B by using the facts that

- The elastic strain vector is coaxial with the stress vector and has amplitude given by Eq. (G.16)
- The plastic strain increment vector is coaxial with the stress.

Then the plastic strain increment is

\[
\Delta \varepsilon^P = \dot{\varepsilon} - \varepsilon^E
\]

\[
= \dot{\varepsilon} - \frac{\sigma}{2G}
\]  \hspace{1cm} (G.19)

where \( \vec{\sigma} \) is the vector \( \vec{OB} \) and is proportional to the current yield value. With the aid of Eqs. (G.16) and (G.14), Eq. (G.19) can be transformed to a scalar equation because all the vectors are coaxial

\[
\Delta \varepsilon^P = \varepsilon - \frac{1}{3G} (\sigma_o + M \Delta \varepsilon^P)
\]  \hspace{1cm} (G.20)

Here we used the facts derived from Eqs. (G.4b) and (G.5b) that

\[
\varepsilon = \sqrt{\frac{2}{3}} |\varepsilon| \hspace{0.5cm} \text{and} \hspace{0.5cm} \sigma = \sqrt{\frac{3}{2}} |\sigma|
\]  \hspace{1cm} (G.21)

Solving for \( \Delta \varepsilon^P \) provides

\[
\Delta \varepsilon^P = \frac{3G \varepsilon - \sigma_o}{M + 3G}
\]  \hspace{1cm} (G.22)
With $\Delta e^P$ known, the yield value can be found from Eq. (G.14). The elastic strain is

$$\varepsilon^E = \varepsilon - \Delta e^P$$  \hspace{1cm} (G.23)

and the individual strains and stresses are

$$\varepsilon^E_{ij} = \varepsilon_{ij} - \frac{e^E}{e}$$  \hspace{1cm} (G.24)

$$\sigma^N_{ij} = \sigma^N_{ij} \frac{e^E}{e} = \sigma^N_{ij} \cdot Y_{ij}$$  \hspace{1cm} (G.25)

where

$$\sigma^N_{ij} = 2G \varepsilon^E_{ij}$$  \hspace{1cm} (G.26)

and $\sigma^N$ is calculated from the $\sigma^N_{ij}$ using equations of the same form as Eq. (G.4). With the aid of Eqs. (G.16) and (G.22), the stresses may also be evaluated as

$$\sigma^*_{ij} = \sigma^N_{ij} \cdot M + 3G \frac{\sigma^N}{\sigma^N}$$  \hspace{1cm} (G.27)

Normally in wave propagation calculations, the strains are computed at each step but not stored. From the strains, new stresses are computed and stored until the next cycle. Equations (G.25) and (G.26) are the only ones needed for perfect plasticity model calculations. For linear work hardening, Eqs. (G.25) and (G.27) are required and a yield value must be stored for each cell.

The individual plastic strain increments are obtained by inverting Eq. G.13 and using the deviator stress from Eq. G.25 or G.26.

$$d\varepsilon^P_{ij} = \frac{3d\varepsilon^P}{2\sigma} \sigma^*_{ij}$$

Note that the plastic strain increments are not necessarily proportional to the applied strain increments.
The foregoing relations are simplified for the cases of one-dimensional flow in Appendix F.

The preceding equations are valid whenever the change in direction of $\vec{\sigma}$ is small in an increment. This restriction arises because $d\varepsilon$ is calculated as if it were proportional to and in the direction of the final values of $\varepsilon$ in the increment. In a more accurate calculation $d\varepsilon^P$ would be directed toward an average $\varepsilon$ during the increment. However, for most calculations with solids this latter refinement is not necessary.

**Strain-Rate Effects**

The linear-viscous model for strain-rate effects is used here. Initially, the analysis is developed for the case of pure shear and then transformed to the multidimensional case. In terms of shear stress $\tau$, the stress-strain relation is

$$\frac{\partial \tau}{\partial t} = C \frac{\partial \gamma}{\partial t} - \frac{\tau - Y}{T}$$  \hspace{1cm} (G.28)

where $T$ is the time constant, $\gamma$ is the shear strain, and $Y_T$ is the yield stress in shear. With this form, a very rapid loading proceeds elastically, because the first two terms dominate. For gradual loading, $\tau$ must remain near $Y_T$ in the plastic range, so the behavior is like rate-independent plastic flow. At intermediate rates, an initial overshoot of $\tau$ above $Y_T$ occurs, and then $\tau$ gradually reduces to $Y_T$. For computational purposes, we consider a short time interval, $\Delta t$, over which the strain rate is known and constant. The shear stress at any time, $t$, in the interval is obtained by integrating Eq. (G.28)

$$\tau = \tau_1 e^{-t/T} + (C\gamma T + Y_T)(1 - e^{-t/T})$$  \hspace{1cm} (G.29)

where $\tau_1$ is the shear stress at the beginning of the interval.

The analogous calculation is performed for a multidimensional flow by casting Eq. (G.28) in the following form.
\[
\frac{d\sigma_{ij}^\prime}{dt} = 3G \frac{d\varepsilon_{ij}^\prime}{dt} - \sigma_{ij}^\prime - \frac{2Y}{3} \frac{d\varepsilon_{ij}^P}{d\varepsilon^P}
\]  

(G.30)

As in Eq. (G.28), the first term on the right-hand side is the elastic relation. The second term represents the excess stress above the static yield value; this excess is driving the rate process. The static yield stress in the \(ij\) direction is obtained from Eq. (G.13) as \(2Y\varepsilon_{ij}^P/3\varepsilon^P\).

Equation (G.30) is then integrated, holding all strain rates constant in the interval:

\[
\sigma_{ij}^\prime = \sigma_{ijo}^\prime + \left[-\sigma_{ijo}^\prime + \frac{2Y\varepsilon_{ij}^P}{3\varepsilon^P} + 2GT \frac{d\varepsilon_{ij}^\prime}{dt} \right] \left[1 - e^{-(t - t_0)/T}\right]
\]

(G.31)

where \(\sigma_{ijo}^\prime\) and \(t_0\) are deviator stress and time at the beginning of the interval. Equation (G.23) can be evaluated for a time step if an estimate of \(d\varepsilon_{ij}^P/d\varepsilon^P\) can be obtained from Eq. (G.13). In our calculations, the first estimate is

\[
\frac{d\varepsilon_{ij}^P}{d\varepsilon^P} \approx \frac{\sigma_{ij}^N}{\sigma_{ij}^N}
\]

(G.32)

where the stress quantities are computed elastically. Subsequent estimates are based on the results of the evaluation of \(\sigma_{ij}^\prime\) from Eq. (G.31). Equation (G.32) represents a good approximation when only small changes are evident in the relative importance of the components of the stress tensor, that is, when only small changes occur in the principal stress directions.
Appendix H

INSERTION PROCEDURE

As new material models are generated, they can be added to SRI PUFF for performing wave propagation calculations. This appendix describes the procedure for inserting material model subroutines.

A wave propagation code normally has four main categories of operations: reading the input data, initializing a finite difference grid, performing calculations for each time increment at each grid point, and printing the computed information. A material model subroutine may be involved in all or some of these operations. Call statements must be provided in SRI PUFF at appropriate locations to accomplish these tasks. Also the new subroutine should be provided with separate sections for each operation and an indicator to show which operation to perform. For example, in SHEAR2 the formal parameter NCALL indicates the operation required, as follows:

NCALL = 0 Initialize the routine and read data for one material
  1 Read data for one material
  2 Calculate stresses and damage
  3 Calculate stresses and damage, and print results
  4 Print results only.

The calls for NCALL = 0 and 1 are in GENRAT. There, NCALL is LSUB(15), a parameter that is initially zero. After the first call, LSUB(15) is set to 1. For NCALL = 2 and 3, the call statement is in HSTRESS. Other calling strategies are also possible. For example, BFRACT is initialized on the first call from HSTRESS; there are no other calls. EXPLODE is called from GENRAT to read data and then called for each cell during the layout to initialize array variables. During propagation calculations, EXPLODE is also called by HSTRESS.
At the point of insertion of the call statement, four elements are provided.

(1) The appropriate branching statements are needed to switch to the new model when it is required. For SHEAR2, it was decided to treat the model as a fracture routine and designate it by NFR(M) = 3. Then the available branching statements in GENRAT and HSTRESS were amplified to include one more branch.

(2) Variables must be initialized, calibrated, or given sign changes just preceding the call statement.

(3) The call statement is provided.

(4) Some variables may need to be reset following the calculations in the routine. Then a jump is provided to the appropriate section of HSTRESS or GENRAT to continue the calculation.

Items (2), (3), and (4) are discussed further below following introduction of a call statement.

A sample call statement for SHEAR2 is listed here as it appears in HSTRESS, but the same call can be used in GENRAT.

CALL SHEAR2 (NCALL, IN, M, J, H(J,3), SX, SY, SZ, TXY, PHL(J), COM(L), DH, DOLD, DT, EH, EOLD, COM(L+1), EMELT(M,1), COM(L+2), EX, EY, EZ, EXY, F, YHL(J), COM(L+3), ROT, DROT, ESC, COM(L+4)). Because SHEAR2 represents a fairly complex case, this call statement will be discussed in detail.

The initialization of NCALL for use in GENRAT was described above. For HSTRESS, NCALL is initialized just before the call statement. NCALL is set to 2 normally, but it is set to 3 on cycles when an EDIT will occur. The parameter IN is the file containing input data. Normally IN is 5 but may be reset in GENRAT to 4 for a special data file. The coordinate number J appears twice because the SHEAR2 subroutine is also used in two-dimensional calculations where two indices are needed. The stress components SX, SY, SZ, TXY are positive in tension in HSTRESS, although the array quantities SHL, PHL, SDT, and SDH are positive in compression. If necessary, sign and magnitude changes can be made in the stresses just preceding the call statement. The strain quantities EX, EY, EZ, EXY are also positive in tension. In SHEAR2 most of the material properties are
inserted in two large arrays: ESC and TSR. The ESC array, listed in Table H.1, is for the usual equation of state parameters, whereas TSR is for the special fracture parameters. The rotation parameter ROT is zeroed before the call are stored in the COM array, beginning at location L = LVAR(J). The use of COM and LVAR is described in Appendix C.

Following insertion of a new material model, it is a good plan to run a simple problem with frequent EDITS to determine whether the routine is performing satisfactorily.

Table H.1
MATERIAL PARAMETER ARRAY ESC

<table>
<thead>
<tr>
<th>No.</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Original density, g/cm³</td>
</tr>
<tr>
<td>2</td>
<td>Bulk modulus (C), dyn/cm²</td>
</tr>
<tr>
<td>3,4</td>
<td>D and S in the pressure equation:</td>
</tr>
<tr>
<td></td>
<td>P = Cμ + Dμ² + Sμ³</td>
</tr>
<tr>
<td></td>
<td>where μ = density/ESC (M,1) - 1</td>
</tr>
<tr>
<td>5</td>
<td>Shear modulus (G), dyn/cm²</td>
</tr>
<tr>
<td>6</td>
<td>YADD, work hardening modulus, dyn/cm²/(g/cm³)</td>
</tr>
<tr>
<td>7</td>
<td>Initial solid density, g/cm³</td>
</tr>
<tr>
<td>9</td>
<td>Grüneisen ratio</td>
</tr>
<tr>
<td>10</td>
<td>Initial yield strength, dyn/cm²</td>
</tr>
</tbody>
</table>

Notes

Array dimension is ESC (6,20) with the first subscript for material number and the second for property number (the number listed above). Thus ESC (M,5) is the shear modulus for material M. The ESC array is initialized in GENRAT at the end of the materials loop.
Appendix I
LISTING OF SRI PUFF 8

The following listing contains all the routines currently used with PUFF. The main program is given first, with all the subroutines following in alphabetical order. Included are SRI PUFF8, BANDRLX, BAUSCHI, BECOM, BEMOD, BFRACT, CAP1, DEPOS, DFRACT, EDIT, EOSTAB, EPLAS, EQST, EQSTPF, ESA, EXPLODE, EXTRA, FMELT, GENRAT, GRAY, HAFSTEP, HDATA, HSTRESS, HYDRO, HYPO, PEST, POREQST, PORHOLT, PRESCR, REBAR, REDR, RELAX, REZONE, SCATTO, SCRIBE, SHEAR2, SIGMAT, SSCALH, STORR, STRES2, and TSQE. A brief description of each subroutine and references for the material models are given in SECTION 2.
SUBROUTINE SRIPUFF

PROGRAM SRIPUFF (INPUT, OUTPUT, TAPE5=INPUT, TAPE6=OUTPUT, TAPE3=2500, SHIPUFF )
\[ \text{TAPE4, TAPE7, TAPE2=400) } \]
\[ \text{SRIPUFF3 } \]
\[ \text{SRIPUFF4 } \]
\[ \text{SRIPUFF5 } \]
\[ \text{SRIPUFF6 } \]
\[ \text{SRIPUFF7 } \]
\[ \text{SRIPUFF8 } \]
\[ \text{SRIPUFF9 } \]
\[ \text{SRIPUFF10 } \]
\[ \text{SRIPUFF11 } \]
\[ \text{SRIPUFF12 } \]
\[ \text{SRIPUFF13 } \]
\[ \text{SRIPUFF14 } \]
\[ \text{SRIPUFF15 } \]
\[ \text{SRIPUFF16 } \]
\[ \text{SRIPUFF17 } \]
\[ \text{SRIPUFF18 } \]
\[ \text{SRIPUFF19 } \]
\[ \text{SRIPUFF20 } \]
\[ \text{SRIPUFF21 } \]
\[ \text{SRIPUFF22 } \]
\[ \text{SRIPUFF23 } \]
\[ \text{SRIPUFF24 } \]
\[ \text{SRIPUFF25 } \]
\[ \text{SRIPUFF26 } \]
\[ \text{SRIPUFF27 } \]
\[ \text{SRIPUFF28 } \]
\[ \text{SRIPUFF29 } \]
\[ \text{SRIPUFF30 } \]
\[ \text{SRIPUFF31 } \]
\[ \text{SRIPUFF32 } \]
\[ \text{SRIPUFF33 } \]
\[ \text{SRIPUFF34 } \]
\[ \text{SRIPUFF35 } \]
\[ \text{SRIPUFF36 } \]
\[ \text{SRIPUFF37 } \]
\[ \text{SRIPUFF38 } \]
\[ \text{SRIPUFF39 } \]
\[ \text{SRIPUFF40 } \]
\[ \text{SRIPUFF41 } \]
\[ \text{SRIPUFF42 } \]
\[ \text{SRIPUFF43 } \]
\[ \text{SRIPUFF44 } \]
\[ \text{SRIPUFF45 } \]
\[ \text{SRIPUFF46 } \]
\[ \text{SRIPUFF47 } \]

C

INTEGER H, POROUS, PRESS, RINTER, SOLID, SPALL
REAL MATL, NEM, NET, NEMH, NETH
COMMON AZERO(1), CEF, CKS, DAVG, DELTIM, DISCPT(10), DOLD, DRHO, DTMAX,
1 DTMIN, DTN, DTNH, DU, DX, EOLD, F, FAC, FIRST, J, JCYS, JINIT,
2 JFIN, JREZ0N(15), JSMAX, JSTAR, JTS, LSUB(30), M, MAXPR(30), NCYC,
3 NEDIT, NPERN, NR, NREZ0N, NSCRB(6), NSEPRAT, NSPALL, NTEDT,
4 NTX, NTR(15), POOL, P6(20), R(30), RLAST, SLAST, SMAX, TEDIT(50),
5 TF, TMAX, TJ, TREZON, TS(6), ULAST, UOLD, UX, UZERO, XLAST, XNOW, XOLD
6
7
C

COMMON DH, DHLAST, DUH, EH, PH, RH, RHLAST, SH, SLAST, UH, UHLAST, XH, XHLAST,
PUPCOM 2
1 NEMH, NETH

COMMON DXX(30), JBNO(30), JMAT(6), NLAYER, NMTRLS,
PUPCOM 3
1 THK(30)

COMMON/COORD/X(200), XO(200), CHL(200), DHL(200), DPDD(200), DPDE(200),
PUPCOM 4
1 COORDC0

COMMON /RAD/ SST0P(5), START(5), SDURM, SSSTOPM, NSPEC, SSJ, SSC/4, 4000
1 X(200), YH(200), ZH(200)

COMMON /RAD/ SST0P(5), START(5), SDURM, SSSTOPM, NSPEC, SSJ, SSC/4, 4000
1 X(200), YH(200), ZH(200)

COMMON /RAD/ SST0P(5), START(5), SDURM, SSSTOPM, NSPEC, SSJ, SSC/4, 4000
1 X(200), YH(200), ZH(200)

COMMON AZERO(1), CEF, CKS, DAVG, DELTIM, DISCPT(10), DOLD, DRHO, DTMAX,
PUPCOM 5
1 DTMIN, DTN, DTNH, DU, DX, EOLD, F, FAC, FIRST, J, JCYS, JINIT,
2 JFIN, JREZ0N(15), JSMAX, JSTAR, JTS, LSUB(30), M, MAXPR(30), NCYC,
3 NEDIT, NPERN, NR, NREZ0N, NSCRB(6), NSEPRAT, NSPALL, NTEDT,
4 NTX, NTR(15), POOL, P6(20), R(30), RLAST, SLAST, SMAX, TEDIT(50),
5 TF, TMAX, TJ, TREZON, TS(6), ULAST, UOLD, UX, UZERO, XLAST, XNOW, XOLD
6
7
C

CALL SECOND(FIRST) $ XIN=FIRST
C

CALL GENRAT
C

IF (JCYCS .LE. 0) GO TO 100

NPERN=MAXO(NPERN,1)
CN=NCYC=NPERN $ IT=MINO(0, NTEDT-1) $ NT=0 $ SF=0.8
N=N+1
C

CALL HYDRO
C

CALL PRINTS
C

IF (MOD(N, 25) .EQ. 0) 205, 400
C

CALTIM=FIRST
C

WRITE(6,889) NT, TMAX, MST, DTNM, SMAX, JSMAX,
C

IF (MOD(N, NEDIT) .EQ. 0 .AND. N .NE. JCYCS) CALL EDIT
C

IF (LSUB(7) .EQ. 1) GO TO 390
C

CALL STORE (JTS, 1000)
C

CALL EDIT
C

CALL REZONE
C

CALL SCRIBE
C

CALL GENRAT
C

IF (TIME .LT. TS) 304, 400
C

IF (N .EQ. JCYCS) 400, 305
C

CALL PRINTS
C

CALL SCRIBE
C

CALL STORE (JTS, 1000)
C

CALL PRINTS
C

CALL SCRIBE
C

CALL STORE (JTS, 1000)
C

CALL PRINTS
C

CALL SCRIBE
C

CALL STORE (JTS, 1000)
C

CALL PRINTS
C

CALL SCRIBE
C

CALL STORE (JTS, 1000)
C

CALL PRINTS
C

CALL SCRIBE
C

CALL STORE (JTS, 1000)
C

CALL PRINTS
C

CALL SCRIBE
C

CALL STORE (JTS, 1000)
C

CALL PRINTS
C

CALL SCRIBE
C

CALL STORE (JTS, 1000)
C

CALL PRINTS
C

CALL SCRIBE
C

CALL STORE (JTS, 1000)
C

CALL PRINTS
C

CALL SCRIBE
C

CALL STORE (JTS, 1000)
C

CALL PRINTS
C

CALL SCRIBE
SUBROUTINE SRIPUFF (Concluded)

305 IF (X(JSMAX) - CKS) 500,400,400
C  
C  ERROR FINISH
390 N=N+1
400 WRITE (6,641)
WHITE (6,640) N,JCYCS,TIME,TS,X(JSMAX),CKS,LSUB(7),DTNH
LSUB(7)=1 CALL EDIT $ CALL STORR $ CALL SCRIBE
C  PROGRAM HALTS ON COMPLETION OF ALL DATA DECKS
GO TO 100
C  
C  TIME STEP CALCULATION
500 DTNH=AMIN1(SF*DTMIN,AMAX1(1.2*DTNH,0.035*SF*DTMIN))
IF (NSPEC .EQ. 0 OR SDURM .EQ. 1.) GO TO 530
SOURM=1.
DO 510 NS=1,NSPEC
IF (TIME+DTNH .LT. START(NS) .AND. TIME .LT. SSTOP(NS)) DTNH =
AMIN1(DTNH,AMAX1(START(NS)-TIME,0.035*(SSTOP(NS)-START(NS))))
IF (TIME+.5*DTN .LT. SSTOP(NS)) SDURM=0.
510 CONTINUE
530 CN=NCYCS=NPERN
C  PERIODIC REZONE
C  
C  IF (NREZON .GE. 0) GO TO 534
C  
C  IF (TIME .LT. TREZON) GO TO 534
C  
C  IF (DTNH .LE. DTMAX) GO TO 534
C  
C  10 IF (IT) 560,550,530
534 CALL EDIT $ CALL REZONE $ NR=NR+1
C  
C  TIME EDIT AND REZONE CALL
535 CALL EDIT $NT=NT+1
537 CALL REZONE $ NR=NR+1
538 IF (NT .EQ. NITEDT) 537,538
540 IT=1 $ GO TO 560
545 IT=0
550 IF (TIME+CN*DTNH .LT. TEDIT(NT)) 560,555
555 NCYCS=TEOIT(NT)-TIME)/DTNH $ CN=NCYCS
DTNH=AMAX1((TEDIT(NT)-TIME)/CN+0.1*DTNH) $ IT=1
560 N=N+1
565 N=N-1 $ GO TO 400
C  
C  840 FORMAT(/ 5H  N =I4,9H, JCYCS =I4, 8H, TIME =1PE10,3,6H, TS =
1 E10+3,12H* X(JSMAX) =E10+3,7H* CKS =E10+3,10H* LSUB(7)=I3,
2 E10+3,8H* DTNH =1PE10+3)
841 FORMAT (/4X,28H«»«*» CRITERION FOR STOP ••••)
849 FORMAT (5H N =I5,8H, JSTAR =I4,7H, TIME =1PE10,3,12H, CALC TIME=
1 F10,3,11H SECS, JTS=I4,7H DTNH=1PE10,3,7H SMAX=1PE10,3,
2 8H JSMAX=I4/)
SUBROUTINE BANDRLX

SUBROUTINE HANORLX (ICON, SO, Y1, DTO, COEF, N, J, M, NM, NT, UT, TSH, MUM,
1 YD, INSR)

C CALLED BY *HSTRESS* TO COMPUTE DEVIATOR STRESS FOR HANILMAN
C GILMAN RELAXATION MODELS; NDS = 2 AND 3.
C FOLLOWING TABLE GIVES CORRESPONDENCE OF COMMON, BANDRLX VARIABLES
C COMMON NSR TSR(1) (2) (3) (4) (5) (6) NEM NET
C COMMON 2 T1 T2 BEE VM GEE EPS NM NT
C GILMAN 3 CEE PHI BEE VM BNMO - NM GAM
C NOTE. MEOW=MUM, YAO=YAD, INS=NSR
C NET AND NET ARE MOBILE AND TOTAL DISLOCATIONS
C GAM IS PLASTIC SHEAR STRAIN
C JK IS A PATH INDICATOR
C INPUT - ALL FORMAL PARAMETERS.
C OUTPUT - SO, ICON, NM, NT, YNOT.
C
REAL NM, NT, NMO, NTO, MEOW, MUM
DIMENSION TSR(b,30)

YAO=0.6667 *YD
YNOT=0.6667*Y1

T1=TSR(M,15) $ T2=TSR(M,16) $ BEE=TSR(M,17)
VM=TSR(M,18) $ GEE=TSR(M,19) $ EPS=TSR(M,20)
MEOW=2.0*MUM

ICOR = ICON $ YNOT=YNOT
NTO = NT $ NMO = NM $ SOO = SO
NIT=4
L=0 $ ENT=FLOAT(NIT) $ IT=0
SIGNH = SIGN(I.*SOO)
IF (ICON .EQ. 2) L=10

INITIAL CONDITIONS INSIDE ELASTIC ZONE
2 SD=SOO*COEF
IF (ABS(SD).GT.YNOT)5,66

DEVIATOR LEAVES ELASTIC ZONE. CALCULATE RELAXATION
5 L = 1
S = .5*(ABS(SD+COEF)-YNOT)
DELT = (SD-SIGN(YNOT,COEF))/(SD-SOO)*DT
SIGNH= SIGN(I.*COEF)
ENT=1. $ SD=SDO

GOTO 40

INITIAL CONDITION OUTSIDE OF ELASTIC ZONE
10 L=2
IT=IT+1 $ SDI=SDO+COEF/2.*ENT
S=ABS(SOI)-YNOT $ DELT=OT/ENT
IF(S.LE.0.)1H,11

AVERAGE DEVIATOR REMAINS OUTSIDE ELASTIC ZONE. CALCULATE RELAXATION
11 L=3
IF(SIGNH.EQ.SIGN(I.*SO1)) 40,17
13 IF(ABS(SD).GT.YNOT)14,16
14 L=4
IF(SIGN(I.*SO1).EQ.SIGNH) 15,16

DEVIATOR REMAINS OUTSIDE ELASTIC ZONE AFTER RELAXATION
15 L=5
IF(IT.EQ.NIT) 62,10
SO=SDI-COEF/(2.*ENT)
16 L=6

DEVIATOR REENTERS ELASTIC ZONE. RECALCULATE RELAXATION
18 S = .5*(ABS(SO)-YNOT) $ YSTAR = SIGN(YNOT,SOO)
DELT = (ystar=SOO)/COEF*DT
GO TO 40
19 SD=SD+COEF/ENT*FLOAT(NIT-IT)
IF (ABS(SD).GT.YNOT)21,20
20 ICON = 2
GO TO 66
C DEVIATOR CHOSSES OVER TO OPPOSITE SIDE OF ELASTIC ZONE, RECALCULATE

186
SUBROUTINE BANDRLX (Concluded)

21 IF (SIGN(1.*SD),EQ.,SIGHN) GO TO 62
SIGHN = -SIGHN $ L = 7
DELT = (SD*YSTAR)/COEF*OT
S = .5*(AHS(SD)-YNOT)
SD=SD-COEFF $ ENT=1.
40 AWG=4.*BEE/3.*S
IF (AWG.GT.20) GO TO 42
42 XPO = 0. $ GO TO 45
43 XPO = EXP(-AWG)
45 GO TO (96,51,52) INSR
C PERFORM RELAXATION CALCULATIONS - BAND MODEL
51 TP=NT
NT=NT*(EPS*GEE*5*(NT-NM)-1./T2)*(NM*XPO)*DELT
NM=NM*(GEE*5*(TP-NM)+1./T1+1./T2)*(NM*XPO)*DELT
GO TO 54
C PERFORM RELAXATION CALCULATIONS - GILMAN MODEL
52 CEE=T1 $ PHI=T2 $ GMU=GEE $ GAM=GAMO=NT
NM=NM*(1.+CEE*GAM)*EXP(-PHI*GAM)
54 SD=SD+COEF/ENT-1.333*NM*XPO*DELT*SIGHN
55 GO TO (60,19,13,96,19,60)L
60 ICON = 2-IFIX(SIGN(1.*SD))
C RECALCULATE YIELD STRENGTH IN CASES OF STRAIN HARDENING
62 YNOT = AMIN1(AHS(SD),YNOT+YAD*ABS(DHO))
IF (YNOT.EQ.AHS(SD)) GO TO 64
64 ICON=2 $ L= L+50
66 CONTINUE
GO TO (96,90,78) INSR
78 OGAM = AHS(SD0-COEF-SD)/2.667/MUM
NT=GMU+OGAM$GMU
Y1 = 1.5 *YNOT
90 RETURN
96 WRITE (6,199) INSR,L
Y1 = 1.5 *YNOT
RETURN
199 FORMAT (25H ERROR IN BANDRXL,INSR = I5,S9H, L = I5)
END
SUBROUTINE BAUSCHI

C ROUTINE PROVIDES A BAUSCHINGER EFFECT WHEN DEVIATOR CHANGES
C
C SIGN. FUNCTION HAS THE FORM (S/S0)**N = EP/EPM
C
C INO=H(J=3), T = COMPRESSION, 0 = TENSION
C SD = DEVIATOR STRESS, INPUT AS SD0, OUTPUT AS SDH OR SDJ
C DS = CHANGE IN DEVIATOR, SD = SD0-DS, INPUT AS ELASTIC CHANGE
C YC = NEM, CURRENT YIELD, SET TO ZERO WHEN SIGN OF DEVIATOR CHANGES
C Y = YHL, YIELD (USED AT 2/3RD ACTUAL VALUE)
C EP = NET(1), PLASTIC STRAIN, RESET TO ZERO WHEN DEVIATOR CHANGES SIGN
C EPM=TSR(1), PLASTIC STRAIN AT WHICH BAUSCHINGER EFFECT CEASES
C HT = TSR(2), WORK HARDENING MODULUS IN TENSION
C HC = TSR(3), WORK HARDENING MODULUS IN COMPRESSION
C XP = TSR(4)=1/N IN DEFINING EQUATION, EXPONENT
C M = 4/3RD ELASTIC SHEAR MODULUS, M = MU*EXMAT(M,1) *(D/RHO-D)
C
C REAL M
Y = 0.6667 * YHL
M = 1.333 * G
IF (DS > SD, GE, 0.) GO TO 100

C BEGIN ROUTE FOR CHANGE IN DIRECTION OF LOADING
IF (SD > SD, GE, 0.) GO TO 400
C STRESS HAS CHANGED SIGN. PREPARE FOR BAUSCHINGER EFFECT
IF (ABS(EP) > 0.) GO TO 100
YC = EP/EP**$10; GO TO 300

C BEGIN ROUTE FOR CONTINUED LOADING IN SAME DIRECTION
IF (ABS(EP) > 1.) GO TO 300
100 IF (ABS(SD*DS, LT, YC) GO TO 400
C BRANCH TO BAUSCHINGER PATH IF PLASTIC STRAIN IS LESS THAN EPM
IF (YC, LT, Y AND ABS(EP) < EPM) GO TO 300
C
C LINEAR WORK HARDENING PATHS
C
C COMPRESSION
200 IF (SD < 0.) GO TO 220
SD = YC = Y = SD***(DS*HC*M*(Y-SD))/(HC*M)
DEP = SD*OS*YC / (M+H) $ EP = EP+DEP
YHL = 1.5 * Y
RETURN
C LINEAR WORK HARDENING IN TENSION
220 SD = SD***(DS*HT-M*(Y*SD))/(HT*M)
YC = Y = -SD
DEP = (SD*DS*YC)/(M+H) $ EP = EP+DEP
YHL = 1.5 * Y
RETURN
C
C BAUSCHINGER - NONLINEAR WORK HARDENING - PATH
C
C SET INITIAL PLASTIC MODULUS AND WORK HARDENING MODULI
300 IF (ABS(EP) < LT, 1., E-4) GO TO 310
H0 = AMIN1(YC*XP/ABS(EP),1,E-14)
DEPA=ABS(SD*DS-SIGN(YC,SD*DS))/(H0*M)
H0 = 0.5*H0*(YC+H0*DEPA)/ABS(EP)+DEPA)
GO TO 315
310 H0 = ABS(SD*DS-SIGN(YC,SD*DS))/(EPM*(ABS(SD*DS)/Y)**(1./XP)-ABS(EP))
315 H = HC
IF (SD > SD, LT, 0.) H = HT
L = 0
C INITIAL ESTIMATES OF EP = ANO - YC -
EP = (SO+DS-SIGN(YC,SD+DS))/(H0*M)
EPABS = ABS(EP+DEP)
YC = YAMIN1(Y = (EPABS)**(EPABS) + H*AMIN1(ABS(DEP),AMAX1(0.,EPABS))
1 - EPM))
H0 = YC*XP/EPABS
IF (EPABS < LT, EPM) H0 = H
DSE = SIGN(YC,DS) = SD
SUBROUTINE BAUSCHI (Concluded)

C BEGIN ITERATIONS FOR PLASTIC STRAIN AND YIELD

L=L+1
DEP2=DEP
DEP=(DS-DSE+HO*DEP)/(HO*M)
IF (DEP*DEP2 .LT. 0.) DEP=DEP2/3.
EPABS=ABS(DEP)
YC=Y*AMIN1(EPABS/EPM)**XP)*H*AMIN1(ABS(DEP))*AMAX1(0.*EPABS)
HO=Y*EPABS/EPM
IF (EPABS .GT. FPM) HO=H
IF (L .GT. 10) GO TO 350

350 SD=SIGN(YC*DS)
Y=AMAX1(Y,YC)
EP=EP*DEP
Y=L = 1.5 * Y
RETURN

C ELASTIC ROUTE

SD=SD*DS
RETURN
END
SUBROUTINE BECOM

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C
SUBROUTINE BECOM (Continued)

*** KS .EQ. 3 - RELOADING FROM AN ELASTIC POINT TO A PLASTIC POINT. THE FRACTION OF THE TIME STEP IN THE PLASTIC REGIME IS CALCULATED. ***

5 0

FT=(TAUEL-ZTAUYMX)/(TAUEL-ZTAUY)
RECOMPUTE TAUEL USING UPDATED MODULUS
C1=AMIN1(ZAMUSV+AAMAX1(ZAM-ALF*ABS(EENP1+FT*VELS),1,1))
TAUEL=C1*(EENP1-1.5*EPN)
FT=(TAUEL-ZTAUYMX)/(TAUEL-ZTAUY)
ZTAUY=TAUN=ZTAUYMX
GO TO 90

*** KS .EQ. 4 - FIRST ELASTICALLY CALCULATED POINT IN UNLOADING PHASE. THE TAU AND TAUY VERSUS STRAIN CURVES CRUSS TTT FORE TIME(N+1). ***

50 IF (KK .GT. 0) GO TO 54

*** UNLOADING FROM INITIAL LOADING PHASE ***

50 PLASTIC STRAIN IN FIRST PART OF TIME STEP IS FROM

EPDOT = 4/3*PHI*PSI/(PHI*PSI) = A

DTAU=TAUN-ZTAUY
A=0.
IF (ABS(DTAU) .LT. 1.E7) GO TO 52
EPTOT=ABS(EPMAXC-EPN)
PHI=SIGNT*AMAX1(1.E-5,AMAX1(1.E6,ABS(DTAU/(VO+BETA*EPTOT*2))))
1 EM
CALL BEMOD(KK,J,SIGNT,EPN,ANM)
A=1.333/(EM+1.)/(1./PHI*BOB2/ANM*(1./DTAU+1.)/(TAUN*ZTAUY))
1 +0.25*(TAUN+ZTAUY)/(BVI0B**2))
52 TAU1=ZTAUY+*SQR1(1.+CO*ABS(ZEPMAXC-EPN-A*DTNP1))
FT=DTAU/(TAU1-ZTAUY+C1*(VELS+1.5*A*DTNP1))
F1=AMAX1(0.,AMIN1(1.,FT))
EC=EEN*FT*VELS
EPNP1=EPN+A*DTNP1*FT
ZTAUYMX=ZTAUY*+SQR1(1.+CO*ABS(ZEPMAXC-EPN))
GO TO 56

*** UNLOADING FROM A LOADING PHASE OTHER THAN THE INIT. LOADING ***

50 DTAU=TAUN-ZTAUY
A=0.
IF (ABS(DTAU) .LT. 1.E7) GO TO 55
EPTOT=ABS(EPMAXC-EPN)
PHI=SIGNT*AMAX1(1.E-5,AMAX1(1.E6,ABS(DTAU/(VO+BETA*EPTOT*2))))
1 EM
CALL BEMOD(KK,J,SIGNT,EPN,ANM)
A=1.333/(EM+1.)/(1./PHI+BOB2/ANM*(1./DTAU+1.)/(TAUN*ZTAUY))
1 +0.25*(TAUN+ZTAUY)/(BVI0B**2))
55 TAU1=ZTAUY*+SQR1(1.+EXP(-SMA*SQRT(ABS(ZEPMAXC-EPN-A*DTNP1))))
FT=DTAU/(TAU1-ZTAUY+C1*(VELS+1.5*A*DTNP1))
F1=AMAX1(0.,AMIN1(1.,FT))
EC=EEN*FT*VELS
EPNP1=EPN+A*DTNP1*FT
ZTAUYMX=ZTAUY+*SQR1(1.+CO*ABS(ZEPMAXC-EPN))
GO TO 56

C UPDATE OF SHEAR STRESS AT TIME N+1

C 56 ZEPDSV=0.
ZTAUY=ZTAUYMX-VELS*C1*(1.-FT)
TAUJ=ZTAUY
IF (TAUJ*ZTAUYMX .GT. 0.) GO TO 340
KS=2
ZTAUYMX=0.
TAUEL=TAUJ
GO TO 20
SUBROUTINE BECOM (Continued)

BEGIN ITERATION LOOP FOR STRESS AND PLASTIC SHEAR STRAIN

90 EPNP1=EPN
EPTOT=ABS(ZEPMAXC-EPN)
IF (FT .LT. 0.) GO TO 300
FTDT=FTDTNP1
FTVELS=FTVELS

200 EPTOT=ABS(ZEPMAXC-EPN-ZEPDSV*FTDT)
DTAU=TAUN-ZTAUY
PHI=SIGNT*AMAX1(1.*E-6,AMIN1(1.*E6,ABS(DTAU/(VO+BETA*EPTOT**2)))).*
1 EM
T13=DTAU/BVIOB
PSI=0.
IF (ABS(T13) .LT. .01) GO TO 205
T14=(TAUJ+ZTAUY)/BVIOB
CALL BEMOD(KK,J,SIGNT,EPN,AMN)
PSI=RAWV/ANM*(T13/(SORT(T13**2+1.)*1.)+T14/(SORT(T14**2+1.)*1.))

205 EPDO=1./(PSI+1./PHI)

NEXT ESTIMATE OF PLASTIC STRAIN IS BASED ON EPDO

EPNP1=EPN+EPDO*FTDT
EPNSUM=EPN

NTIMES=MAX1(5.3,ABS(EPNP1-EPN)/(ABS(EPTOT)+1.E-12))
DEPB=DEPA-ZEPDSV*FTDT/NTIMES
DEPR=DEPA-SIGN1(1.E-12*VELS)
DEPA=EPDO*FTDT/NTIMES
DO 280 NNN=1,NTIMES

ITERT=1
EPTOT=ABS(ZEPMAXC-EPNSUM-0.5*DEPB)
ECC=EENP1+(1.-(NNN-0.5)/NTIMES)*FTVELS
C1=AMIN1(ZAMUSV,AMAX1(ZAM-ALF*ABS(ECC)+1.))
TAUJ=C1*(ECC-1.5*(EPN+0.5*DEPB))
IF (KK .EQ. 0) TAUJ=STAUO*SORT(1.*COEPTOT)
IF (KK .EQ. 1) TAUJ=STAUO*1.*(EXP(-SMA*SURT(EPTOT)))
DTAU=TAUJ-TAUYJ

CALL BEMOD(KK,J,SIGNT,EPNSUM,0.5*DEP8.ANM)
PSI=RAWV/ANM*(T13/(SORT(T13**2+1.)*1.)+T14/(SORT(T14**2+1.)*1.))

DO 265 NNN=1,NTIMES

ITERT=ITERT+1
EPTOT=ABS(ZEPMAXC-EPNSUM-0.5*DEPB)
IF (ABS(T13) .LT. .01) GO TO 265
T14=(TAUJ+ZTAUY)/BVIOB
CALL BEMOD(KK,J,SIGNT,EPN,AMN)
PSI=RAWV/ANM*(T13/(SORT(T13**2+1.)*1.)+T14/(SORT(T14**2+1.)*1.))

265 EPDP=DEPB+DEPA*EPDO*FTDT/NTIMES
DEP=(DEPB+DEPA*DEPAA*DEP)/DEPB+DEPA+DEP+1.E-12
LOC=265

EPNP1=EPN+DEP
IF (ABS(DEP) .LT. 0.02*ABS(DEPBB)) OR. ABS(DEP) .LT. 1.E-10) GO TO 275
IF (ITERT .GE. 20) GO TO 295
IF (ITERT .EQ. 0) GO TO 267
IF (ABS(DEP) .GT. ABS(DEPA-DEP)) AND. MOD(ITERT+3) .NE. 3)
1 GO TO 270

GO TO 268

267 DEP=DEPB
DEPA=DEP
DEPAA=DEPBB
1 ITERT=ITERT+1
DEPA=DEP
GO TO 262
DEPBB=DEPBB
EPNSUM=EPNSUM+DEP
IF (ITERT .EQ. 1 AND. DEP .EQ. 0.) EPNSUM=EPNSUM+DEPBB
CONTINUE
EPNP1=EPNSUM
GO TO 300
SUBROUTINE BECOM (Concluded)

295     LOC=295
296     EPNP1=EPN*EPDU*FTDT
297     PRINT 1295, LOC,J,N,EPNP1,DEPA,DEPB,DEPAA,DEPBB,EPN,EPDU,FTDT
1295   FORMAT(* LOC=*14,** J,N=*2I4,** EPNP1=**1P9E11.3)  
300     CONTINUE
301     C1=AMIN1(ZAMUSV,AMAX1(ZAM-ALF*ABS(EENP1)+1.,))
302     TAUJ=C1*(EENP1-1.5*EPN1)
303     EPTOT=ABS(ZEPMAXC-EPN1)
304     IF (KK .EQ. 0) TAUJJ=STAUI*SQRT(1.+CU*EPTOT)
305     IF (KK .EQ. 1) TAUJJ=STAUI*(1.-EXP(-SMA*SQRT(EPTOT)))
310     IF (ABS(TAUJ) .GT. ABS(TAUJ)) GO TO 330
311     KS=4
312     GO TO 50
C
330     IF (ABS(TAUJ) .LE. 0.) TAUJ=SIGN(1.,TAUJ)
331     ZTAUJ=TAUJ
332     ZEPDSV=EPDUJ
340     SDH=4.*TAUJ/3.
341     ZEP=EPN1
C
RETURN
C
END
SUBROUTINE BEMOD

COMMON /S2/ ALF,CO,EEN,EENP1,EPN,KS,TAUEL,TAUI,TAUN,TAUO,VELS,VMU,ALCOM
1 ZAM,ZAMUSV,ZEP,ZEPDSV,ZEPMAXC,ZEPMAXS,ZEPSAVE,ZTAUY,ZTAUYMX

DATA ANMO,C8,ANMI2,A2/2.75E6,1.E12,1.E6,1.E4/

*** SUBROUTINE BEMOD CALCULATES THE MOBILE DISLOCATION DENSITY
AND ITS DERIVATIVE WITH RESPECT TO PLASTIC STRAIN FOR
BERYLLIUM ***

IF (KK .GT. 0) GO TO 10

*** LOADING PHASE ***

ANM=ANMO*C8*ABS(ZEPMAXC-EPNPH)*2
RETURN

*** UNLOADING OR RELOADING PHASE ***

10 EPB=ZEPMAXS-ABS(ZEPSAVE)*ABS(ZEPMAXC-EPNPH)
ANMS=ANMO*C8*ABS(ZEPSAVE)*2
ANM=ANMI2+(ANMS-ANMI2)*EXP(-A2*EPB*2)
RETURN

END
SUBROUTINE BFRACT
SUBROUTINE BFRACT(LS, SXXEN, SYXEN, STTEN, TXXEN, TXYEN, EXX1, EYY1, ET11, EXY1, BFRAC2 2 1 P, NM, NT, VO, VCDL, DTO, E, EEST, EGSCM, EGSTM, ELMU, TSR, Y, BD, FS, JS, BFRAC2 3 2 M, NN, RHOS, DROT, ROT, FU2D, CL, CN) BFRAC2 4
C BFRAC2 5 NEM -- RELATIVE VOLUME OF CRACKS BFRAC2 6 C BFRAC2 7 NET -- NUMBER OF CRACKS/UNIT VOLUME BFRAC2 7 C BFRAC2 8 T1 -- CRACK GROWTH COEFFICIENT, CM2/DYN/SEC BFRAC2 8 C BFRAC2 9 T2 -- THRESHOLD STRESS FOR GROWTH, DYN/CM2 BFRAC2 9 C BFRAC2 10 T3 -- PARAMETER OF NUCLEATION DISTRIBUTION, CM BFRAC2 10 C BFRAC2 11 T4 -- NUCLEATION RATE COEFFICIENT BFRAC2 11 C BFRAC2 12 T5 -- THRESHOLD STRESS FOR NUCLEATION BFRAC2 12 C BFRAC2 13 T6 -- DENOMINATOR OF EXPONENTIAL STRESS FUNCTION BFRAC2 13 C BFRAC2 14 T7 -- NOT USED BFRAC2 14 C BFRAC2 15 T8 -- THRESHOLD STRESS FOR ENTERING BFRACT BFRAC2 15 C BFRAC2 16 T9 -- SWITCH TO INDICATE WHETHER S OR SDH GOVERNS NUCLEATION BFRAC2 16 C BFRAC2 17 0 STRESS GOVERNS BFRAC2 17 C BFRAC2 18 1 DEVIATOR STRESS GOVERS BFRAC2 18 C BFRAC2 19 T10-- BETA, RATIO OF NO. OF FRAGMENTS TO NO. OF CRACKS BFRAC2 19 C BFRAC2 20 T11-- GAMMA, RATIO OF FRAGMENT RADIUS TO CRACK RADIUS BFRAC2 20 C BFRAC2 21 T12-- VALUE OF CRACK VOLUME WHICH DEFINES THRESHOLD OF COALESCENCE BFRAC2 21 C BFRAC2 22 T13-- TF, WHERE FRAGMENT VOLUME = TF*RF**3 BFRAC2 22 C BFRAC2 23 CN -- CRACK DENSITY, NUMBER/CM3 BFRAC2 23 C BFRAC2 24 CL -- CUBE OF CRACK RADIUS, CM3 BFRAC2 24 C BFRAC2 25 DIMENSION TSR(6,30), FN(7), CL(1), CN(1), COS2TH(4), SIN2TH(4), CL3(5), BFRAC2 26 1 FNUC(5), STH(5), INIT(6), VCR(6), VFR(6), VCN(6) BFRAC2 27 REAL NM, NT BFRAC2 28 DATA ALF, SMF, NANG/1.0, 1.88, 5/ BFRAC2 29 IF (LS, ST. 0) GO TO 20 BFRAC2 30 C BFRAC2 31 ************** ************** BFRAC2 32 C BFRAC2 33 ************** ************** BFRAC2 33 C BFRAC2 34 ************** ************** BFRAC2 34 LS = 1 BFRAC2 35 DO 5 I = 1, 6 BFRAC2 36 INIT(I) = 0 BFRAC2 37 DO 5 J = 1, 7 BFRAC2 38 FN(J) = 0. BFRAC2 39 5 CONTINUE BFRAC2 40 NANG1 = NANG - 1 BFRAC2 41 FNUC(1) = 0.707107/NANG1 BFRAC2 42 FNUC(NANG) = 0.292893 BFRAC2 43 COS2TH(1) = 1.0 BFRAC2 44 SIN2TH(1) = 0. BFRAC2 45 DO 10 NG = 2, NANG1 BFRAC2 46 FNUC(NG) = FNUC(1) BFRAC2 47 T1WOTH = 6.283185*FLOAT(NG-1)/FLOAT(NANG1) BFRAC2 48 COS2TH(NG) = COS(T1WOTH) BFRAC2 49 T2WOTH = 2.826161*FLOAT(NG-1)/FLOAT(NANG1) BFRAC2 50 10 CONTINUE BFRAC2 51 C *** INITIALIZE -Tsr- COEFFICIENTS FOR EACH MATERIAL BFRAC2 52 20 IF (INIT(M) .EQ. M) GO TO 25 BFRAC2 53 TSR(M, 3) = TSR(M, 3)**3 BFRAC2 54 VCR(M) = 8.*((1./ELMU+1.)/(EGSCM+ELMU/3.)) BFRAC2 55 VFR(M) = 6.*TSR(M, 13)*TSR(M, 10)*TSR(M, 11)**3 BFRAC2 56 VCN(M) = TSR(M, 3)*TSR(M, 4) BFRAC2 57 INIT(M) = M BFRAC2 58 PRINT 1025, M, (TSR(M, I), I = 1, 14), VCR(M), VFR(M), VCN(M) BFRAC2 59 1025 FORMAT(* INITIALIZE BFRACT FOR M=*, I=1,14, VCR(M), VFR(M), VCN(M) BFRAC2 60 1, VCR, VFR, VCN=1*1P3E11.3) BFRAC2 61 1 CONTINUE BFRAC2 62 C *** INITIALIZE GENERAL ARRAYS - COS2TH, SIN2TH, ROT, CN, CL, FNUC BFRAC2 63 BFRAC2 64 LS = 1 BFRAC2 65 DO 5 I = 1, 6 BFRAC2 66 INIT(I) = 0 BFRAC2 67 DO 5 J = 1, 7 BFRAC2 68 FN(J) = 0. BFRAC2 69 5 CONTINUE BFRAC2 70 NANG1 = NANG - 1 BFRAC2 71 FNUC(1) = 0.707107/NANG1 BFRAC2 72 FNUC(NANG) = 0.292893 BFRAC2 73 COS2TH(1) = 1.0 BFRAC2 74 SIN2TH(1) = 0. BFRAC2 75 DO 10 NG = 2, NANG1 BFRAC2 76 FNUC(NG) = FNUC(1) BFRAC2 77 T1WOTH = 6.283185*FLOAT(NG-1)/FLOAT(NANG1) BFRAC2 78 COS2TH(NG) = COS(T1WOTH) BFRAC2 79 T2WOTH = 2.826161*FLOAT(NG-1)/FLOAT(NANG1) BFRAC2 80 10 CONTINUE BFRAC2 81 C *** INITIALIZE -Tsr- COEFFICIENTS FOR EACH MATERIAL BFRAC2 82 20 IF (INIT(M) .EQ. M) GO TO 25 BFRAC2 83 TSR(M, 3) = TSR(M, 3)**3 BFRAC2 84 VCR(M) = 8.*((1./ELMU+1.)/(EGSCM+ELMU/3.)) BFRAC2 85 VFR(M) = 6.*TSR(M, 13)*TSR(M, 10)*TSR(M, 11)**3 BFRAC2 86 VCN(M) = TSR(M, 3)*TSR(M, 4) BFRAC2 87 INIT(M) = M BFRAC2 88 PRINT 1025, M, (TSR(M, I), I = 1, 14), VCR(M), VFR(M), VCN(M) BFRAC2 89 1025 FORMAT(* INITIALIZE BFRACT FOR M=*, I=1,14, VCR(M), VFR(M), VCN(M) BFRAC2 90 1, VCR, VFR, VCN=1*1P3E11.3) BFRAC2 91 1 CONTINUE BFRAC2 92 C *** COMPUTATIONS BFRAC2 93 C ************** ************** BFRAC2 94 C ************** ************** BFRAC2 95 C ************** ************** BFRAC2 96 C ************** ************** BFRAC2 97 IF (NM .LT. 0.) GO TO 410 BFRAC2 98 IF (NT .EQ. 0.) GO TO 20 BFRAC2 99 FUG = FUG2D BFRAC2 100 VSG = VOLD*1.*N)/FUG/RHOS BFRAC2 101 VVO = VOLD/RHOS-VSO BFRAC2 102 DO = VCDL/(VCDL-VOLD)/RHOS BFRAC2 103 DOLD = RHOS/VOLD BFRAC2 104 PGE = P/(VSG0=FUG=DOLD) BFRAC2 105 R = ROT BFRAC2 106 CONTINUE BFRAC2 107 C *** SET VALUES FOR MULTIPLE LOOPS IN CASE OF LARGE STRAIN BFRAC2 108 C *** MULTIPLE LOOPS IF STRAIN CORRESPONDS TO A STRESS GREATER THAN BFRAC2 109
SUBROUTINE BFRACT (Continued)

C 0.33*TSR(M,5)
SDH=AMIN1(SXSEN,SYSEN,STTEN)
NL00P=MAX1(1.-4.*DV*EGSTCM/VS0/TSR(M,5)+0.5,DT0*20.*TSR(M,1)*(PSO)
1+SDH-TSR(M,2)+0.5)
NL00P=MING(NL00P,10)
PS=(EGSTCM/RHOS+EGSTCM+EEST)/(VS0+DV)-EGSTCM
IF(P>0. .AND. PS .GT. 0.) NL00P=1
DPJ=0.5*(ABS(TSR(M,1))+ABS(PSO))

100 DELV=DV/NL00P
IF(ABS(DV) .LT. 1.E-9) DELV=1.E-9
EXX=EXX1/NL00P*DV/DVO
EYY=EYY1/NL00P*DV/DVO
EXY=EXY1/NL00P*DV/DVO
VH=1./DOLD $ YT=Y
DE=(EEST-E)/NL00P
E1=E
TEMP1=1./RHOS+EGSTCM*E1/EQSTCM
DR=DELV/DVO*DROT
DT=DELV/DVO*DTO
A1=3.*TSR(M,1)**DT

BEGIN -DO- LOOP FOR EACH STEP IN STRAIN
DO 380 NL=1,NL00P
VH=VH+DELV
DH=1./VH
E1=E1+DE
TEMP0=TEMP1
TEMP1=1./RHOS+EGSTCM*E1/EQSTCM
SDH = AMIN1(SXSEN,SYSEN,STTEN)
VPO=0.
DO 120 NA=1,NANG1
VPO=VPO+CN(NA)*CL(NA)
TAUO=VFR(M)*VPO
VPO=-VCR(M)*VPO
*****
*****
*****
CONTINUE
*****
*****
******
PS=PG=PN=EQSTCM*E1/(VS0+DELV)-1.)
IF(P .LT. 0.) GO TO 130

C CRACK OPENING BASIS FOR PRESSURE ESTIMATE
PG=PSO+(DELV-TEMP1*1.-PSO/EQSTCM+VSO)/(VPO-1./EQSTCM*TEMP1)
IF (PG .LT. 0.) PSO=-(DELV-TEMP1*1.-PSO/EQSTCM+VSO)/(TEMP1/
1.VPO-1./EQSTCM*TEMP1)
PG=PSO+(DELV-TEMP1*1.-PSO/EQSTCM+VSO)/(VPO-1./EQSTCM*TEMP1)

380 GO TO 150
C NUCLEATION BASIS FOR PRESSURE ESTIMATE
IF (DELV .GT. 0.) PN=-PSO+2.*TSR(M,5)+2.*TSR(M,6)*ALOG(ABSCDELV/
1.VCR(M)/VCN(M)/DT/PSO))
PG=PSO+(DELV-TEMP1*1.-PSO/EQSTCM+VSO)/(VPO-1./EQSTCM*TEMP1)

120 DO 120 NA=1,NANG1
120 C COMPUTE STRESSES AT TIME(N-1) FOR EACH ANGULAR GROUP
STH(NANG)=STTEN+PSO
DO 170 NA=1,NANG1
170 C COMPUTE STRESSES AT TIME(N-1) FOR EACH ANGULAR GROUP
STH(NA)=(SXSEN+SYSEN)/2.+PSO+(SXSEN-SYSEN)/2.*COS2TH(NA)*COSR-
1*SIN2TH(NA)*SINR+TXYEN+(SIN2TH(NA)*COSR+COS2TH(NA)*SINR)
SINR=SIN(2.*R)
C0SR=COS(2.*R)
NC=0
ETAU=0.
IF (TAUO .LT. 0. .AND. ETAU .LT. 0.) ETAU=EXP(A1*AMIN1(0.,PSO+SDH-TSR(M,2)))*ETAU

1220 CONTINUE
C******** BEGIN ITERATION LOOP
C********
200 CONTINUE
NC=NC+1
SUBROUTINE BFRACT (Continued)

C *** COMPUTE PRESSURE
PA = EQSTCM * (TEMP1 / (VS0 + DVS) - 1.)
TAU = ETAU + EXP((AMIN1(2., A1/2. * (PA - PS0))))
F1 = AMAX1(0., AMIN1(1., 1. - TAU) / (1. - TSR(M, 12))))
VV = VH - F1 * (VS0 + DVS)

C *** COMPUTE DEVIATOR STRESS
RED = AMAX1(0., 1. - 4. * VV/DO)
RED1 = AMAX1(1. - SMF * VV/DO < 0.)

C *** COMPUTE DEVIATOR STRESS
RED = AMAX1(0., 1. - 4. * VV/DO)
RED1 = AMAX1(1. - SMF * VV/DO < 0.)

C *** COMPUTE DEVIATOR STRESS
RED = AMAX1(0., 1. - 4. * VV/DO)
RED1 = AMAX1(1. - SMF * VV/DO < 0.)

C *** COMPUTE DEVIATOR STRESS
RED = AMAX1(0., 1. - 4. * VV/DO)
RED1 = AMAX1(1. - SMF * VV/DO < 0.)

C *** COMPUTE DEVIATOR STRESS
RED = AMAX1(0., 1. - 4. * VV/DO)
RED1 = AMAX1(1. - SMF * VV/DO < 0.)

C *** COMPUTE DEVIATOR STRESS
RED = AMAX1(0., 1. - 4. * VV/DO)
RED1 = AMAX1(1. - SMF * VV/DO < 0.)

C *** COMPUTE DEVIATOR STRESS
RED = AMAX1(0., 1. - 4. * VV/DO)
RED1 = AMAX1(1. - SMF * VV/DO < 0.)

C *** COMPUTE DEVIATOR STRESS
RED = AMAX1(0., 1. - 4. * VV/DO)
RED1 = AMAX1(1. - SMF * VV/DO < 0.)

C *** COMPUTE DEVIATOR STRESS
RED = AMAX1(0., 1. - 4. * VV/DO)
RED1 = AMAX1(1. - SMF * VV/DO < 0.)

C *** COMPUTE DEVIATOR STRESS
RED = AMAX1(0., 1. - 4. * VV/DO)
RED1 = AMAX1(1. - SMF * VV/DO < 0.)

C *** COMPUTE DEVIATOR STRESS
RED = AMAX1(0., 1. - 4. * VV/DO)
RED1 = AMAX1(1. - SMF * VV/DO < 0.)

C *** COMPUTE DEVIATOR STRESS
RED = AMAX1(0., 1. - 4. * VV/DO)
RED1 = AMAX1(1. - SMF * VV/DO < 0.)

C *** COMPUTE DEVIATOR STRESS
RED = AMAX1(0., 1. - 4. * VV/DO)
RED1 = AMAX1(1. - SMF * VV/DO < 0.)

C *** COMPUTE DEVIATOR STRESS
RED = AMAX1(0., 1. - 4. * VV/DO)
RED1 = AMAX1(1. - SMF * VV/DO < 0.)

C *** COMPUTE DEVIATOR STRESS
RED = AMAX1(0., 1. - 4. * VV/DO)
RED1 = AMAX1(1. - SMF * VV/DO < 0.)
SUBROUTINE BFRACT (Continued)

PJ=PJA+EQSTCM*(VVA-DELV)/VSO.
IF (PJ .LT. 0.) PJ=AMAX1(PJ,PA)/2.
279 PJ=PA+SIGN(AMIN1(ABS(PJ-PA),DPJ),DELVA-DELV)
DVS=TEMP1/(1. +PJ/EGSTCM) - VSO
280 IF (NC .GT. 2) GO TO 285
IF (NC, EQ. 1) GO TO 290
281 IF (DELVA .LT. DELVB) 293,289
282 IF (DELVA .LT. DELVA .LT. DELVA) GO TO 287
283 IF (DELVA .LT. DELVA .GT. DELVA) GO TO 289
284 IF (DELVA .LT. DELVA .AND. DELVA .LT. DELVB) GO TO 287
285 IF (DELVA .LT. DELVA .AND. DELVA .GT. DELVA) GO TO 289
286 DELVC=DELVB
287 DVS=TEMP1/(1.+PJ/EQSTCM)-VSO
288 IF (NC .GT. 2) GO TO 285
289 IFCNC.EQ.1) G0 TO 290
290 DELVB=DELVA
291 DELVB=DELVA
292 DELVC=DELVB
293 DELCV=DELVA
294 DVS=DSVB
295 GO TO 200
C ENDING ROUTINE
300 CONTINUE
NT=0.
R=R+DR
DO 320 NA=1 ^ANG
301 STHW=STTE+PJ  *  GO TO 310
302 STHW=PJ+(SXXE+SYYE)/2.+ (SXXE-SYYE)/2.*(C0S2TH(NA)*COSR-SIN2TH(NA)*
1   SINR)+TXYE*(SIN2TH(NA)*C0SR+C0S2TH(NA)*SINR)
303 SAVG=(STH(NA)+STHW)/2,
304 STH(NA)=STHW
305 SCN = SAVG-TSR(M,9)*(PSO+PJ)/2.-TSR(M,5)
306 DN=0.
307 IF (SCN .LT. 0.) DN=TSR(M,4)*EXP(SCN/TSR(M,6))*DT*FNUC(NA)
308 CN0=CN(NA)
309 CN(NA)=CN(NA)*DN/DOLE+DN
310 IF (CN(NA) .EQ. 0.) GO TO 320
311 CL(NA)=(CN0*CL(NA)*EXP(A1*AMIN1(SAVG-TSR(M,2),0.))+
1   DN*TSR(M,3))/CN(NA)
312 NT=NT+CN(NA)
313 CONTINUE
314 NM=(VVA+(1.-FU1)* (VSO+DVSA))/DH
315 FU2D=FU1
316 PSO=PJ
317 IF (FU1 .LT. 0.01) GO TO 400
318 P=PSO
319 SXSE=SXXE
320 SYSE=SYYE
321 STTEN=STTE
322 TXYEN=TXYE
323 P=PJ
324 Y=YT
C *************
C END OF SUBCYCLING LOOP
C *************
VSO=VVA
VSO=VSA+DVSA
FU0=FU1
326 DOLD=DH
327 ROT=R
328 IF (LS .EQ. 2) G0 TO 500
329 RETURN
C END WITH SEPARATION
400 CONTINUE
329 SXSE=0.
330 SYSE=0.
331 STTEN=0.
332 TXYEN=0.
333 P=0.
334 Y=YT
C NM=ABS(NM)
RETURN
SUBROUTINE BFRACT (Continued)

410 CONTINUE
SXXEN=0.
SYYEN=0.
STTEN=0.
TXYEN=0.
EMU=1./VO-1.
P=(EQSTCM*EMU)*(1.-.5*EQSTGM*EMU)+EQSTGM*E*RHOS/VO
IF(P .LT.O.) P=0.
IF(LS .EQ. 2) GO TO 500
RETURN
C ** PROVISION FOR ABORT IN CASE OF ITERATION FAILURE
450 NTRY=NTRY+1
IF (NTRY .GE. 5) GO TO 460
DV=V0/RHOS-1./DOLD
NLOI.D=NLOOP
NLOI.P=MAX1(3.,-4.*2.*NTRY*DV*EQSTCM/VSO/TSR(M,5)+0.5,2.*NLOOP)
NPERI0=MIN1(MAX1(3.,4.*2.*NTRY*DV*EQSTCM/VSO/TSR(M,6),0.),10.*NLOLD)
GO TO 100
460 PRINT 1600,NN,KS,JS,SDH,P,DV,DELVA,DELVB
IF (NTRY .EQ. 5) STOP 22
NT=0.
R=R+DR
TAU=0.
DO 620 NA=1,NANG
IF (NA .LT. NANG) GO TO 607
STHW=STTE+PJ $ GO TO 610
607 STHW = PJ+(SXXE+SYYE)/2.+(SXXE-SYYE)/2.*(C0S2TH(NA)*COSR-SIN2TH(NA)*
1   SINR)+TXYE*(SIN2TH(NA)*COSR+C0S2TH(NA)*SINR)
610 SAVG=(STH(NA)+STHW)/2.
SCN=SAVG-TSR(M,9)*(PS0+PJ)/2.-TSR(M,5)
DN = 0.
IF (SCN .LT. 0.) DN=TSR(M,4)*EXP(SCN/TSR(M,6))*DT*FNUC(NA)
CNO=CN(NA)
CN(NA)=CN(NA)*DH/DOLD+DN
IF(CN(NA) .EQ. 0.) GO TO 620
CL(NA)=(CNO+CL(NA))*EXP(A1*AMIN1(SAVG-TSR(M,2),0.))+
1   DN*TSR(M,3)/CN(NA)
NT=NT+CN(NA)
620 TAU=TAU+CN(NA)*CL(NA)
TAU=VFR(M)*TAU
FU1=AMAX1(0.,AMIN1(1.,(1.-TAU)/(1.-TSR(M,12)))
FU2D=FU1
NM=VVAC/(1.-FU1)*(VSO+DVS)*DH
IF (FU1 .LT. 0.01) GO TO 400
PJ=PJ*FU1*(VSO+DVS)*DH
EEST=EEST+(PO-PJ)*DELV
SXXEN=SXXE
STTEN=STTE
TXYEN=TXYE
P=PJ
Y=Y+T
VVG=VVA
VSO=VSG+DVS
DOLD=DH
ROT=R
IF(LS .EQ. 2) GO TO 500
RETURN
C ** FINAL PRINTOUT
500 IZER0=1
IF (NT .EQ. 0.) GO TO 520
IZERO=2
CNSUM=0.
CRIT2=0.
CRIT3=0.
DO 610 NA=1,NANG
CL3(NA)=CL(NA)\((.3333333333)
CRIT2=CRIT2+CN(NA)*CL3(NA)**2
CNSUM=CNSUM+CN(NA)
610 CRIT3=CRIT3+CN(NA)*CL(NA)
IF(CNSUM .EQ. 0.) GO TO 520
CRIT2=3.1416*CRIT2

199
SUBROUTINE BFRACT (Concluded)

RAD=(CRIT3/CNSUM)**(1./3.)
FRAGRAD=0.
FRAGNUM=0.
IF(FU2D .EQ. 1.) GO TO 515
FRAGRAD=FRAGRAD+TSR(M,11)
FRAGNUM=FRAGNUM+TSR(M,11)*TSR(M,10)*(1.-FU2D)
515 CONTINUE
PRINT 1510,(CL3(I),I=1,5),RAD,CRIT2,ROT,FU2D,KS,JS,(CN(I),I=1,5),
CNSUM,FRAGRAD,FRAGNUM
520 CONTINUE
RETURN
1510 FORMAT(13H0CELL CL = 1P4E10.3,2X,E10.3,11H CL-AVG = E10.3,
1 12H PI*N*R**2=0PF6.0,6H R0T=F6.0,5H FU=F6.4/213,7H CN =
2 1P4E10.3,2X,E10.3,10H CN-TOT =E10.3,16H FRAGMENT RAD.=E10.3,
3 6H NO.=E10.3)
1600 FORMAT(32H ITERATION FAILURE IN BFRACT, N=I5,4H, K=I3,4H, J=I3,
1 1P5E12.3/5X,1P3E12.3)
END
SUBROUTINE CAP1

SUBROUTINE CAP1( LS, IN, N, M, N1, DH, DOR, EG, EX, EZ, EXY, SY, SY, CAP1 2
 1, S2, SXY, ZEV, K, J, TENV )  CAP1 3
 C CAP1 - WRITTEN BY L. SEAMAN - INSERTED INTO THE CPS CODE 10 - 78  CAP1 4
 C *** DEFINITION OF INDICATOR - 1H-  CAP1 5
 C 6 - MOHR-COULOMB SURFACE  CAP1 6
 C 7 - CAP SURFACE  CAP1 7
 C 8 - CAP AND MOHR-COULOMB SURFACE  CAP1 8
 C 9 - CONSOLIDATED  CAP1 9
 C 10 - SEPARATION  CAP1 10
 C
 INTEGER DBUG1, DBUG2  CAP1 11
 REAL MUP, MUP2  CAP1 12
 DIMENSION INIT(4), AMC(4), AMC2(4), AMC3(4), AMC4(4), AMC5(4),  CAP1 13
 1, AMC6(4), AK(4), AK2(4), MUP(4), MUP2(4), NREG(4), DAMG(4),  CAP1 14
 2, SCRIT(4), W2(4), AKSOL(4),  CAP1 15
 DIMENSION PA(5), AJ(5), DL(2)  CAP1 16
 COMMON /EIGF/ EOSTC(6), EQSTD(6), EQSTE(6), EQSTO(6), EQSTH(6), EQSTN(6)  CAP1 17
 1, EQSTS(6), ROH(6), RHOS(6), YC(6), YD(6), YAD(6), MU(6), ESC(6, 20), CLIN, CQSO,  CAP1 18
 2, TRIQ, AMAT(6, 4), SP(6), G2(6), PMIN(6)  CAP1 19
 COMMON /POR/ P0RA(6, 5), PORBCS, PJ(5)  CAP1 20
 DATA INIT/4/O/  CAP1 21
 GG(X,Y) = MUP(M) + MUP2(M) * AMIN1(AMC1(M), X)  CAP1 22
 BKK(X,Y) = AMIN1(AMC1(M), X)  CAP1 23
 IF (AKSOL(M) .LT. AK(M)) AKSOL(M) = 2.*AK(M)  CAP1 24
 PRINT 1021, IN  CAP1 25
 IF (LS), 30, 30  CAP1 26
 C *** READ AND INITIALIZE MATERIAL ARRAYS.  CAP1 27
 C ***  CAP1 28
 C *****  CAP1 29
 CAP1 30
 30 IF (INIT(M) .EQ. M) GO TO 50  CAP1 31
 INIT(M) = M  CAP1 32
 READ (IN, 1020) A1, A2, AK(M), A3, A4, AK2(M), A5, A6, MUP(M), A7, A8, MUP2(M)  CAP1 33
 PRINT 1040, A1, A2, AK(M), A3, A4, AK2(M), A5, A6, MUP(M), A7, A8, MUP2(M)  CAP1 34
 PRINT 1021, IN  CAP1 35
 READ (IN, 1022) A1, A2, AMC1(M), AMC2(M), AMC3(M), AJ10, EN  CAP1 36
 PRINT 1042, A1, A2, AMC1(M), AMC2(M), AMC3(M), AJ10, EN  CAP1 37
 IF (AMC4(M) = - (AMC1(M) + AMC2(M) * EXP(AJ10/AMC3(M))) * EXP(-EN))  CAP1 38
 AMCS(M) = AJ10/EN  CAP1 39
 AMCS(M) = AJ10/EN  CAP1 40
 READ (IN, 1020) A1, A2, SCRIT(M), A3, A4, DAMG(M), A5, AK6, AKSOL(M)  CAP1 41
 IF (AKSOL(M) .LT. AK(M)) AKSOL(M) = 2.*AK(M)  CAP1 42
 PRINT 1040, A1, A2, SCRIT(M), A3, A4, DAMG(M), A5, AK6, AKSOL(M)  CAP1 43
 PRINT 1021, IN  CAP1 44
 READ (IN, 1022) A1, A2, EVP(M, I), I = 1, 5  CAP1 45
 PRINT 1042, A1, A2, EVP(M, I), I = 1, 5  CAP1 46
 PRINT 1021, IN  CAP1 47
 READ (IN, 1024) A1, A2, NREG(M), A3, A4, NP, NREG(M), A5, A6, P1, A7, A8, W2(M)  CAP1 48
 PRINT 1044, A1, A2, NREG(M), A3, A4, NP, NREG(M), A5, A6, P1, A7, A8, W2(M)  CAP1 49
 PRINT 1021, IN  CAP1 50
 C *** COMPUTATION OF PARAMETERS ON HYDROSTAT  CAP1 51
 PORB(M, 1) = P1  CAP1 52
 PORR(M, 1) = P1  CAP1 53
 PORC(M, 1) = P1  CAP1 54
 NP = NREG(M, 4)  CAP1 55
 DO 15 NG = 1, NP  CAP1 56
 READ (IN, 1020) A1, A2, P1, A3, A4, DEXP  CAP1 57
 PRINT 1040, A1, A2, P1, A3, A4, DEXP  CAP1 58
 PRINT 1021, IN  CAP1 59
 DE = EVP(M, NQ+1) - EVP(M, NQ)  CAP1 60
 DP = 4. * DEXP / DE  CAP1 61
 FORB(M, NQ+1) = P1 - EVP(M, NQ) / DE * (P1 - EVP(M, NQ+1))  CAP1 62
 FORR(M, NQ+1) = P1 - EVP(M, NQ+1) / DE * (P1 - EVP(M, NQ))  CAP1 63
 P1 = P1 + DP * (EVP(M, NQ) - EVP(M, NQ+1))  CAP1 64
 NMAX = 30  CAP1 65
 FOR1 = 60.  CAP1 66
 FOR2 = 300.  CAP1 67
 DFCR = 1.5  CAP1 68
 15 P1 = P1  CAP1 69
 15 P1 = P1  CAP1 70
 15 P1 = P1  CAP1 71
 15 P1 = P1  CAP1 72
 15 P1 = P1  CAP1 73
 15 P1 = P1  CAP1 74
 C *** SET LPATH=0 FOR CONSTANT VOLUME ON M-G, =1 FOR NORMALITY  CAP1 75
 LPATH = 0  CAP1 76

201
SUBROUTINE CAP1 (Continued)

PRINT 1004,FCR1,FCR2,DFCR1,NMAX,LPATH         CAP1  77
1004 FORMAT (* ACCURACY AND SUBCYCLING CRITERIA FCR1,FCR2 =*P2E10.3, CAP1  78
        1 * DFCR1,NMAX=** E10.3,14,/* MOHR-COULOMB PATH, LPATH=**I2, +CAP1  79
        2 * -0 - FOR CONSTANT VOLUME, -1 - FOR NORMALITY*)         CAP1  80
C
DBUG1=0                 CAP1  81
DBUG2=1                 CAP1  82
 IF(DBUG2.EQ.0) GO TO 41    CAP1  83
 DO 31 I=1,5            CAP1  84
  PA(I)=(1+1)*PORA(M,1)/2.  CAP1  85
  AJ(I)=0.             CAP1  86
31                        CAP1  87
 PRINT 3000, (PA(I),I=1,5)     CAP1  88
3000 FORMAT (*1MOHR-COULOMB AND CAP COORDINATES/20X,*J1*,9X,*M-C*, CAP1  89
  9X,*J2 ON CAPS CORRESPONDING TO PR VALUES*/32X,*J2*,1P5E12.3)  CAP1  90
I 1=0                CAP1  91
AJ2=0.               CAP1  92
 PRINT 3001, I I  CAP1  93
3001 FORMAT (I 10,1P7E12.3) CAP1  94
DP=0.2*PORA(M,1)        CAP1  95
 DO 32 I=1,5            CAP1  96
  PA(I)=PA(I)**2        CAP1  97
32                        CAP1  98
 IF (DP .LE. PA(I)) GO TO 38  CAP1  99
 DO 40 I 1=1,50         CAP1 100
  AJ1=AJ1+I**DP       CAP1 101
  AMC1(M)+AMC2(M)=EXP(AJ1/MCM(M))+AMC4(M)*EXP(AJ1/MCM(M))     CAP1 102
3001 FORMAT (I 10,1P7E12.3) CAP1 103
 DO 35 JJ=1,5          CAP1 104
  AJ(JJ)=0.           CAP1 105
35                        CAP1 106
 IF (PP .GE. PA(JJ)) GO TO 35  CAP1 107
35                        CAP1 108
 CONTINUE         CAP1 109
38                        CAP1 110
 CONTINUE         CAP1 111
40                        CAP1 112
 CONTINUE         CAP1 113
41                        CAP1 114
 RETURN            CAP1 115
C
C **********       CAP1 116
C COMPUTATION OF STRESS                                     CAP1 117
C **********       CAP1 118
50  AJ10=SX+SY+SZ                               CAP1 119
 PO=AJ10/3.                                    CAP1 120
 NRE=0                                          CAP1 121
 DOLD=DORQ                                     CAP1 122
 EPRAT1=EPRAT2=0.25                            CAP1 123
 RR=1                                           CAP1 124
 RSUM=0.                                        CAP1 125
 IF (DBUG1.EQ.1) PRINT 1052,N,K,J,IH,SX,SY,SZ,SXY,EX,EY,EXY,TEVP, CAP1 126
 1  ZEVP,DH,DORG                                CAP1 127
1052 FORMAT (OBEGIN N,K,J,IH=«4I4,* SX,SY,SZ,SXY=«1P4E10.3, CAP1 128
  1  * EX,EY,EXY=*3E10.3/10X,* TEVP,ZEVP=*2E10.3,* DH,DORG=*0P2F10.6 CAP1 129
  2 )                                           CAP1 130
 ZEVT=ALOG(DORQ/DH)                      CAP1 131
 EZ=ZEVT+1.*EX-EY                      CAP1 132
 DEVZ=ZEVT                                CAP1 133
 IF (IH .NE. 10) GO TO 80                  CAP1 134
 TEVP=TEVP+ZEVT                        CAP1 135
 IF (TEVP.GT.0.) GO TO 580                CAP1 136
 DE3=(ZEVT-TEVP)/3.                      CAP1 137
 EX=EX-DE3                                 CAP1 138
 EZ=EZ-DE3                                 CAP1 139
 ZEVT=TEVP                            CAP1 140
 TEVP=1.                                  CAP1 141
 IH=5                                    CAP1 142
80 CONTINUE                                    CAP1 143
AJ20=SQRT(((SX-P0)**2+(SY-P0)**2+(SZ-P0)**2)/2.+SXY**2) CAP1 144
 EV=DEVZ/3.                                 CAP1 145
 CONTINUE                                    CAP1 146
C
C COMPUTE STRESSES ON ELASTIC BASIS AND TEST FOR YIELDING.     CAP1 147
BG=GH(AJ20)                                    CAP1 148
 SXT=(SX-P0)+2.*BG*(EX-EV)                    CAP1 149
 SYT=(SY-P0)+2.*BG*(EY-EV)                    CAP1 150
 SZT=(SZ-P0)+2.*BG*(EZ-EV)                    CAP1 151
SUBROUTINE CAP1 (Continued)

SXY = SXY + 2. * BG * EXY
AJ2 = SORT((SXT**2 + SYT**2 + SZT**2) / 2. + SXY**2)
ZEIT = (Aj2 - Aj10) / (2. * BG)
IF (MUP2(M) \ .EQ. 0.) GO TO 95

95 BG = GG(AJ2T, AJ20)
BK = (AK(M) + AK2(M) * AJ10) * (1. + 1.5 * AK2(M) * ZEVT)
AJ1 = AMIN1(AK(S0L(M), AMAX1(BK, AK(M)))
AJ10 = AJ10 + 3. * BK * ZEVT
D = DH
PAT = AJ1T / 3.
PT = PAT
F10 = AJ20 - (AMC1(M) + AMC2(M) * EXP(AJ10 / AMC3(M)) + AMC4(M) * EXP(AJ10 / AMC5(M)))
AMC = AMC1(M)
IF (AJ1LT. LT. .5 * (AMC3(M) + AMC5(M)) ) GOTO 110
AMC = AJ1T
IF (AJ1LT. 8T. AMC6(M) ) GOTO 110
AMC = AMC1(M) + AMC2(M) * EXP(AJ10 / AMC3(M)) + AMC4(M) * EXP(AJ10 / AMC5(M))
F10 = AJ20 - (AMC1(M) + AMC2(M) * EXP(AJ10 / AMC3(M)) + AMC4(M) * EXP(AJ10 / AMC5(M))
NINC = 0
RR = 1. / AMIN0(600, NINC)

105 10 = 1P4E10.3, 205, 210
145 1 = 4E10.3, / 10X, *FIT, F1T, F10, F20 = 4E10.3, * PT = 1E10.3)

C TEST FOR PURELY ELASTIC CASE.
IF (F1T .LE. 0. AND. F2T .LE. 0. ) GO TO 500

C **********   **********
C BEGIN SUBCYCLING LOOP OVER EACH STRAIN INCREMENT.
C **********   **********
NINC = MAX1(1., FCR1 * ABS(AJ1T - AJ10) / BK, FCR2 * (AJ2T - AJ20) / BG)
IF (NINC .GT. 40.) GOTO 110

C SET INITIAL STRAIN INCREMENT.
NINC = NINC + 1
D = DOLD * EXP(-ZEVT * RR)
DEV = ZEVT * RR
AJ10 = SX + SY + SZ
PO = AJ10 / 3.
AJ20 = SORT((SXT-P0)**2 + (SY-P0)**2 + (SZ-P0)**2) / 2. + SXY**2

C COMPUTE STRESS INVARIANTS.
BK = (AK(M) + AK2(M) * AJ10) * (1. + 1.5 * AK2(M) * DEV) * (1. + AK2(M) * DEV)
AJ1 = AMIN1(AK(S0L(M), AMAX1(BK, AK(M)))
AJ10 = AJ10 + 3. * BK * DEV
PAT = AJ1T / 3.
EV = DEV / 3.
BG = GH(AJ20)
SXT = (SXT-P0) + 2. * BG * (EX*RR - EV)
SYT = (SY-P0) + 2. * BG * (EY*RR - EV)
SZT = (SZ-P0) + 2. * BG * (EZ*RR - EV)
SXY = SXY + 2. * BG * EXY
AJ2 = SORT((SXT**2 + SYT**2 + SZT**2) / 2. + SXY**2)
IF (MUP2(M) .EQ. 0.) GO TO 205
DEIT = (AJ2T - AJ20) / (2. * BG)
AJ2T = (MUP(M) / MUP2(M)) * EXP(2. * MUP2(M) * DEIT)
1 - MUP(M) / MUP2(M)

205 BG = GG(AJ2T, AJ20)
AJ1 = AMIN1(AJ2T, AJ20 + 2. * BG * DEIT)
IF (DBUG1 .EQ. 1) PRINT 1205, AJ10, AJ1T, AJ20, AJ2T, BK, BG, DEVT, DEIT,
1 RR, RSUM, D, DOLD
SUBROUTINE CAP1 (Continued)

C EVALUATE F1 AND F2 FROM ELASTIC STRESSES.
AMC=AMC1(M)
IF(AJ1T.LT.-5.*(AMC3(M)+AMC5(M)))GO TO 215
AMC=-AJ1T
IF(AJ1T.GT.AMC6(M))Q0TQ215
AMC=AMC1(M)+AMC2(M)*EXP(AJ1T/AMC3(M))+AMC4(M)*EXP(AJ1T/AMC5(M))
F1T=AJ2T-AMC
CALL CAPPRCPT.ZEVP.M.DOLD,IHH.YM)

C COMPUTE F2 FROM ELASTIC STRESSES AND PREVIOUS PLASTIC STRAIN.
245 F2T=AJ2T**2/W2(M)+AJ1T**2/9.-PT**2
DZEP=AMIN1(-1.E-S.DEVT)
DZEP=ZEVP+DZEP
CALL CAPPRCPZ.ZEP.M.D,IHH.YM)
DPDE=3.*(PZ**2-PT**2)/DZEP
NQ = 0
IF (DBUG1 .EQ. 1)
1PRINT 1270,F1T,F2T,PT,PZ,ZEP,DZEP,RR,EPRAT1,EPRAT2
1270 FORMAT (* 270 - F1T,F2T,PT,PZ=1P4E10.3, ZEP,DZEP=*2E10.3, /
/10X, RR,EPRAT1,EPRAT2=0P3F10.6)
IF (FIT .LE. 0. .AND. F2T .LE. 0.) GO TO 500
AJ1=AJ10+EPRAT1*(AJ1T-AJ10)
AJ21=AJ20+EPRAT2*(AJ2T-AJ20)
C ********

C COMPUTATION OF YIELDING PROCESS
NQ=1
BK=BKH(AJ11)
BG=QH(AJ21)

C YIELD ON MOHR-COULOMB SURFACE.
IFCFIT.LT.0.) GO TO 350
IFCZEVT.GT.0.) GO TO 550
IF (F2T .GT. 0.) GO TO 400
310 AJ1=AMIN1(AJ1T,AMC6(M))
NC=0
AJ2B=AJ20
320 NC=NC+1
TAU2=AMC2(M)*EXP(AJ1/AMC3(M))
TAU3=AMC4(M)*EXP(AJ1/AMC5(M))
AJ2=AMC1(M)+TAU2+TAU3
IF (LPATH .EG. 0) GO TO 330
DJ2=AJ2-AJ2B
IF (NC .GE. 10) GO TO 700
IF (ABS(DJ2) .LT. DFCR1 .AND. NC .GT. 1) GO TO 330
XI=TAU2/AMC3(M)+TAU3/AMC5(M)
BK = BKH(AJ11)
BG=QH(AJ21)
XIB=AMC2(M)*EXP((AJ1+AJ10)/(2.*AMC3(M)))/AMC3(M)
1 +AMC4(M)*EXP((AJ1+AJ10)/(2.*AMC5(M)))/AMC5(M)
DJ=(AJ2T-AJ2+(AJ1T-AJ1)*BG/(9.*BK*XIB))/(XI+BG/(9.*BK*XIB))
IF (DBUG1 .EQ. 1) PRINT 1320,NC,AJ1,AJ2,DJ2,DJ,TAU2,TAU3,XI,BK,BG
1320 FORMAT (* M-C NC=I2, J1, J2, DJ2, DJ = 1P4E12.5,* K1, K2,* /0X, K1=0P5F10.3)
AJ2B=AJ2
AJ1=AJ1+DJ
GO TO 320
330 F21=AJ2**2/W2(M)+AJ1**2/9.-PT**2
IF (DBUG1 .EQ. 1) PRINT 1330,NC,AJ1,AJ2,DJ2,DJ,TAU2,TAU3,XI,BK,BG
1330 FORMAT ( M-C END, J1, J2, DJ2, DJ=1P4E12.5,* K1, K2,*, /0X, K1=0P5F10.3)
IF (F21 .GT. 0.) GO TO 410
IH=6
GO TO 600
C YIELD ON THE CAP SURFACE

C
350 BB=0.
AJ10P=AJ10
AJ20P=AJ20
IF (IH .EQ. 7 .OR. IH .EQ. 8) GO TO 353
AJ10P=SIGN(PT*AJ1T/SQRT(AJ1T**2/9.+AJ2T**2/W2(M)),AJ1T)
AJ20P=SIGN((PZ-PT)/(AJ1T-AJ10P),AJ20P)
ZEP=ZEVP+AMIN1(0.,DEV-(AJ10P-AJ10)/BK)
CALL CAPPRPZ(ZEP,M,D,PK,PK)
353 IF (ABS(AJ1T-AJ10P) .GE. 1.) BB=(PZ-PT)/(AJ1T-AJ10P)

204
SUBROUTINE CAP1 (Continued)

IH=7
ILO=1
IHI=2
C FIRST ESTIMATE OF J1
NCAP=1
INT=1LO
AJ1=AJ10*P*(1.+BB/(PT+BB*AJ10P)*(3.+DEVT*BBK(AJ10P)))
C COMPUTATION OF J2 AND ERROR DLA
BK=BBK(AJ1, AJ10).
DZEP=AMIN1(0., DEVT-(AJ1-AJ10)/(3.*BK))
CALL CAPPR(PR, ZEV, DZEP, M, D, IH, YM)
AJ1=AMAX1(AJ1, 3.*PR)
PJ=AMIN1(ABS(AJ1), ABS(AJ1-6.*PR))/3.
AJ2=0.
IF (PR=2-PJ)**2 .GT. 1.)
1AJ2=SIGN(SORT(W2(M)**2/(PR**2-PJ**2)), AJ1-3.*PR)
BG=BB(ABS(AJ2), AJ20)
IF (1H .EQ. 9 .AND. NCAP .EQ. 2) GO TO 480
IF (NCAP=2) 358,360,370
C SECOND ESTIMATE OF J1
AJ1=AJ1
IF (DBG1 .EQ. 1)
1PRINT 1365, NCAP, AJ1, AJ2, DLA, BG, DEIP, PJ, PR
1365 FORMAT (* 365 NCAP*13, * J1, J2=1P2E13.6, * DLA, BG, DEIP, PJ, PR=*
1 5E10.3)
IF (2.*ABS(DLA)-BG .LT. AFCH1) GO TO 390
IF (NCAP .GE. 30) GO TO 700
IF (NCAP=2) 358,360,370
C MOST TENSILE ESTIMATE OF J1
IF (ABS(AJ1) .LT. 1E4) AJ1=-3.*SQRT(AMAX1(0., PT**2-AJ2T**2/ W2(M)))
12W(M))
DZ2=2.*DZEP*BG
IF (ABS(AJ1) .GT. 1E4) AJ1=-9./W2(M)*ABS(AJ2)/AJ1+DJ2 + AJ1
GO TO 382
360 INT=1LO
IF (D(L(ILO))=D LA .LT. 0.) GO TO 375
INT=1HI
IF (AMAX1(D(L(ILO)), DLA) .LT. 0.) GO TO 366
C MOST COMPRESSION ESTIMATE OF J1
AJ1=0.5*(AJ1+AMAX1(AJ1T, AJ10))
GO TO 362
C MOST COMPRESSIVE ESTIMATE OF J1
AJ1=3.*PT
IF (ABS(DEVT) .LT. 1.E-7) GO TO 370
DEVZP=DEV=(PT-AJ10/3.)/BK(AJ10, 3.*PT)
DEVP=DEV+(PT-AJ10/3.)/BK(AJ10, 3.*PT)
IF (DBG1 .EQ. 1)
GO TO 382
370 INT=1HI
IF (MOD(NCAP,5) .EQ. 0) AJ1=AJ1+AJA+AJ(AINT))/3.
C STORAGE OF RESULTS OF PREVIOUS ITERATIONS
INT=1HI
IF (NCAP .EQ. 2) GO TO 382
IF ((0) .EQ. 1) GO TO 385
IF (D(L(ILO)) .GT. 0. .OR. (DL(IHI)) .GT. 0. .AND. D LA .GT. 0.))
1 D(L(ILO)) .AND. D(L(ILO)) .LT. 0.) GO TO 386
IF (D(L(ILO)) .GT. 0. .OR. (DL(IHI)) .GT. 0. .AND. D LA .GT. 0.))
1 GO TO 382
INT=1LO
382 DL(I)D=DLA
A(I,INT)=AJA
IF (NCAP .EQ. 1) GO TO 385
IF (D(L(IHI)) .GT. D(L(ILO))) GO TO 385
INT=1HI
IHI=1LO
1LO=INT
SUBROUTINE CAP1 (Continued)

385 CONTINUE
NCAP = NCAP + 1
IF (DBG1 .EQ. 1) GO TO 1385
1PRINT 1385, NCAP, IH, IL, AJ(1), AJ(2), DL(1), DL(2)
GO TO 355
C CHECK FOR CONVERGENCE TO POINT ABOVE THE M-C CURVE
390 F1 = AJ2 - (AMC1(M) + AMC2(M)) * EXP(AJ1/M) - AMC4(M) * EXP(AJ1/1)
AJ5 = AMC5(M))
AJ2 = 2. * DL + AJ5
IF (ABS(AJ5) .GE. 1.0) AJ1 = -9. / W2(M) * ABS(AJ2) / AJ1 + DJ
AJ2 = AMC1(0.0, AJ2 + DJ)
IF (F11 .LT. DFCR1) GO TO 600
IF (DBG1 .EQ. 1) GO TO 390
1PRINT 1390, NCAP, AJ1, AJ2, F11
1390 FORMAT (* 1390 SKIP TO JOINT, NCAP=*, I3, *, J1, J2, F11=*, 1P3E10.3)
GO TO 418
C YIELD AT JOINT OF CAP AND MOHR-COULOMB
C 400 IF (IH .NE. 8) GO TO 350
CRIT = (AJ1 - TAU2) / W2(M) * BK * AJ10 + CJ
CRIT2 = (AJ1 - TAU2) / W2(M) * BK * AJ20
IF (DBG1 .EQ. 1) GO TO 401
1PRINT 1400, CRIT, CRIT2, AJ10, AJ20
1400 FORMAT (* 1400 CRIT, CRIT2=*, 1P2E10.3, *, AJ10, AJ20=*, 1P2E10.3)
 IF (CRIT .GT. CRIT2) GO TO 350
410 DF1 = AMC5(M) + AMC6(M) * EXP(AJ1/M) - AMC5(M) + AMC6(M)
DF1 = AMC5(M)
DF21 = 2. * AJ1
DF22 = 2. * AJ2 / W2(M)
DET = 9. * BK * (DF11 * DF22 - DF21) - DPDE
AJ1 = DPDE / (AJ1 - AJ10) / DET + AJ10
AJ2 = DPDE / (AJ1 - AJ10) / DET + AJ20
AJ1 = AMC1(0.0, AJ1)
AJ2 = AMC1(0.0, AJ2)
IF (DBG1 .EQ. 1) GO TO 412
1PRINT 1900, AJ1, AJ2, AJ10, AJ20, DF1, DF21, DF22, DPDE
1 * DF11, DF21, DF22, DPDE =*, 1P4E10.3)
DEP = DET - (AJ1 - AJ10) / (3. * BK)
418 NC = 0
420 NC = NC + 1
IH = 8
TAU2 = TAU3 = 0.
IF (AJ1 .LT. -10. * AMC1(M)) GO TO 430
AJ1 = AJ1
IF (AJ1 .GT. TAU1 / AMC3(M)) AJ1 = AMC6(M)
TAU2 = AMC2(M) / EXP(AJ1 / AMC3(M))
TAU3 = AMC4(M) / EXP(AJ1 / AMC5(M))
430 AJ2 = AMC1(0.0, AJ2)
XI1 = TAU2 / AMC3(M) + TAU3 / AMC5(M)
XI2 = TAU2 / AMC3(M) + TAU3 / AMC5(M)
DZEP = DEPT - (AJ1 - AJ10) / (3. * BK)
ZEP = 2 * VP + 1
CALL CAPPRP(PR, ZEP, M, D, IH, YM)
IF (IH .EQ. 9) GO TO 480
YM = YM / (3. * BK)
440 IF (ABS(SQRT(AJ1 * W2(M) * AJ2 =aji)**2 / W2(M) + VR) .LT. DFCR1) GO TO 600
440 FORMAT (* 440 AJ1, AJ2 =aji)**2 / W2(M) * AJ1, DJ, CRIT, PR, AA, BB, CRC
450 IF (DBG1 .EQ. 1) GO TO 447
1PRINT 1480, AJ1, DJ, CRIT, PR, AA, BB, CRC
450 FORMAT (* 450 AJ1, DJ, CRIT, PR =aji)**2 / W2(M) * AJ2 =aji)**2 / W2(M) + VR, 1
1 * AJ1, AJ2 =aji)**2 / W2(M) + VR =aji)**2 / W2(M) + VR
1 * AA, BB, CRC =aji)**2 / W2(M) + VR
460 IF (NC = GE. 5) GO TO 700
CAP1 377
CAP1 378
CAP1 379
CAP1 380
CAP1 381
CAP1 382
CAP1 383
CAP1 384
CAP1 385
CAP1 386
CAP1 387
CAP1 388
CAP1 389
CAP1 390
CAP1 391
CAP1 392
CAP1 393
CAP1 394
CAP1 395
CAP1 396
CAP1 397
CAP1 398
CAP1 399
CAP1 400
CAP1 401
CAP1 402
CAP1 403
CAP1 404
CAP1 405
CAP1 406
CAP1 407
CAP1 408
CAP1 409
CAP1 410
CAP1 411
CAP1 412
CAP1 413
CAP1 414
CAP1 415
CAP1 416
CAP1 417
CAP1 418
CAP1 419
CAP1 420
CAP1 421
CAP1 422
CAP1 423
CAP1 424
CAP1 425
CAP1 426
CAP1 427
CAP1 428
CAP1 429
CAP1 430
CAP1 431
CAP1 432
CAP1 433
CAP1 434
CAP1 435
CAP1 436
CAP1 437
CAP1 438
CAP1 439
CAP1 440
CAP1 441
CAP1 442
CAP1 443
CAP1 444
CAP1 445
CAP1 446
CAP1 447
CAP1 448
CAP1 449
CAP1 450
CAP1 451

206
SUBROUTINE CAP1 (Continued)

IF(NMC.GT.1) GO TO 420
BK=BKK(AJ1,AJ10)
BG=GO(AJ2,AJ20)
GO TO 420

480 AJ1=3.*PR
AJ2=AMIN1(AJ2,AMC1(M))
GO TO 600

480
AJ1=3.*PR
AJ2=AMIN1(AJ2,AMC1(M))
GO TO 600

C ********** CAP1 **********

500 AJ1=AJ1T
P=AJ1/3.
SX=SX-P0+2.*BG*EX*RR/3.+P
SY=SY-P0+2.*BG*(EY*RR/3.)*P
SZ=AJ1-EX-SY
SXY=SXY+2.*BG*EXY*RR
AJ2=AJ2T
IH=5
GO TO 630

C COMPLETION OF STRESS CALCULATION FOR ELASTIC CASE

500 AJ1=AJ1T
P=AJ1/3.
SX=SX-P0+2.*BG*EX*RR/3.+P
SY=SY-P0+2.*BG*(EY*RR/3.)*P
SZ=AJ1-EX-SY
SXY=SXY+2.*BG*EXY*RR
AJ2=AJ2T
IH=5
GO TO 630

C TENSILE FAILURE ON THE MOHR-COULOMB SURFACE

550 AJ1=AMIN1(AJ1T,AMC6(M))
AJ2=AMC1(M)+AMC2(M)*EXP(AJ1/AMC3(M))+AMC4(M)*EXP(AJ1/AMC5(M))
P=AJ1/3.
SX=SX-P0+2.*BG*EX*RR/3.+P
SY=SY-P0+2.*BG*(EY*RR/3.)*P
SZ=AJ1-EX-SY
SXY=SXY+2.*BG*EXY*RR
AJ2T=SQRT(0.5*(SX**2+SY**2+SZ**2)+SXY**2)
FAC=AJ2/AMAX1(1.,AJ2T)
SX=SXD+FAC+P
SY=SYD*FAC+P
SZ = SZD+FAC+P
SXY=SXYD*FAC
DEPT=SQRT((EX**2+EY**2+EZ**2)/2.+EXY**2)*RR*(AJ2T-AJ2)/
(1+AJ2T+1.)
IF (TEVP .EQ. -1.) TEVP=0.
TEVP=TEVP+DEPT
IH=6
SMAX=AMAX1((SX+SY+SQRT(4.*SXY**2+(SX-SY)**2))/2.,SZ)
IF (TEVP .GT. DAMQ(M) .AND. SMAX .GE. SCRIT(M) .AND. AJ1 .GE. 0.)
GO TO 570
ZEVP=ZEVP+AMIN1(0,,ZEVT*RR-(AJ1-AJ10)/BK/3.)
GO TO 630

570 CONTINUE

ENU=(3.-2.*BG/BK)/(6.+2.*BQ/BK)
EMOD = 2.*n .+ENU)*BG
DEX=(SX-ENU*(SY+SZ))/EMOD
DEY=(SY-ENU*(SX+SZ))/EMOD
DEZ=(SZ-ENU*(SX+SY))/EMOD
DEXY=SXY/2./BG
DEPF=SQRT((DEX**2+DEY**2+DEZ**2)/2.+DEXY**2)
IF (TEVP = EQ. -1.) TEVP=0.
TEVP=TEVP+DEPF+(1.-RSUM)/RR*DEPT

580 SX=0.
SY=0.
SZ=0.
SXY=0.
AJ1=0.
AJ2=0.
D=DH
RR=1.-RSUM
ZEVP=ZEVP+AMIN1(0.,ZEVT*RR-(AJ1-AJ10)/AK(M)/3.)
IF (IH .NE. 10) PRINT 1590,K,J,N
1590 FORMAT (22H SEPARATION AT CELL K=13,4,H, J=13,9H ON CYCLE14)
IH=10
GO TO 630

C compute stresses at end of iterations

600 CONTINUE

AJ2=AMAX1(0.,AJ2)
DEP=DEVT-(AJ1-AJ10)/(3.*BK)
ZEVP=ZEVP+AMIN1(0.,DEP)
P=AJ1/3.
EV=(EX+EY+EZ)/3.
SX= SX-P0+2.*BG*(EX-EV)*RR

207
SUBROUTINE CAP1 (Concluded)

SYD=SY-PO+2.*BG*(EY-EV)*RR
CAP1 527
SZD=SZ-PO+2.*BG*(EZ-EV)*RR
CAP1 528
SXYD=SXY+2.*BG*EXY*RR
CAP1 529
AJ2T=SQRT(0.5*(SXD**2+SYD**2+SZD**2)+SXYD**2)
CAP1 530
FACEJ2/AMAX1(1.,AJ2T)
CAP1 531
SX=SXD*FACEJ2+P
CAP1 532
SY=SYD*FACEJ2+P
CAP1 533
SZ=SZD*FACEJ2+P
CAP1 534
SXY=SXYD*FACEJ2
CAP1 535
C  ********
C  PREPARE FOR NEXT SUBCYCLE
C  ********
630 CONTINUE
CAP1 536
1 DOLED,1H,ZEVT,ZEVP,TEVP
CAP1 537
1630 FORMAT (* -FINAL--K,J=213,* RR,RSUM**2F8.5,* SX,SY,SZ,SXY=*
CAP1 538
1 1P4E10.3,/4X, * AJ1
CAP1 539
11P4E10.3) /4X, * AJ2 = 1P4E13.6, * D, DOLD=*0P2F10.6,* IH=12,
CAP1 540
1 2 * ZEVT, ZEVP, TEVP=*1P4E10.3)
CAP1 541
IF(1.-RSUM.LT.1.E-10) RETURN
CAP1 542
IF(ABS(AJ1T-AJ10) .GT. 1.) EPRAT1=AMAX1(-1.,AMIN1(1.,
CAP1 543
1 (AJ1T-AJ10)/(AJ1T-AJ10))
CAP1 544
IF (ABS(AJ2T-AJ20) .GT. 1.) EPRAT2 = AMAX1(-1.,AMIN1(1.,
CAP1 545
1 (AJ2T-AJ20))/(AJ2T-AJ20))
CAP1 546
RR=AMIN1(1.-RSUM,1.3*RR)
CAP1 547
DOLD=D
CAP1 548
IF(NINO.LT. NMAX) GO TO 200
CAP1 549
PRINT 1630^, J.RR.RSUM.SX.SY.SZ.SXY.AJI , AJ2,D
CAP1 550
J
CAP1 551
DOLD
CAP1 552
STOP 3121
CAP1 553
STOP 3120
CAP1 554
C  ********
C  CUT STRAIN INCREMENT AND RESTART
C  ********
700 RR=0.5*RR
CAP1 555
NRE=NRE+1
CAP1 556
IF(NRE .GE. 1) PRINT 1700,NRE,N,K,J,DH,DOLD,DORG,RR,SX,SY,SZ,SXY,
CAP1 557
1 EX,EY,EZ,EXY,ZEVP,TEVP,1H,AJ1,AJ2,F11,F21,AJ10,AJ20,F10,F20,
CAP1 558
1 F1T,F2T,DPDE
CAP1 559
EPRAT1=EPRAT2=0.1
CAP1 560
IFCNRE.GE.10) STOP 3120
CAP1 561
GO TO 200
CAP1 562
STOP 3121
CAP1 563
END
CAP1 564
C  ********
C  ********
1020 FORMAT (4(2A5,E10.3))
CAP1 565
1021 FORMAT (1H+,60X,3H1N=12,4H CAP)
CAP1 566
1022 FORMAT (2A5,7E10.3)
CAP1 567
1023 FORMAT (2A5,10A10,2A5,E10.3,2A5,E10.3)
CAP1 568
1040 FORMAT (4(2A5,1PE10.3))
CAP1 569
1042 FORMAT (2A5,1P7E10.3)
CAP1 570
1044 FORMAT (2A5,10A10,2A5,1PE10.3,2A5,E10.3)
CAP1 571
1700 FORMAT (* RESTART WITH NRE,N,K,J=12,15,213,* DH,DOLD,DORG,RR=*
CAP1 572
1 4F15.10/* SX,SY,SZ,SXY=1P4E10.3,* EX,EY,EZ,EXY=4E10.3/)
CAP1 573
2 *ZEVP,TEVP,1H=2E10.3,15,* AJ1,AJ2=2E10.3,* F11,F21=2E10.3
CAP1 574
3 /* AJ10,AJ20,F10,F20=4E10.3,* F1T,F2T,DPDE=3E10.3)
CAP1 575
END
CAP1 576
C  ********
SUBROUTINE CAPPR

SUBROUTINE CAPPR(P, EP, M, D, IH, YM)

COMMON /EQS/EQSTC(6), EQSTD(6), EQSTE(6), EQSTG(6), EQSTH(6), EQSTN(6)
1 EQSTG(6), RHO(6), RHO(6), YC(6), YAD(6), MU(6), ESR(6, 20), CLIN, QSQ,
2 TRIQ, AMAKS, SP(6), G2(6), PMIN(6)
COMMON /POR/ POR(6, 5), PORB(6, 5), PORC(6, 5), EVP(6, 5)

P = POR(1, M)
IF (EP .GE. -1.E-6) GO TO 145
NC = 5
IF (EP .LT. EVP(M, 5)) GO TO 130
NC = 0
125 NC = NC + 1
IF (EP .LT. EVP(M, NC)) GO TO 125
130 P = (POR(1, NC) + (PORB(M, NC) + PORC(M, NC) * EP) * EP)
YM = -(PORB(M, NC) + 2. * EP * PORC(M, NC))
IF (D .LT. RHO(6)) GO TO 145
EMU = D / RHO(6) - 1.
PS = EMU * (EQSTC(M) + EMU * (EQSTD(M) + EMU * EQSTG(M)))
IF (PS .LT. -P) GO TO 145
YM = -D * EQSTC(M) / RHO(6)
P = -P
IH = 9
145 RETURN
END
SUBROUTINE DEPOS

SUBROUTINE DEPOS(NPART, IN)

THIS ROUTINE USED WITH SRI GENRAT.
CALLED BY GENRAT FOR RADIATION DEPOSITION CALCULATIONS.
ROUTINE IS SEPARATED INTO 3 PARTS BY INDICATOR, NPART, TO
1 READ DATA ON MATERIAL ABSORPTION PROPERTIES
2 READ SPECTRA, AND DEPOSIT ENERGY INTO SS ARRAY.
3 PRINT OUT COORDINATE ARRAYS IN DEPOSITION EDIT

INPUT -
* TWO FORMAL PARAMETERS
* READS ABSORPTION SPECTRA, RADIATED SPECTRA FROM CARDS.
OUTPUT -
* FILLS SS ARRAYS.
* SETS *SSTOM*, *JSTAR*, *NSPEC*.
* WRITES DEPOSITION EDIT.

THIS IS A VERSION MODIFIED TO ACCEPT THE ABSORPTION COEFFICIENT
DATA DIRECTLY FROM FSCATT.

INTEGER H, PORCS, PRESS, RINTER, SOLID, SPALL
REAL MATL, NEM, NET, NEM, NET

COMMON AZERO (1), CEF, CFS, DAEG, DLETIM, DISGPT (10), DOLD, DRHC, DTMAX,
1 DMIN, DTON, DU, DX, EOL, F, FAC, FIRST, J, JCYS, JINIT,
2 JFIN, JREZON (15), JMAX, JSTAR, JTS, LSUB (3D), M, MAXPR (3D), N, NCCYCS,
3 NEDIT, NFRN, NR, NSCGB (6), NSFPR, NSPALL, NTEDIT,
4 NTEX, NTN (15), P, PD (20), R (3D), RLST, SLST, SMAX, TEDIT (5D),
5 TF, TIME, TJ, TREZON, TS, T2 (20), ULAST, UOLD, UZERO, XLAST, XNOW, XOLD
1, XMD (20), MS

HALFSTEP VALUES

COMMON DH, DLAST, DUH, PH, RH, RLHST, SH, SLHST, UH, UHSLAST, UX, XHLAST
1, NEM, NETH

CONDITION INDICATORS

COMMON INFL, INTER, MIRROR, NORMAL, PORCS, PRESS, RINTER, SOLID, SPALL

CELL LAYOUT

COMMON DXX (3D), JBN (3D), JMAT (3D), NAUT (MATL (6), NLAYER, NMTRLS,
1, THK (3D)

COORDINATE ARRAYS

COMMON/COORD/X(2D), XDI(2D), CHL (2D), DHL (2D), DPD (2D), DPDE (2D),
1 EHL (2D), H(2D), H, NEM (2D), NET (2D), PHL (2D), RHL (2D), SDT (2D),
2 SHL (2D), T (2D), U (2D), YHL (2D), ZHL (2D)

COMMON/NCS/A(5DD)

NUCLEAR DATA

REAL MU, N, MUM

COMMON /EOS/ EQST (6), EQSTC (6), EQSTD (6), EQSTE (6), EQSTOL (6),
1 EQSTH (6), EQST (6), EQST (6), EQST (6), EQST (6), EQST (6),
2 EQTE (6), EQT (6), EQT (6), EQT (6), EQT (6), EQT (6),
3 EQTE (6), EQT (6), EQT (6), EQT (6), EQT (6), EQT (6),
4 EQTE (6), EQT (6), EQT (6), EQT (6), EQT (6), EQT (6),
5 EQTE (6), EQT (6), EQT (6), EQT (6), EQT (6), EQT (6),
6 EQTE (6), EQT (6), EQT (6), EQT (6), EQT (6), EQT (6),
7 EQTE (6), EQT (6), EQT (6), EQT (6), EQT (6), EQT (6),
8 EQTE (6), EQT (6), EQT (6), EQT (6), EQT (6), EQT (6),
9 EQTE (6), EQT (6), EQT (6), EQT (6), EQT (6), EQT (6),
10 NSD (6), NRF (6), NCF (6), NVAR (6)

COMMON/RAD/SSTP (9), START (9), SDRH, SSTPOM, NSPEC, SSJ, SSS, IPILOT (4)

COMMON/X (4), XMIN (4), XP (4), XMLN (4), IA (7), ITITLE (24), NARZ, TARZ

COMMON/SS/S (5D)

COORDINATION

COMMON/PES/LVMAX, LVTOT, LVAR (2D), COM (4000)

DIMENSION AC (109), AAD (6, 6, 1D), AA (6, 6, 1D), AA (6, 6, 1D),
1 A (6, 6, 1D), A (6, 6, 1D), A (6, 6, 1D), A (6, 6, 1D), A (6, 6, 1D),
2 ATW (6), BBDY (100), BW (6), NAME (6), E1B (100)

DIMENSION DELX (2D), EPGJ (2D), PCT (2D), CPG (2D), P (2D),
1 DIPMCC (2D), FRONT (3D), XPL (2D), YPL (2D), EABS (2D)

DIMENSION ANGLE (3D)

EQUIVALENCE (A (2D), AC), (A (3D), AAD), (A (670), AA), (A (103D), AA),
1 (A (103D), AAD), (A (179D), AAD), A (21D), E1, A (2219), RHOC
2 (A (2555), TBL), A (2364), NOE, A (24DOD), ATW

EQUIVALENCE (DELS, A), (EPGJ, A (2D)), (PCT, A (2D)), (PG, A (6D)),
1 (P, A (6D)), (A (2D), DIPMCC, A (2D)), (XPL (2D), EABS (2D)),
2 (YPL (2D), EABS (2D))

DATA BBBDY, D1, D0, D0, D0, D0, D0, D0, D0, D0, D0, D0, D0, D0, D0
1, 1.85, 0.95, 1.05, 1.15, 1.25, 1.35, 1.45, 1.55, 1.65 1.75, 1.85, 1.95, 2.05
2, 22.15, 2.25, 2.35, 2.45, 2.55, 2.65, 2.75, 2.85, 2.95, 3.05, 3.15, 3.25
3, 33.35, 3.45, 3.55, 3.65, 3.75, 3.85, 3.95, 4.05, 4.15, 4.25, 4.35, 4.45

210
SUBROUTINE DEPOS (Continued)
44.55,4 .65
55.75,5 .85
68.3,8. 5,8
716.5,1 7.5
DATA E IBB
13.312E -4,
27 057E -3,
31 544E -2,
42 067E "2,
52 183E -2,
61 998E -2,
71 657E -2,
81 279E -2,
99 350E -3,
. 1 021E -2,
. 4 380E -3,
. 1 730E -3,
, 1 092E -4,

.75,4 .85,4.95,5 05,5.15 ,5.25,5.35,5.45,5.55,5.65,
. 95,6 .10,6.3,6.5 ,6.7,6. 9,7.1,7.3,7.5,7.7,7.9,8. 1 ,
,8.9, 9.1,9.3,9.5 ,9.7,9. 9,10.5,11.5,12.5,13,5,14.5,15.5,
8.5, 1 9.5,20.5,5* 0.0/
.076E -7,2.829E-6 7.604E 6,1 ,466E -5,2.393E-5,
555E 4,1.582E-3, 2.475E- 3,3. 498E 3)4.622E-3,5 .818E-3,
330E- 3,9.595E-3,
085E- 2, 1 . 208E 2,1.326E-2,1 .438E-2,
644E- 2,1.736E-2,
819E- 2, 1 . 893E 2,1.960E-2,2 .017E-2,
106E- 2,2.138E-2,
163E- 2,2. 178E 2,2.187E-2,2 .188E-2,
172E- 2,2.155E-2,
132E- 2,2. 105E 2J2.073E-2,2 .037E-2,
956E- 2,1.910E-2,
863E- 2, 1 . 814E 2,1.763E-2,1 .711E-2,
603E- 2,1.549E-2,
495E- 2,1 • 440E 2,1.387E-2,1 .332E-2,
227E- 2,1.176E-2,
125E- 2,1 . 076E 2,1.027E-2,9 .800E-3,
91 OE- 3,8.470E-3,
572E- 2, 1 . 417E- 2,1.274E-2,1 .142E-2,
1 10E- 3,8.100E-3,
200E- 3,6. 370E- 3,5.640E-3,4 .970E-3,
850E- 3,3.380E-3,
960E- 3,2. 510E- 3,2.350E-3,1 .980E-3,
500E- 3,5,008E-3,
425E- 3,1 . 147E 3,5.322E-4,2 .429E-4,
852E- 5,2.131E-5,
269E- 6, 3, 996E- 6,2.960E-6,5 *0./

DEPOS
DEPOS
DEPOS
DEPOS
DEPOS
DEPOS
DEPOS
DEPOS
DEPOS
DEPOS
DEPOS
DEPOS
DEPOS
DEPOS
DEPOS
DEPOS
DEPOS
DEPOS
C
DEPOS
2
F0RMAT(1H+,79X,5H IND=A2,5H, IN=I2,9H -DEPOS- )
DEPOS
3
F0RMAT(1H+,103X,« ANGLE FROM NORMALCDEG)«)
DEPOS
4
F0RMAT(1H+,103X,*, , ,0AL/CM2,SEC,SEC*)
DEPOS
5
F0RMAT(1H+,103X,*,CMJERG/G*)
DEPOS
10
FORMAT(A1,A9,A5,A2, I 3,3(A10,El 0.3))
DEPOS
11
F0RMAT(4(A10,1PE10.3))
DEPOS
14
FORMAT (*1«,10A10//3X,*J*,9X,*X*,9X,*X*,2(5X,*DEPOS*),
DEPOS
1 3X,*PCT TR,*,5X,«TEMP.*,4X,*PRESS.*,3X,«IMPULSE*,5X,*MATERIAL«4X, DEPOS
2*C0ND*6X,*J*,8X,«DX*2X,*ABS0RBED*/1OX,*INCH*8X,*CM*3X,*ERGS/GM*,4X DEPOS
3 ,*CAL/GM*,14X,*DEG. C«,6X,»KBAR*,5X,*KTAPS«,36X,*CM*3X,*CAL/CM2*) DEPOS
15
FORMAT(I4,2F10.6,1P2E10.3,Fl0.3,1P3E10.3,3X,A9,3X,3R2,2X,15,1PE10. DEPOS
Td.FIO.d)
DEPOS
54
FORMAT(AID, I 10,A7,A3, 1P5E10.3)
DEPOS
55
FORMATCAI,A9,A8,12,A10,F10.7)
DEPOS
56
FORMATCA1,A9,2A10,I10,A10,F10.3)
DEPOS
57
FORMATC1P8E10.3)
DEPOS
70
F0RMAT(*0*8X,*YIELD=«1PE10.3,« SOUND SP=*1PE10.3,* DENSITY=*
DEPOS
1
1PE10.3,* TENS STR=*1PE10.3,* INTERFACE STRENGTH=*1PE10.3/)
DEPOS
72
F0RMATC10X,*DEPOS - CONST. DENSITIES (G/CM3), RHOC =*1P6E10.3)
DEPOS
73
FORMATMOX, «DEPOS - ESUM =*1PE10.3,* CAL/CM2*)
DEPOS
74
FORMAT(*
TOTAL ENERGY ABSORBED IS*1PE12.3,* CAL/CM2*)
DEPOS
75
FORMAT (A1,A9,2A10, I 10,4A10)
DEPOS
76
FORMAT(* TOTAL ENERGY ABSORBED IS*1PE12.3,* CAL/CM*)
DEPOS
77
FORMAT(* TOTAL ENERGY ABSORBED IS*1PE12.3,* CAL*)
DEPOS
80
F0RMAT(A7,1P4E13.5)
DEPOS
81
F0RMAT(7X,1P4E13.5)
DEPOS
82
FORMAT (« ERROR IN MCCLOSKEY INTEGRAL FOR LAYER*13,*, X(J + 1)=«
DEPOS
1
1PE10.3,* DID NOT LIE BETWEEN *1PE10,3,* AND *1PE10.3)
DEPOS
89
FORMAT(13X,E12.5,1X,E12.5,1X,E12.5,1X,E12.5)
DEPOS
92
F0RMAT(4(2A5,1PE10.3))
DEPOS
93
FORMAT(A1 , A9, 2A1 0, 1 PE1 0. 3, A1 0, 1 PE1 0 . 3 )
DEPOS
C
DEPOS
240
GO TO (250,400,700) NPART
DEPOS
C *************
***********
DEPOS
C
ENTRY FOR READING MATERIAL ABSORPTION PARAMETERS
DEPOS
250
CONTINUE
DEPOS
NCONST=NCON(M)
DEPOS
NN0E=10H X-RAY ABS
DEPOS
IDD=1H
$
IN5=5
DEPOS
DO 260 NC=1,NCONST
DEPOS
READ ( IN,55)A1 , NAMEC NO , A2, I TAPE, A3, PBWC NO
DEPOS
WRITE (6,55)A1 , NAMEC NO, A2, I TAPE, A3, PBWC NO
DEPOS
WRITE C6,2) IDD, IN
DEPOS
INL=IN
DEPOS
IF CI TAPE .EQ. O GO TO 255
DEPOS
INL=ITAPE
DEPOS
CALL REDRCNAMECNO,NNOE, INL,2)
DEPOS
255
READ C I NL, 56) A1 , NAMEC NO , A2, A3, NOECM, NO , A4, ATWTCM, NO
DEPOS
WRI TE C6, 56)A1 , NAME( NO , A2, A3, NOECM, NO , A4, ATWTCM, NO
DEPOS
WRITE CS^) IDD, INL
DEPOS
NaED=NOECM,NO
$
NOEI=NOED+1
DEPOS
READ
CINL,89)
CEDGECM,NC,ND),ND=1,NOED)
DEPOS
FN=7H EDGE
DEPOS
WRITE C 6,80)FN,CEDGECM,NC,ND),ND=1,NOED)
DEPOS
WRITE (6,2) IDD,INL
DEPOS
READ
CINL,89)
CAAOCM,NC,ND),AA1CM,NC,ND),AA2CM, NC,ND),
DEPOS

211

47
48
49
50
51
52
53
54
55
56
57
58
59
60
51
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
62
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
1 10
1 1 1
1 12
1 13
1 14
1 15
1 16
1 17
1 18
1 19
120
121


SUBROUTINE DEPOS (Continued)

1  AA3(M,NC,ND),ND=1,NOED)
2 FN=7H COEFS
3 WRITE (6,80) FN,AA0(M,NC,1),AA1(M,NC,1),AA2(M,NC,1),AA3(M,NC,1)
4 IF (NOED .EQ. 1) GO TO 256
5 WRITE (6,81) (AA0(M,NC,ND),AA1(M,NC,ND),AA2(M,NC,ND),AA3(M,NC,ND),
6 1 ND=1,NOED)
7 256 WRITE (6,2) IDD, INL
8 260 CONTINUE
9 WTOT=1.
10 IF (NCONST .EQ. 1 .OR. PBW(I) .LT. 1.) GO TO 260
11 WTOT=0.
12 DO 270 NC=1,NCONST
13 PBW(NC)=PBW(C NC)*ATWT(M,NC)
14 WTOT=WTOT+PBW(NC)
15 DO 290 NC=1,NCONST
16 RHOC(M,NC)=RHO(M)*PBW(NC)/WTOT
17 IF (NCONST .EQ. 1) RHOC(M,1)=RHO(M)
18 WRITE (6,72) (RHOC(M,NC),NC=1,NCONST)
19 RETURN
20 C *****ENTRY FOR DEPOSITING RADIATION IN THE SS ARRAY*****
21 C
22 READ (5,54) A1,NSPEC,N2,A2,(ANGLE(NL),NL=1,5)
23 WRITE (6,54) A1,NSPEC,N2,A2,(ANGLE(NL),NL=1,5)
24 WRITE (6,2) IDD, INL
25 IF (N2 .EQ. 7H ANGLES ) GO TO 402
26 DO 401 NL=2,NLAYER
27 ANGLE(NL)=ANGLE(1)
28 GO TO 403
29 IF (NLAYER .LE. 5) GO TO 403
30 READ (5,57) (ANGLE(NL),NL=6,NLAYER)
31 WRITE (6,57) (ANGLE(NL),NL=6,NLAYER)
32 WRITE (6,2) IDD, INL
33 C BEGIN LOOP OVER EACH SPECTRUM
34 TOTCAL=0.
35 DO 4040 I=1,500
36 SS(I)=0.
37 DO 485 NS=1,NSPEC
38 JFINNS=JFIN*(NS-1)
39 IN = 5
40 IDD=5H
41 INDICATOR IN COLUMNS 11 THROUGH 15 SHOWS SPECTRUM TYPE
42 5H NHNU = ARBITRARY SPECTRUM
43 5H NBB  = SERIES OF BLACK BODIES (NBB OF THEM)
44 5H NARB = DEPOSITION FROM SCATT PROGRAM
45 READ (5,10) IND,SPECNAM, A1 ^2, NHNU, AS^CAL, A4,START(NS),A5
46 ) SSTOP(NS)
47 WRITE (6,10)IDD,SPECNAM,A1,A2,NHNU,A3,ECAL,A4,START(NS),A5,
48 ) SSTOP(NS)
49 WRITE (6,2) IDD, INL
50 IF (IND .EQ. IDD) GO TO 405
51 IN = 4
52 CALL REDR(SPECNAM,IDD,IN,1)
53 CONTINUE
54 C ARBITRARY SPECTRUM INPUT
55 READ (IN,75)A1,SPECNAM,A2,A3,NHNU,(IVAR(I ),I=1,4)
56 WRITE (6,75)A1,SPECNAM,A2,A3,NHNU,(IVAR(I ),I=1,4)
57 WRITE (6,2) IDD, IN
58 IF (IVAR(I ) .NE. IDD) GO TO 412
59 DO 410 NH=1,NHNU
60 READ (IN,11) A1,TBL(NH),A2,EI(NH),A3
61 WRITE (6,11) A1,TBL(NH),A2,EI(NH),A3
62 IF (NH .LT. NHNU) GO TO 410
63 ESUM = 0.
64 412 READ (IN,IVAR)(TBL(NH),EI(NH),NH=1,NHNU)
65 WRITE (6,IVAR)(TBL(NH),EI(NH),NH=1,NHNU)
66 WRITE (6,2) IDD, IN
SUBROUTINE DEPOS (Continued)

413 DO 413 NH = 1, NHNU
415 ESUM = ESUM + E1(NH)
417 EI(NH) = EI(NH)*ECAL/ESUM

C BLACK BODY INPUT

420 NRAD = NBB $ NHNU = 95
422 BLACK BODY INPUT

\[
\text{WRITE (6,93) A1,SPECNAM,A2,TEMP,A4,ECAL}
\]

\[
\text{WRITE (6,2) I0D,IN}
\]

428 EI(NH) = ECAL*EIBB(NH)
430 ESUM = 0.

DO 431 NH=1,NHNU

ESUM = ESUM + EI(NH)

WRITE (6,73) ESUM

C COMPUTATION OF ABSORPTION COEFFICIENT - AC

432 X(1) = 0.
433 NCONST = NCON(M)
434 DO 445 NC = 1, NCONST

445 DO 446 NH = 1, NHNU

AC(NH) = 0.

NEDG = 1

DO 445 NH = 1, NHNU

ALNE = ALOG(TBL(NH)*TEMP)

IF (TBL(NH)*TEMP .GE. 1.) GO TO 438

AC(NH) = AC(NH) + RHOC(M,J) * EXP(AA0(M,J) + ALNE * AA1(M,J,1))

1 * (0.602252/ATWT(M,J))/ANGLE(L)

GO TO 444

438 IF (NEDG .GE. NOE(M,J)) GO TO 440

IF (EDGE(M,J,NEDG+1) .GT. TBL(NH)*TEMP) GO TO 440

NEDG = NEDG + 1

GO TO 438

AC(NH) = AC(NH) + RHOC(M,J) * EXP(AA0(M,J) + ALNE * AA1(M,J,NEDG))

1 * ALNE * AA2(M,J,NEDG) + ALNE * AA3(M,J,NEDG)) * (0.602252/ATWT(M,J))/ANGLE(L)

GO TO 444

440 CONTINUE

C DISTRIBUTE ENERGY INTO CELLS

441 XBNDM = XBNDM + THK(L)

442 IF (J .GT. JBEG) XBNDM = JBN(L) - 1

443 J = JBEG

444 IF (J .GT. JBEG+1) AND. XBNDM .EQ. 0.) GO TO 447

DEP = 0.

445 DEP = DEP + AC(NH) * E1(NH)

446 IF (J .EQ. JBEG) GO TO 4462

FRONT(4,L) = DEP/RH0(M) + FRONT(4,L)

GO TO 447

447 IF (XBNDM .GT. 0.) GO TO 4481

DX = X(J+1) - X(J)

GO TO 449

448 DX = ABS(PERCNT/DEP)

IF (DX .GT. 1.05*DX1) DX = 1.05*DX1

IF (XBNDM .GT. X(J) + DX) GO TO 448

DX2 = 2.0*DX

DX = XBNDM - X(J)

X(J+2) = XBNDM

JBN(L) = J+1

JBN(J) = J
SUBROUTINE DEPOS (Continued)

448  X(J+1)=X(J)+DX
449  DX1=DX
450  ESUM=0.
451  DO 450 NH=1,NHNU
452     IF (EI(NH) .LT. 1.E-20) GO TO 450
453        EIZ=EI(NH)*((1.-EXP(-1.*AC(NH)*DX))
454        EI(NH)=EI(NH)-EIZ
455        ESUM=ESUM+EIZ
456     CONTINUE
457    SS(JFINNS+J)=ESUM*4.186E7/RH0(M)/DX/CSSTOP C NS)-START(NS))*ANGLE(L)
458     1 +SS(JFINNS+J)
459    TOTCAL=ESUM+TOTCAL
460    IF (J .EQ. JBNDM) GO TO 460
461       J=J+1
462     GO TO 446
463
464  JBEG=JBND(L)+1
465    IF (JFIN .GT. 0) GO TO 462
466       JFIN=JBEG
467       X(JFIN)=X(J+1)
468
469  JINIT=1
470    NR=NR+1
471    IF (NR-NRAD) 424,424,485
472
473  DEPOSITION FROM SCATT PROGRAM
474  ETOT=0.
475  DO 483 L=1.NLAYER
476     M=JMAT(L)
477     RATIO = 1.
478     IF (NARB .GE. 0) GO TO 466
479        READ (5,11) A1,RHOOLD
480        WRITE (6,11) A1,RHOOLD
481        RATIO = RHOOLD/RHOCM)
482 466  CONTINUE
483    READ (1^75) Al ,SPECNAM
484    J
485    A2, A3, NPOINT, ( I VAR(I), I = 1 , 4)
486    WRITE (6,75) A1,SPECNAM
487    J
488    A2, A3, NPOINT, (IVAR(I),I=1,4)
489    WRITE (6,2) IDD,IN   $   WRITE (6,5)
490 476  CONTINUE
491    FRONT(1,L) = EI(I) * ECAL + FR0NT(1,L)
492     IF (SPH(M) .GT, 0.) FR0NT(2,L)=FR0NT(1,L)/SPH(M)+22.2
493        FRONT(3,L)=FR0NT(1,L)*RHO(M)«EQSTG(M)«4.186E-2
494     J=1
495     IF (L .GT. 1) J=JBND(L-1)+1
496     IF (ABS(TBL(1)-X(J)) .LT. l.E-10 .AND. RATIO .EQ. 1.) GO TO 478
497        DX = X(J)-TBL(1)
498        DO 477 I=1,NPOIN
499            X1=TBL(I-1)
500            X2=TBL(I)   $   X3=TBL(I+1)
501            Z1=(X1-X3)/(X2-X1)*(X1-X2)/(X3-X1)
502            Z2=(X1-X2)/(X3-X2)*(X1-X3)/(X2-X3)
503            Z3=(X1-X2)/(X3-X2)*(X1-X3)/(X2-X3)
504            FRONT(4,L)=ECAL*E1(I)»*Z1*E1(2)»Z2*E1(3)»Z3 + FRONT(4,L)
505 477  CONTINUE
506    CALL SCATTO(TBL,EI,ECAL,NPOINT,NS, L,ESUM)
507    ETOT=ESUM*RHO(M)+ETOT
508
509 483  CONTINUE
510     RATIO = ECAL
511     IF (lABS(NARB) .EQ. 1)  RATIO = ECAL/ETOT
512     DO 484
513         J=1
514        JFIN=JBEG
515        IF (JFIN .GT. 0) GO TO 462
516 9/12/79  1
517 482  CONTINUE
518     PRINT 82,L,XJP1, TBL(L), TBL(NPOINT)
519     GO TO 481
520
521 460  I=MINO(I,NPOINT-1)
522     X1=TBL(I-1) $ X2=TBL(I) $ X3=TBL(I+1)
523     Z1=(X1-X3)/(X2-X1)*(X1-X2)/(X3-X1)
524     Z2=(X1-X2)/(X2-X2)*(X1-X3)/(X3-X2)
525     Z3=(X1-X2)/(X3-X2)*(X1-X3)/(X2-X3)
526     FRONT(4,L)=ECAL*E1(I)»*Z1*E1(2)»Z2*E1(3)»Z3 + FRONT(4,L)
527 481  CONTINUE
528    CALL SCATTO(TBL,EI,ECAL,NPOINT,NS,L,ESUM)
529    ETOT=ESUM*RHO(M)+ETOT
530 483  CONTINUE
531     RATIO = ECAL
532     IF (IABS(NARB) .EQ. 1)  RATIO = ECAL/ETOT
533     DO 484 J=1,JFIN
534
535 214
SUBROUTINE DEPOS (Continued)

484  SS(JFINNS+J)=SS(JFINNS+J)*RATIO
485  TOTCAL=TOTCAL+RATIO*ETOT
500  RETURN

C **ENTRY FOR PRINTING DEPOSITION EDIT
700  WRITE (6,14) (DISCPT(I),I=1,10)
701  CONTINUE
702  RETURN

C **FIND IMPULSE IN EACH LAYER
100  ENTRY FOR PRINTING DEPOSITION EDIT
600  WRITE (6,14) CDISCPTCI),I=1,10)
601  CONTINUE
602  RETURN

C0EF=1.2*SQRT(2.)
215
SUBROUTINE DEPOS (Continued)

DO 709 J=1,JFIN
DIMPMEC(J)=COEF*SQRT(DIMPMEC(J))*1.0E-3
FPCT=100.
JJ=0
FEPS=FRONT(1,1)*4.186E7
WRITE (6,15) JJ
X(1)=FRONT(1,1),FPCT,FRONT(2,1),
1 DIMPMEC(J),MATL(1,1)
L=K=J1=1 M=JMAT(L)
709 DIMPMCC(J)=COEF*SQRT(DIMPMCC(J))*1.0E-3
FPCT=100.
JJ = 0
FEPG=FRONT(1,1)*4.186E7
WRITE (6,15) JJ
Xm
X(1)
J
FEPG ,FRONTC 1 ,1)
1 PRONTO,1 J.XCI
J.MATLd
1 )
L=K=J1=1 M=JMAT(L)
710 J2=MINO(JFIN-1,50*K,JBND(L))
DO 712 J=J1,J2
XINCH=X(J)/2.54
WRITE (6,15)
J,XINCH,X(J),EPGJ(J),CPG(J),PCT(J),TC(J),P(J),
1 DIMPMCC(J),MATL(M,1),(H(J,1),I=1,3),J,DELXCJ),EABS(J)
712 CONTINUE
IF (J2 .EQ. JFIN-1) GO TO 740
J1=J2+1
IF (J2 .NE. 50*K) GO TO 718
K=K+1
WRITE (6,14)(DISCPT(I,1),I=1,10)
718 IF (J2 .NE. JBND(L)) GO TO 710
WRITE (6,15)
J,XINCH,X(J1)/2.54
FEPG=FRONT(1,L)*4.186E7
WRITE (6,15)
JJ,XINCH,X(J1),FEPG,FRONT(1,L),PCT(J2),FRONT(2,L),
1 FRONT(3,L),DIMPMEC(J2),MATL(M,1)
GO TO 710
740 WRITE(6,70)
YHL(J2),CHL(J2),DHL(J2),T(J2-1),T(J2)
L=L+1
M=JMAT(L)
FEPG=FRONT(1,L)*4.186E7
XINCH=X(J1)/2.54
WRITE (6,15)
JJ,XINCH,X(J1),FEPG,FRONT(1,L),PCT(J2),FRONT(2,L),
1 FRONT(3,L),DIMPMEC(J2),MATL(M,1)
GO TO 710
742 PRINT 74, SUMCAL
GO TO 746
743 SUMCAL=3.14159*SUMCAL
PRINT 76, SUMCAL
GO TO 746
744 SUMCAL=4.18879*SUMCAL
PRINT 77, SUMCAL
GO TO 746
746 CONTINUE
IF (IPL0T .EQ. 0) GO TO 780
C ********************
C GRAPHS OF DEPOSITED ENERGY
JEND=JFIN-1
L=1 JJ=1 XPL(1)=X(1)
DO 754 J=1,JEND
JJ=JJ+1
IF (J .EQ. JBND(L)) GO TO 752
XPL(JJ)=0.5*(X(J)+X(J+1))
GO TO 754
752 XPL(JJ)=X(J)
IF (J .EQ. JEND) GO TO 754
JJ=JJ+1
L=L+1
M=JMAT(L)
754 CONTINUE
JMAX=JJ
DO 776 I=1,4
IF (IPL0T(I),6,0) GO TO 776
GO TO (756,756,756,756,756)
756 ITITLE(17)=10HABSORBED E ITITLE(18)=10HENERGY - CA
ITITLE(19)=10HTEMP. FROM ITITLE(18)=10HABS. ENER
ITITLE(19)=10HGY - DEG C
ITITLE(19)=10HPSUE0O PRE ITITLE(18)=10HSSURE AT D
ITITLE(19)=10HEP. - KBAR
762 CONTINUE
L=1 JJ=1
GO TO (769,770,771,1)
769 YPL(JJ)=CPG(J)
GO TO 772
770 YPL(JJ)= TC(J)
GO TO 772
771 YPL(JJ)= P(J)
GO TO 772
772 IF (J .NE. JBND(L)) GO TO 774
IF (J .EQ. JEND) GO TO 774
JJ=JJ+1 L=L+1
216
SUBROUTINE DEPOS (Concluded)

    YPL(JJ)=FRONT(I,L)

    CONTINUE
    CALL GRAPH4(XPL,YPL,JMAX,1,XMAX(I),XMIN(I),YMAX(I),YMIN(I),1TITLE,
        IA)

    CONTINUE
    RETURN
    END
SUBROUTINE DFRACT

C
C
C
C
C
E-9) DVO=1.E-9
C
C
C **
C
100
C
C
DIMENSION TSR(6,30)
REAL NM,NT
DATA SMF/1.88/
IF (NM .LT. 0.) RETURN
NTRY=0
DOLD=DOLDO
VVO=NM/DOLD
VVA=VVO
VS0=1./DOLD-VVO
PS0=P/(VSO«DOLD)
DV0=1./DHO-1./DOLD
IF (ABS(DVO) .LT. 1
DV=DVO
IF (TSR(M,7) .EQ. 0.

BEGIN SUBCYCLING LOOP FOR CASE OF LARGE STRAIN
NL00P=MAX1(1,,-2.«DV*EQSTCM/VS0/TSR(M,5)+0.5,2.5*TSR(M,1)«DTO_*
AMIN1(P-TSR(M,2),TSR(M,2)))
DELV=DV/NLOOP
EXX=EXX1*DELV/DV0
EYY=EYY1*DELV/DV0
ETT=ETT1«DELV/DV0
EXY=EXY1*DELV/DV0
VH=1./DOLD
YT=Y
DT=DELV/DVO«DTO
A1=TSR(M,1)«DT
DPJ=0.2*(ABS(TSR(M,5))+ABS(P))
DO 380 NL=1.NLOOP
VH=VH+DELV
DH=1./VH
DE=(EH-EOLD)*(VH-1./DOLD)/DVO
E=(EH-EOLD)«(VH-1./DOLD)/DVO+EOLD
TEMP1=1.-RHOS«EQSTGM«E/EQSTCM
ESTIMATE OF PRESSURE BASED ON STRAIN, GROWTH, NUCLEATION
PN = 0,
YS=VSC**2*RHOS/EGSTCM
YSC=YS/(PSO+RHOS*EGSTGM*DE)
DVS=DELV
PG=AMAX1((YSC-DELV)/YS,EGSTCM*(1./RHOS/(VSC+DELV)-TEMP1))
PS=PG
IF(C.5*(PJ+PSO) .GT. AMAX1 (TSRCM^J^SRfM.S) ) )GO TO 300
IF (DELV .GT. 0.) PN=2.«TSR(M,6)*ALOG(DELV«DH/TSR(M,7)/DT)+
2 2.*TSR(M,5) - PSO
IF (VVO .LE. 0.) GO TO 150
XN=0. $ XP=1.0
IF (PSO .LT. TSR(M,5)) XN=TSR(M,7)/DH*DT*EXP((PSO-TSR(M,5))/TSR
1 (M,6))
IF (PSO .LT. TSR(M,2)) XP=EXP(A1*(PSO-TSR(M,2)))
YG=VVO*XP*A1/2.
YG0=VVO*(XP-1.)-YG*PSO
YN=XN/(2.*TSR(M,6))
YN*PSO
PG=(DELV-YSC-YGC-YNC)/(-YS+YQ+YN)

218
SUBROUTINE DFRACT (Continued)

CH=1.
IF (0.5*(PG+PSO).GT.TSR(M,2)) YG=YQC=CH=0.
IF (0.5*(PG+PSO).GT.TSR(M,5)) YN=YNC=CH=0.
IF (CH.EQ.0.) PG=(DELV-YSC-YGC-YNC)/(-YS+YG+YN)
IF (DELV.GT.0. .AND. PSO .LT. TSR(M,2)) PG=AMIN1(PG,TSRCM,2)
PJ=AMAX1(PG,PSO)
150 DVS=1./RHOS/(PJ/EQSTCM+TEMP1)-VSO
VVA=VVO+DELV-DVS
NC = 0.
C BEGIN ITERATION LOOP
200 NC=NC+1
VV=VVO+DELV-DVS
PJ=PA+EQSTCM*(1./RHOS/(VSO+DVS)-TEMPI)
PN=AMIN1(0.5*(PA+PSO)-TSRM,0.0)
IF (PN .LT. 0.) PN=EXPCPN/TSRM
VNA=TSRM*PN*DT/DH
VGA = WQ
PG=AMIN1(0.5*(PA+PSO)-TSRM,0.)
IF (PG .LT. 0.) VGA=VVO*EXP(A1*PG)
VVA=VGA+VNA
250 IF (NC.EQ.1) GO TO 270
IF (NC.EQ.2) GO TO 280
IF (NC.EQ.3) GO TO 290
IF (NC.EQ.4) GO TO 300
IF (NC.EQ.5) GO TO 310
IF (NC.EQ.6) GO TO 320
IF (NC.EQ.7) GO TO 330
IF (NC.EQ.8) GO TO 340
IF (NC.EQ.9) GO TO 350
IF (NC.EQ.10) GO TO 360
IF (NC.EQ.11) GO TO 370
IF (NC.EQ.12) GO TO 380
IF (NC.EQ.13) GO TO 390
IF (NC.EQ.14) GO TO 400
IF (NC.EQ.15) GO TO 410
IF (NC.EQ.16) GO TO 420
IF (NC.EQ.17) GO TO 430
IF (NC.EQ.18) GO TO 440
IF (NC.EQ.19) GO TO 450
IF (NC.EQ.20) GO TO 460
IF (NC.EQ.21) GO TO 470
IF (NC.EQ.22) GO TO 480
IF (NC.EQ.23) GO TO 490
IF (NC.EQ.24) GO TO 500
IF (NC.EQ.25) GO TO 510
IF (NC.EQ.26) GO TO 520
IF (NC.EQ.27) GO TO 530
IF (NC.EQ.28) GO TO 540
IF (NC.EQ.29) GO TO 550
IF (NC.EQ.30) GO TO 560
IF (NC.EQ.31) GO TO 570
IF (NC.EQ.32) GO TO 580
IF (NC.EQ.33) GO TO 590
IF (NC.EQ.34) GO TO 600
IF (NC.EQ.35) GO TO 610
IF (NC.EQ.36) GO TO 620
IF (NC.EQ.37) GO TO 630
IF (NC.EQ.38) GO TO 640
IF (NC.EQ.39) GO TO 650
IF (NC.EQ.40) GO TO 660
IF (NC.EQ.41) GO TO 670
IF (NC.EQ.42) GO TO 680
IF (NC.EQ.43) GO TO 690
IF (NC.EQ.44) GO TO 700
IF (NC.EQ.45) GO TO 710
IF (NC.EQ.46) GO TO 720
IF (NC.EQ.47) GO TO 730
IF (NC.EQ.48) GO TO 740
IF (NC.EQ.49) GO TO 750
IF (NC.EQ.50) GO TO 760
IF (NC.EQ.51) GO TO 770
IF (NC.EQ.52) GO TO 780
IF (NC.EQ.53) GO TO 790
IF (NC.EQ.54) GO TO 800
IF (NC.EQ.55) GO TO 810
IF (NC.EQ.56) GO TO 820
IF (NC.EQ.57) GO TO 830
IF (NC.EQ.58) GO TO 840
IF (NC.EQ.59) GO TO 850
IF (NC.EQ.60) GO TO 860
IF (NC.EQ.61) GO TO 870
IF (NC.EQ.62) GO TO 880
IF (NC.EQ.63) GO TO 890
IF (NC.EQ.64) GO TO 900
IF (NC.EQ.65) GO TO 910
IF (NC.EQ.66) GO TO 920
IF (NC.EQ.67) GO TO 930
IF (NC.EQ.68) GO TO 940
IF (NC.EQ.69) GO TO 950
IF (NC.EQ.70) GO TO 960
IF (NC.EQ.71) GO TO 970
IF (NC.EQ.72) GO TO 980
IF (NC.EQ.73) GO TO 990
IF (NC.EQ.74) GO TO 1000
IF (NC.EQ.75) GO TO 1010
IF (NC.EQ.76) GO TO 1020
IF (NC.EQ.77) GO TO 1030
IF (NC.EQ.78) GO TO 1040
IF (NC.EQ.79) GO TO 1050
IF (NC.EQ.80) GO TO 1060
IF (NC.EQ.81) GO TO 1070
IF (NC.EQ.82) GO TO 1080
IF (NC.EQ.83) GO TO 1090
IF (NC.EQ.84) GO TO 1100
IF (NC.EQ.85) GO TO 1110
IF (NC.EQ.86) GO TO 1120
IF (NC.EQ.87) GO TO 1130
IF (NC.EQ.88) GO TO 1140
IF (NC.EQ.89) GO TO 1150
IF (NC.EQ.90) GO TO 1160
IF (NC.EQ.91) GO TO 1170
IF (NC.EQ.92) GO TO 1180
IF (NC.EQ.93) GO TO 1190
IF (NC.EQ.94) GO TO 1200
IF (NC.EQ.95) GO TO 1210
IF (NC.EQ.96) GO TO 1220
IF (NC.EQ.97) GO TO 1230
IF (NC.EQ.98) GO TO 1240
IF (NC.EQ.99) GO TO 1250
IF (NC.EQ.100) GO TO 1260
IF (NC.EQ.101) GO TO 1270
IF (NC.EQ.102) GO TO 1280
IF (NC.EQ.103) GO TO 1290
IF (NC.EQ.104) GO TO 1300
IF (NC.EQ.105) GO TO 1310
IF (NC.EQ.106) GO TO 1320
IF (NC.EQ.107) GO TO 1330
IF (NC.EQ.108) GO TO 1340
IF (NC.EQ.109) GO TO 1350
IF (NC.EQ.110) GO TO 1360
IF (NC.EQ.111) GO TO 1370
IF (NC.EQ.112) GO TO 1380
IF (NC.EQ.113) GO TO 1390
IF (NC.EQ.114) GO TO 1400
IF (NC.EQ.115) GO TO 1410
IF (NC.EQ.116) GO TO 1420
IF (NC.EQ.117) GO TO 1430
IF (NC.EQ.118) GO TO 1440
IF (NC.EQ.119) GO TO 1450
IF (NC.EQ.120) GO TO 1460
IF (NC.EQ.121) GO TO 1470
IF (NC.EQ.122) GO TO 1480
IF (NC.EQ.123) GO TO 1490
IF (NC.EQ.124) GO TO 1500
IF (NC.EQ.125) GO TO 1510
IF (NC.EQ.126) GO TO 1520
C ENDING ROUTINE
C
300 NM=VVA*DH
NT=NT*DH/DOLD+TSRM*PN*DT
IF (NM .LT. 0.6) GO TO 400
SUBROUTINE DFRACT (Concluded)

BETA=2.*TXY*ALFA/NLOOP
ELMUF=2.*ELMU*AMAX1(1.-SMF*VVA*DH,0.)
WS1=0.6667*(DOLD-DH)/(DOLD+DH)
TXY=TXY+ELMUF*EXY+(SYY-SXX)*ALFA/NLOOP
SXX=SXX+ELMUF*(EXX-WS1)+BETA
SYY=SYY+ELMUF*(EYY-WS1)-BETA
STT=STT+ELMUF*(ETT-WS1)
WS4=SXX**2+SYY*2+STT**2+TXY**2
YES=YF*AMAX1(1.-4.*VVA*DH,0.)
IF (WS4 .LT. YE**2/1.5) 00 TO 340
WS3=YE/SQRT(1.5*WS4)
PTERM=(DOLD-DH)/(DOLD+DH)/DT/TSR(M,1)
WS5=1.5/TSR(M,1)/DT
SXX=SXX*WS3+EXX*WS5-PTERM
SYY=SYY*WS3+EYY*WS5-PTERM
STT=STT*WS3+ETT*WS5-PTERM
TXY=TXY*WS3+EXY*WS5
CONTINUE

WSQ=PJ
P=PJ*(VSO+DVS)*DH
Y=YT
VVO=VVA
VSO=VH-VVA
DOLD=DH
RETURN

C END WITH SEPARATION

P=0.
Y=0.
SXX=0.
SYY=0.
STT=0.
TXY=0.
NM=ABS(NM)
RETURN

C PROVISION FOR ABORT IN CASE OF ITERATION FAILURE

NTRY=NTRY+1
DV=1./DHO-1./DOLD
NL0AP = MAX1(3.,-4.*2.*NTRY*DV*EQSTCM/VSO/TSR(M,5)+0.5)
GO TO 100

C FORMATS

FORMAT (30H ITERATION FAILURE IN DFRACT/5H J,K=212,3H M=12,
1 4H PJ=1PE10.3,6H DELV=E10.3,7H DELFA=E10.3,7H DELVB=E10.3,
2 7H DELVC=E10.3)
END
SUBROUTINE EDIT

* EDIT LISTS COORDINATE QUANTITIES FOR TIME OF EDIT

INPUT - JSTAR,
OUTPUT - NTEX,

INTEGER H,POROUS,PRESS,RINTER,SOLID,SPALL
REAL MATL,NEM,NETH,NEMH,NETH
COMMON AZERO,J,CFS,CS,AVG,OELTIM,DISCPT(10),DOLO,ORHO,OTMAX,
1 DMIN,DTN,DTNH,DU,EOLD,FAC,FIRST,J,JCYS,JINIT,
2 JFIN,JREZON,JSTAR,JTS,LSUR(30),M,MAXPR(30),NM,NCYCS,
3 NEDIT,NPERN,NREZON,NSCRB(6),NP,PRAT,NSPALL,NTEDT,
4 NTEX,NTR(15),P0L0,PLH(20),P6(20),PLAST,SLAST,SMAX,TEEDIT(50),
5 TF,TIME,TJ,TS,TREZON,ULAST,U0L,UZERO,UXLAST,U0UX
COMMON DH,DH0LAST,DU,H,PRESS,RINTER,SOLID,SPALL

INNER,CONDITION INDICATORS
COMMON INF,LINNER,MIRROR,NORMAL,POROUS,PRESS,RINTER,SOLID,SPALL

COORDINATE ARRAYS
COMMON/COORD/X(200),X0(200),CHL(200),DH(200),DPHD(200),DPDE(200),DPCO(200),
1 EHL(200),EHL0(200),3NEM(200),NETH(200),PMH(200),RHL(200),SDO(200),
2 SHL(200),SH(200),1J(200),2YHL(200),ZHL(200),
COMMON /IND/ IEDS(6),INDR(20),NALPHA,NCRP(6),NFR(6),NPOR(6),
1 NDS(6),NPR(6),NCON(6),NVAR(6),
COMMON /PES/ LMAX,LVTO,LVAR(200),COM(4000)

DIMENSION PI(300),P2(300),EMOM(300)

PRINTOUT FOR EACH EDIT

NTEX=NTEX+1
CALL SECOND(CHANGE) $ OUR=CHANGE-FIRST
JSTAR=MIN0(JSTAR+1,JFIN-1) $ NPTS=JSTAR-JINIT+1
WRITE (6,1025)(DISCPT(I),I=1,10) $ EEDIT=EMSUM(1)
WRITE(6,1026)NTEX,N,FIRST,JSTAR,DUR,OTNH
EMSUM = EEDIT(JINIT) = 0.
J1=JINIT $ L=1. $ M=JMAT(L)
J2=MIN0(JSTAR,JBN(15))
NJ=J2-J1+1
IF (NFR(M) .EQ. 1) GO TO 7
IF (NFR(M) .EQ. 1 .OR. NFR(M) .EQ. 2) GO TO 7
IF (NFR(M) .EQ. 3) GO TO 5
IF (NFR(M) .EQ. 3) GO TO 5
IF (NFR(M) .EQ. 3) GO TO 5
IF (NFR(M) .EQ. 3) GO TO 5
IF (NFR(M) .EQ. 4) GO TO 9
GO TO 7

DO 6 J=J1,J2
P1(J)=YHL(J)
6 P2(J)=SHL(J)-PHL(J)
IF (NFR(M) .GE. 3) GO TO 10
N1 = 10H $ YIELD = 10H DEVIATOR
N2 = 10H $ G = 10H DEVIATOR
GO TO 13

DO 8 J=J1,J2
P1(J)=NEM(J)
8 P2(J)=NET(J)
N1 = 10H $ N2 = 10H NET
IF (NFR(M) .EQ. 1) N1 = 10H $ FBURN
IF (NFR(M) .EQ. 0) N1=10H $ RVV
GO TO 13

N1=10H $ N2=10H DEVIATOR
DO 91 J=J1,J2
91 221
SUBROUTINE EDIT (Concluded)

P1(J)=NEM(J)
91 P2(J)=SHL(J)-PHL(J)
GO TO 13
10 IF (H(J,3).EQ. 3 .AND. ANO .H(J,3).EQ. 5R M) LV=LVAR(J)+3
P1(J)=COM(LV)
P2(J)=COM(LV+1)
11 CONTINUE
N1=10H/NEM/RVV
N2=10HSD/NET/ENV
13 CONTINUE
DO 14 J=J1,J2
14 EMOM(J+1)=EMSUM+0.5*ZHL(J)*(U(J)+U(J+1))*EMSUM
WRITE (6,1029) N1,N2
1029 FORMAT (IHG,10AIQ)
1025 FORMAT (1H0,10A10)
1026 FORMAT (18H0 TIME EDIT NO.3,7H AT N =15.8H) TIME =1PE12.5 ,
1 16H SECS; JSTAR =15.14H02.5 ,CALC TIME IS 0PF10.3,13H SECS; DTNH =
2 1PE10.3,5H SECS/
1027 FORMAT (15,0PF9.6,F9.0,1PE10.3,3,0PF8.6,1PE11.4,3,0PE11.4,3,0PE10.3,3,
1 1PE11.4,3)
1028 FORMAT (15,0PF9.6,F9.0,1PE10.3,3,0PF8.6,1PE11.4,3,0PE11.4,3,0PE10.3,3,
1 1PE11.4,3)
1029 FORMAT (4X,1HJ,3X,1HX,3X,1HU,3X,3HRL,7X,3HPHL,7X,3HSHL,7X,3MEHL,1
5X,3HRL,8X,3HCM,2X,4HCOND,17X,3HMDM,1X,A10,1X,A10/
2 5H CELL,7X,2HCM,3X,6HCM/SEC,3(10H OYN/CM2 ),6X,4HERGS,2X,
3 6HCM/CM3,5X,6HCM/SEC,22X,4HTAPS)
END

222
SUBROUTINE EOSTAB(NCALL, IN, XN, YN, ZN)
DIMENSION X(30), Z(30), EX(30)
IF (NCALL .GE. 0) GO TO 100

C  INITIALIZE AND READ DATA
READ (IN, 1001) A1, IMAX, 12, 13
IF (12 .NE. 10H VOLUME) 12=10H DENSITY
IF (13 .NE. 10H LINEAR) 13=10H LINEAR
PRINT 1001, A1, IMAX, 12, 13
READ (IN, 1002) A1, (X(I), Z(I), I=1, IMAX)
PRINT 1012, A1, (X(I), Z(I), I=1, IMAX)

C  VOLUME TRANSFORMATION
IF (12 .NE. 10H VOLUME) GO TO 45
DO 30  I=1, IMAX
X(I) = 1./X(I)
IM1 = IMAX-1
DO 50 I=1, IM1
EX(I) = (Z(I+1)-Z(I))/(X(I+1) -X(I))
IF (13 .NE. 10H LOG) GO TO 80
DO 65 I=1, IM1
IF (Z(I) .LE. 0. .OR. Z(I+1) .LE. 0.) GO TO 65
EX(I) = ALOG(Z(I+1)/Z(I))/ALOG(X(I+1)/X(I))
80 IM2 = IMAX-2
N1 = 2
NM = IMAX-1
NORDER = 1
IF (X(1) .LT. X(2)) RETURN
NORDER = 0
N1 = IMAX-1
NM = 2
IM2 = IMAX-1
RETURN

C  CALCULATE PRESSURE
IT=N1-NORDER
IF (XN .LT. X(N1)) GO TO 175
IT = NM-1+NORDER
IF (XN .GT. X(NM)) GO TO 175
DO 140 I=2, IM2
N4 = I+1
IF (NORDER .EQ. 0) N4 = IMAX-1+1
IT = N4-NORDER
IF (XN .LT. X(N4)) GO TO 175
140 CONTINUE
175 IF (13 .EQ. 10H LOG) GO TO 190
180 ZN=Z(IT)+(XN-X(IT))*EX(IT)
RETURN
190 IF (Z(IT) .LE. 0. .OR. Z(IT+1) .LE. 0.) GO TO 180
ZN=Z(IT)+(XN-X(IT))*EX(IT)
RETURN
1001 FORMAT(A10, 110, 2A10)
1002 FORMAT(A10, 6E10.3/(10X, 6E10.3))
1003 FORMAT(1H+, 79X, 4H IN=, 12, 11H EOSTAB P-V )
1012 FORMAT(A10, 1P6E10.3/(10X, 6E10.3))
END
SUBROUTINE EQST

SUBROUTINE EQST(EJ, DJ, PJ, MJ, CJ, DPDDJ, DPDEJ)

COMPUTES PRESSURE AND SOUND SPEED FOR SOLIDS AND EXPLOSIVES

* MIE-GRUNEISEN FOR COMPRESSION

PUFF HUGONIOT IN P - MU FORM

MURNAGHAN HUGONIOT FORM (FOR EQST=1.0)

LINEAR US-UP HUGONIOT FORM (FOR EQST=2.0)

* EXPANSION EQUATION OF STATE FOR DENSITIES LESS THAN RHOS

POLYTROPIC GAS EQUATION FOR EXPLOSIVES (NPR=i)

INPUT - FORMAL PARAMETERS EJ, DJ, MJ, CJ.

OUTPUT - PJ, CJ

NAMED COMMON

REAL MU, MUM

COMMON /EUS/ EQSTA(6), EQSTC(6), EQSTD(6), EQSTE(6), EQSTG(6),

1 EQSTH(6), EQSTV(6), EQSTN(6), EQSTCOM2, EQSTCOM3

COMMON /EUS/ EQSTA(6), EQSTC(6), EQSTD(6), EQSTE(6), EQSTG(6),

1 EQSTH(6), EQSTV(6), EQSTN(6), EQSTCOM2, EQSTCOM3

COMMON /MELT/ EMELT(6,5), SPH(6), THERM(6,8)

COMMON /RHO/ PHO(6), RHOS(6)

COMMON /TSR/ TSR(6,30), EXMAT(6,20), TENS(6,3)

COMMON /Y/ Y0(6), YADD(6), MU(6), MUM, YADM

COMMON /IND/ IE0S(6), INDK(20), NALPHA, NCMP(6), NFR(6), NPO(6),

1 NDS(6), NP0(6), NC0N(6), NVAR(6)

DIMENSION NHUG(6), AMURN(6), BPMUHN(6), Sl(6)

EXPLANATION OF SOME MODIFIED PUFF EXPANSION MODEL PARAMETERS

* ABSOLUTE VALUE OF EQSTV IS EXPONENT IN GRUNEISEN EXPRESSION, USE 0.5

* IF EQSTV .GT. 0., THEN MCCLOSKEY-THOMPSON LOG VARIATION IS USED FOR EJ .GT. 0.

* EQSTA IS THE COEFFICIENT OF THE SECOND TERM ASSUMED IN THE GRUNEISEN SERIES AND A NONZERO VALUE INDICATES THAT THE SLOPES OF MIE-GRUNEISEN EOS AND EXPANSION HUGONIOT FORMS DO NOT MATCH AT THE INITIAL SOLID DENSITY.

* IF EQSTA=0. OR IS UNSPECIFIED, THEN THE PRESSURE-VOLUME SLOPE IS NOT MATCHED.

INITIALIZE PORTION

IF (EQSTNMJ .GT. 0,) GO TO 200

EQSTNMJ = 1.

IF (EQSTG(MJ) EQSTE(MJ) RHOS(MJ) NE. 0,) EQSTNMJ = EQSTCMJ /

1 (EQSTG(MJ) EQSTE(MJ) RHOS(MJ))

IF (EQSTV(MJ) .GT. 0,) PRINT 1005

ENN=ABS(EQSTVMJ)

IF (EQSTV(MJ) .NE. 0,) PRINT 1007,ENN

AMURN(MJ)=0.

IF (EQSTA(MJ) .EQ. 0,) GO TO 35

PRINT 1009, EQSTA(MJ)

AMURN(MJ) = (EQSTA(MJ) *ENN* (EQSTHMJ EQSTGMJ)) / EQSTGMJ

IF (AMURN(MJ) EQSTNMJ .GT. 0,) GO TO 30

PRINT 1055, EQSTNMJ, AMURN(MJ)

STOP

30 EQSTA(MJ) = AMURN(MJ)

35 CONTINUE

NHUG(MJ) = 1

IF (EQSTS(MJ) .EQ. 1,) NHUG(MJ) = 2

IF (EQSTS(MJ) .EQ. 2,) NHUG(MJ) = 3

NHUGM = NHUG(MJ)

GO TO (180*40+60) NHUGM

C INITIALIZE FOR MURNAGHAN HUGONIOT FORM

P = A*(D/RHOS)**BOP-1

A = 20/BOP IS READ AS -C-, -BOP- IS READ IN AS -D-

AMURN(MJ) = EQSTCMJ

BPMURN(MJ) = EQSTDMJ

EQSTCMJ = EQSTCMJ EQSTDMJ

EQSTDMJ = 0.5*EQSTCMJ *(EQSTDMJ-1.)

EXTERNAL EQST
SUBROUTINE EQST (Continued)

PRINT 1010*EQSTC(MJ),EQSTD(MJ)
GO TO 180
C  INITIALIZE LINEAR US-UP HUGONIOT FORM
C  US = C1 + S1 * UP
C  -C1- IS READ IN AS -C-, -S1- IS READ IN AS -D-
C  S1(MJ) = EQSTD(MJ)
C  EQSTC(MJ) = RHOS(MJ)*EQSTC(MJ)**2
C  EQSTD(MJ) = EQSTC(MJ)+(2*EQSTD(MJ)-1.)
PRINT 1020*EQSTC(MJ),EQSTD(MJ)
GO TO 180
180 IF (EQSTN(MJ) .NE. 1.) PRINT 1050
RETURN
C  *** COMPUTATION PORTION  ***

C 200 IF (NPR(MJ) .EQ. 1) GO TO 400
AMU=1.332*MUM  $ IF (EJ .GE. EMELT(MJ)) AMU=0.
VJ=RHOS(MJ)/DJ  $ EMU=(1.-VJ)/VJ
IF (EMU .GE. 0.) GO TO 300
C
C  EQST FOR EXPANDED ZONES
ENN=0.5  $ ESUBC=1.0  $ IF (EQSTV(MJ) .NE. 0.)ENN=ABS(EQSTV(MJ))
IF (EQSTV(MJ) .GT. 0.) AND. EJ .GT. EQSTE(MJ) ESUBC=1.0
IF (EJ .GT. CJ) 60 TO 500
ERAT=EQSTN(MJ) % IF (EJ .GT. EQSTE(MJ)) ERAT=1.0
ENU2=(EQSTN(MJ)+ERAT*EQSTA(MJ))*(1.-VJ)/VJ/ESUBC
TS1=EQSTE(MJ)+ESUBC
GHNU=(EQSTG(MJ)-EQSTH(MJ))/VJ*ENN
EX2=EQSTC(MJ) % IF (ENU2 .LT. 10.) EX2=EXP(ENU2)
TS1=TS1*1.0-EX2)
TS2=EQSTG(MJ)+GHNU  $ PJ=(EJ-TS1)*DJ*TS2
IF (EJ .LT. EQSTN(MJ)) PJ=AMAX1(0.,PJ)
IF (CJ .EQ. 1) 60 TO 500
CP=TDJ*TS2+TS2*ENN+GHNU)+TS2*(EQSTE(MJ)+ESUBC*1.0)
1 *EQSTN(MJ)+ERAT*EQSTA(MJ)+ESUBC*2.*VJ-1)*VJ
DPDEJ=DJ*TS2*1.0*DJ+TS2*VJ*1.0)
IF (EJ .LT. 0.) AND. EJ .GT. EQSTE(MJ) DPDEJ=DJ*TS2
EX1=EX2*1.0-ENN)
IF (EJ .LT. 0.) AND. EJ .GT. EQSTE(MJ) DPDEJ=DJ*TS2
CSQ=DPDQJ*(EJ-TS1)*TS2/DJ*DPDEJ/AMU/DJ
GO TO 450
C  EQST FOR COMPRESSED ZONES
300 IF (NHUG(MJ) .LT. 2) 310,320,330
C  PUFF HUGONIOT
310 PH = ((EQSTS(MJ)+EMU+EQSTD(MJ))*EMU+EQSTC(MJ))*EMU
DPHDD = ((3.*EQSTS(MJ)+EMU+2*EQSTD(MJ))*EMU+EQSTC(MJ))/RHOS(MJ)
GO TO 370
C  MURNAGHAN HUGONIOT
320 PH = AMURNAGHAN*MU*(DJ/RHOS(MJ))*BPMURNAGHAN(MJ)-1.0)
DPHDD = (EQSTC(MJ)+BPMURNAGHAN(MJ)*PH)/DJ
GO TO 370
C  LINEAR US-UP HUGONIOT
330 PH = EQSTC(MJ)*(1.+VJ)/(1.+S1(MJ)*(1.+VJ))
DPHDD = VJ/DJ*(1.+S1(MJ)*(1.+VJ))/(1.+S1(MJ)*(1.+VJ))
C  COMPUTE PRESSURE DERIVATIVES AND SOUND SPEED
370 GF = 1.-0.5*EQSTG(MJ)*(1.+VJ)
PJ = PH*GF*EQSTG(MJ)/RHOS(MJ)
IF (CJ .EQ. 1) 80 TO 500
DPDEJ = DPHDD*GF-0.5*PH*EQSTG(MJ)/RHOS(MJ)/(1.+EMU)*2
DPDEJ = EQSTG(MJ)*PHOS(MJ)
CSQ = DPDEJ*PJ*DPDEJ/DJ*AMU/DJ
GO TO 450
C  EQST FOR EXPLOSIVE (NPR = 1)
400 PJ = EQSTG(MJ)*DJ*EJ
DPDEJ=EQSTG(MJ)*DJ
DPDEJ=EQSTG(MJ)*EJ
CSQ=EQSTG(MJ)*(EJ-PJ/DJ)
C  SOUND SPEED COMPUTATION

RETURN
SUBROUTINE EQST (Concluded)

450   IF (CSQ .LE. 0.) RETURN 
      $ CQ=CSQ/(CJ*CJ)
        CJ=CJ*(CQ/(CQ+1.))+0.25*(CQ+1.)
        IF (CQ .LT. 0.5 .OR. CQ .GT. 2.) CJ=SQRT(CSQ)
1005  FORMAT(* EQST: EFFECTIVE VAPORIZATION ENERGY HAS MCCLOSKEY-THOMPSON*)
      IN LOG VARIATION ABOVE EQST*)
1007  FORMAT(* EQST: EXPONENT IN GRUNEISEN EXPRESSION =*1PE10.3)
      EQST 135
1009  FORMAT(* EQST=*1PE10.3, * IS COEFFICIENT OF SECOND TERM ASSUMED IN*)
      EQST 136
1010  FORMAT(* MURNAGHAN HUGONIOT, CONSTANTS CHANGED TO EQSTC=*1PE10.3,*EQST 138
      1, EQSTD=*1PE10.3)
      EQST 139
1020  FORMAT(* LINEAR US-UP HUGONIOT, CONSTANTS CHANGED TO EQSTC=*1PE10.3,EQST 140
      1, EQSTD=*1PE10.3)
1050  FORMAT(* EXPANSION PORTION OF EQUATION OF STATE IS INCOMPLETE*)
    1, EXPANSION EOS WILL BE UNSTABLE ABOVE SUBLIMATION
1055  FORMAT(* EXPANSION OF ENERGY FOR CHOSEN VALUE OF EQST=*1PE10.3,ADDITIONAL EXPONENT=*1PE10.3)
    1, EQSTN=*1PE10.3,EQST 145
500   RETURN 
END
SUBROUTINE EQSTPF

C
C EQSTPF COMPUTES PRESSURE FROM A THREE-PHASE EQUATION OF STATE
C DEVELOPED BY PH1LCO-FORD. ROUTINE HAS TWO PARTS, ONE FOR
C READING AND INITIALIZING AND THE OTHER FOR COMPUTING PRESSURE.
C
C READ INPUT (NCALL=0). CALL IS FROM GENRAT.
C INPUT - NCALL, IN, M, AND MATERIAL PROPERTY CARDS
C OUTPUT - PRINTS CARD IMAGES, ORGANIZES DATA INTO ARRAYS
C COMPUTE PRESSURE (NCALL=1) CALL IS FROM HSTRESS USUALLY
C INPUT - NCALL, M, CJ, D, E
C OUTPUT - P (CURRENT PHASE OR STATE OF MATERIAL IS AVAILABLE)
C
C NAMED COMMON
C
C REAL MU, MUM
C COMMON /EQS/ EQSTA(6), EQSTC(6), EQSTD(6), EQSTG(6),
C 1 EQSTH(6), EQSTN(6), EQSTV(6), CZQ(6), CWQ(6), C2(6)
C COMMON /MET/ EMETL(6,3), SML(6,3), THERM(6,3)
C COMMON /RHO/ RHO(6), RHOS(6)
C COMMON /TSR/ TSR(6,3), EXMAT(6,2), TENS(6,3)
C COMMON /Y/ Y0(6), YA0D(6), MU(6), MUYADD
C DIMENSION A(6), A2(6), B(6), BP(6), C1(6), CBT(6), CC(6), CV(6), O1(6),
C 1 DEDV(6), EBL(6), EBS(6), EC(6), EES(6), E0(6), E0V0(6), EPS1(6), EPS2(6)
C 2 EPS0(6), EV0(6), HC(6), HLT(6), HLM-HSM
C 3 V0(6), VSO(6), VSOV0(6), VSOV0(6), VSOV0(6), VSOV0(6), VSOV0(6),
C 2 ZN(6), ZM(6)
C DATA ACC. R1 /1.E-4, 1.E-3, 2.5U*E7/
C
TO INITIALIZATION OR COMPUTATION PORTIONS GO TO 200

20 IF(NCALL.EQ.0) GO TO 200

C **BRANCH TO INITIALIZATION OR COMPUTATION PORTIONS**

C **READ INPUT DATA AND INITIALIZE CONSTANTS**

IND = 5
READ(IN,1101) Z1, C1(M), DL1, DMS, O1(M), HLM-HSM
WRITE(6,1101) Z1, C1(M), DL1, DMS, O1(M), HLM-HSM
WRITE(6,1102) INC, IN
READ(IN,1101) Z1, HVM, TBK, TCK, TMK, WT(M), ZK0(M)
WRITE(6,1101) Z1, HVM, TBK, TCK, TMK, WT(M), ZK0(M)
WRITE(6,1102) INC, IN
V0(M) = 1./RHOS(M)
ESO(M) = HSM
IF(DSM .GT. 0.) GO TO 50
COMPUTE -OSM- IF UNSPECIFIED
ERG = EQSTG(M) * RHOS(M) * ESO(M)
EMU = -ERG / (EQSTC(M) + ERG)
EMU = -ERG / (EQSTC(M) + (EQSTO(M) * EQSTS(M) * EMU) * EMU + ERG)
NC2 = 0
NC2 = NC2 + 1
IF(NC2 .GT. 20) GO TO 42
P = EMU * (EQSTC(M) + EMU * (EQSTD(M) + EMU * EQSTS(M)) + ERG) * ERG
PP = EQSTC(M) + ERG + EMU * (2. * EQSTD(M) + 3. * EMU * EQSTS(M))
EMU = EMU - P / PP
IF(ABS(EMU - EMUO) .GT. 0.) ACC GO TO 40
GO TO 44

42 PRINT 1103, EMUO, P, PP, EMU + M
STOP 42

44 CONTINUE
VSO(M) = V0(M) / (EMU + 1.)
GO TO 60

C ADJUST -ESO- AND -VSO- TO AGREE WITH -DSM-
VSO(M) = 1. / DSM
EMU = OSM / RHOS(M) - 1.
ESO(M) = -EMU * (EQSTC(M) + EMU * (EQSTD(M) + EMU * EQSTS(M))) / (EQSTG(M) * EMU + EMU)
1 RHOS(M) * (1. * EMU)

60 ELM(M) = ESO(M) / HLM-HSM
C COMPUTE -DLM- IF UNSPECIFIED
IF(DLM .LE. 0.) DLM = 0.935 / VS0(M)
SUBROUTINE EQSTPF (Continued)

VLO(M) = 1./DLM
TM(M) = TMK/TCK
T R = TBK/TCK
ELB = HVB-HLB
C

SOLVE FOR -CL- FROM EQ. 3.21
CL = (HLB-HLM)/(TBK-TMK)
CV(M) = (HVB-HVM)/(TBK-TMK)
DLTC = CV(M)-CL

SOLVE FOR -A1-, -A2- AND -ALPHA- FROM Eqs. 3.24
A1(M) = DLTC/R1*WT(M)
A2(M) = (ELH-DLTC)/(TBK-TMK)

SOLVE FOR -CL- FROM Eqs. 3.21
CL = (HLB-HLM)/(TBK-TMK)

SOLVE FOR -CV- FROM Eqs. 3.33
VC(M) = (HVB-HVM)/(TBK-TMK)

SOLVE FOR -CL- FROM Eqs. 3.24
CL = (HLB-HLM)/(TBK-TMK)

SOLVE FOR -AR- FROM Eqs. 3.25
AR = (A2(M)/TBK*A1(M)+0.31425*X)/(1.0*0.838*X)

SOLVE FOR -ZC- FROM Eqs. 3.27
ZC(M) = 1.0/(3.72*0.26*(AR-7.))

SOLVE FOR -VC- FROM Eqs. 3.33
VC(M) = 1.0/(3.72*0.26*(AR-7.))

SOLVE FOR -CRITICAL PRESSURE- FROM Eqs. 3.34
PC(M) = ZC(M)*R1*TCK/VC(M)*WT(M)

C

COMPUTE -KO-, -KI- AND -K2- (Eqs. 3.7)
IF(ZKO(M).EQ.0.) ZKO(M) = BL
ZK1(M) = HL-ZKO(M)
ZK2(M) = (1.*ZK1(M)*B1-A1(M)-A2(M))/2.

C

SOLVE EQ. 3.28 FOR HV TO FIND EVO, PVO, DVO, WO
PV = EXP(A2(M)*(1.-1./T)*A1(M)*ALOG(T))
X1 = T/ZC(M)
A = ZK0(M)*ZK1(M)/T
AP = ZK2(M)*T/(T-1./T)

C

SOLVE EQ. 4.5 FOR RV
RV = PV/X1
NC3=0

C

SOLVE Eqs. 4.4C, D, AND E FOR EV, RL, EL
EV = E0(M)*EPS2(M)*T*EPS1(M)*((ZK0(M)+2.*ZK1(M)/T)-ZK2(M)/RV)*T*RVEQSTP124
RL = 1.*CL(M)*(-1.-T)*01(M)/01(M)*(-1.-T)
EL = EV*EPS1(M)*PV*(1./RV-1.)/RL*(A2(M)/T*A1(M)-1.)
E1(M) = E0(M)+EL0(M)-EL
EVO(M) = EV+E1-E0(M)
PVO(M) = PV
VVO(M) = VC(M)/RV
E0(M) = E1

C

SOLVE EQ. 4.4D FOR -EC- WITH T = 1, RV = 1
EC(M) = E0(M)+EPS2(M)-EPS1(M)*((ZK0(M)+2.*ZK1(M)-ZK2(M)))

228
SUBROUTINE EQSTPF (Continued)

DEDV(M) = (ELO(M) - ESO(M)) / (VLO(M) - VS0(M))
EVO(M) = DEDV(M)
EES(M) = VO(M) * DEDV(M)
CC(M) = (C1(M) / D1(M)) ** 3 / 27.
C50 = ESO(M) / (TMK - 298.)
HDCT(M) = 0.5 * (CL - C50) * TMK
CBT(M) = 0.5 * (CL + C50) * TMK
ERL(M) = ELO(M) - C50 * TMK
EBS(M) = ESO(M) - CL * TMK
Y1(M) = 2. * CBT(M)
Y3(M) = Y1(M) * (CL - C50) * TMK

C ***** CONSTRUCT A FIT TO APPROXIMATE RV-T RELATION ON LV-V HOUNDRY
T1 = T0 = 0.95
NPART = 5
GO TO 650

100 R1 = RV
T2 = T0 = 0.9
NPART = 6
GO TO 650

105 R2 = RV
Z(M) = ALOG(1. - R1) / (1. - R2) / ALOG((1. - T1) / (1. - T2))
Z(M) = (1. - R1) / (1. - T1) ** ZN(M)
RETURN

C ***** CALCULATIONS TO FIND P(V,E)
C *****
C
C *** SELECT REGION OF PHASE DIAGRAMS
200 CONTINUE
V = 1./D
C SELECT S, SL, L OR L, LV, AND V REGIONS
IF (V .GE. VLO(M)) GO TO 300
C TEST FOR COOL SOLID
IF (E .LE. ESO(M)) GO TO 700
C *** SOLVE FOR VS ON S-SL BOUNDARY WITH ES=E
Y2 = E - EBS(M)
EZ = E
NPART = 1 $ GO TO 600
C SECOND BRANCH FOR SOLID MATERIAL, CONTINUE WITH SL AND L
220 IF (V .LT. VS) GO TO 700
C TEST FOR COOL LIQUID
IF (E .LT. ELO(M)) GO TO 750
C *** SOLVE FOR TEMP OF E AS IF E IS ON SL-L LINE
Y2 = E - EBL(M)
TF = (Y2 * SORT(Y2 * Y2 - Y3(M))) / Y1(M)
C COMPUTE ES FOR TF
EZ = ES = EBS(M) + CBT(M) * TF * HDCT(M) / TF
C GO TO 600 TO GET VS ON S-SL LINE
NPART = 2 $ GO TO 602
C COMPUTE VLM OR SL-L LINE
250 VLM = VS + (E - ES) / DEDV(M)
NL = 1
C SEPARATE SOLID-LIQUID AND LIQUID
IF (V - VLM) SETS 755 $ 755, 810
C
C *** BEGIN SWITCHING FOR L, LV, AND V REGIONS
300 IF (V .LT. VC(M)) GO TO 350
C BRANCH FOR HIGHLY VAPORIZED MATERIAL
IF (V .GT. VVO(M)) GO TO 900
C COMPUTE EC(V) AT CRITICAL TEMP TO COMPARE WITH E
ECV = EO(M) * EPS2(M) - EPS1(M) * ((ZKO(M) + 2. * ZK1(M)) * RV - ZK2(M) * RV) * RV
C SECOND PARTIAL ISOLATION OF V FROM LV REGION
IF (E .GT. ECV) GO TO 900
C COMPUTE T AND THEN EV ON LV-V LINE TO MAKE THIRD TEST FOR
C SEPARATING LV AND V
RV = VC(M) / V
X1 = RV / (IC(M) * (1. - (B(M) - BP(M) * RV) * RV)) - ZK2(M) * RV ** 3
X2 = -ZK0(M) * RV * RV
SUBROUTINE EQSTPF (Continued)

310 PV = PV 
NC4 = NC4 + 1 
IF (NC4 .GT. 20) GO TO 312 
TA = T 
PG = X1*TA*X2*X3/T 
PVP = PV*(A2(M)/T+1*A1(M))/T 
PVP = AMAX(0.*TA*X2*X3/T,AMAX(TA*(PG-PV)/(PVP-PGP),TMIN+ACC)) 
IF (ABS(PV-PVT)/PV .GT. ACC) GO TO 310 
EV = EO(M)-EPS2(M)/T-EPS1(M)*(ZK0(M)*2.*ZK1(M)/T-ZK2(M)/T) 
C BRANCH TO EITHER V OR LV REGIONS 
IF (T .LE. TM(M)) GO TO 985 
IF (E-EV) 850 TO 900 TO 900 
STOP 312 
C C ** TEST TO SEPARATE L AND LV REGIONS 
C FIRST COMPUTE T ON L-LV LINE, THEN EL 
350 NL = 2 
IF (E .GT. EC(M)) GO TO 800 
RL = VC(M)/V 
X1 = (1.-RL)/D1(M)/2. 
X = SORT(X1*X1*CC(M)) 
C GO TO 650 TO OBTAIN EL 
NPART = 1 
GO TO 650 
C BRANCH TO EITHER L OF LV REGIONS 
375 NL = 3 
IF (E-FL) 855 TO 800 TO 800 
C **** BUILT-IN SUBROUTINES 
C **** **** EUSTP02 
C **** **** EUSTP03 
C **** **** EUSTP04 
C **** **** EUSTP05 
C **** **** EUSTP06 
C **** **** EUSTP07 
C **** **** EUSTP08 
C **** **** EUSTP09 
C **** **** EUSTP10 
C **** **** EUSTP11 
C **** **** EUSTP12 
C **** **** EUSTP13 
C **** **** EUSTP14 
C **** **** EUSTP15 
C **** **** EUSTP16 
C **** **** EUSTP17 
C **** **** EUSTP18 
C **** **** EUSTP19 
C **** **** EUSTP20 
C **** **** EUSTP21 
C **** **** EUSTP22 
C **** **** EUSTP23 
C **** **** EUSTP24 
C **** **** EUSTP25 
C **** **** EUSTP26 
C **** **** EUSTP27 
C **** **** EUSTP28 
C **** **** EUSTP29 
C **** **** EUSTP30 
C **** **** EUSTP31 
C **** **** EUSTP32 
C **** **** EUSTP33 
C **** **** EUSTP34 
C **** **** EUSTP35 
C **** **** EUSTP36 
C **** **** EUSTP37 
C **** **** EUSTP38 
C **** **** EUSTP39 
C **** **** EUSTP40 
C **** **** EUSTP41 
C **** **** EUSTP42 
C **** **** EUSTP43 
C **** **** EUSTP44 
C **** **** EUSTP45 
C **** **** EUSTP46 
C **** **** EUSTP47 
C **** **** EUSTP48 
C **** **** EUSTP49 
C **** **** EUSTP50 
C **** **** EUSTP51 
C **** **** EUSTP52 
C **** **** EUSTP53 
C **** **** EUSTP54 
C **** **** EUSTP55 
C **** **** EUSTP56 
C **** **** EUSTP57 
C **** **** EUSTP58 
C **** **** EUSTP59 
C **** **** EUSTP60 
C **** **** EUSTP61 
C **** **** EUSTP62 
C **** **** EUSTP63 
C **** **** EUSTP64 
C **** **** EUSTP65 
C **** **** EUSTP66 
C **** **** EUSTP67 
C **** **** EUSTP68 

230
SUBROUTINE EQSTPF (Continued)

C SOLVE FOR P, RL, EL, RV, EV, ON LV-V BOUNDARY

650 PV=EXP(A2(M)*(1.-1./T)+A1(M)*ALOG(T))
X1=ZK0(M)*ZK1(M)/T
A=ZK2(M)*(T-1./T)
AP=ZK0(M)*(T-1./T)
PX=PV/X1
BAX=B(M)*A/X1
IF (PX*BAX -.0.25) AND. NPART (.LT. 5) GO TO 653
RL=PV/(X1/(1. - (B(M)*BP(M)*RV)^2)- (A*AP*RV))
1.)*RVR
GO TO 654
RV=1.-ZM(M)*RZ(M)
654 RV1=NC7=0
IF (NC7 (GT. 20) GO TO 670
X2=1.-B(M)*BP(M)*RV)^2
PO=X1*X2-(A*AP*RV)*RV**2
POP=X1*X2*(X1*PV*(B(M)-2.*BP(M)*RV))/(X2**2-(2.*A+3.*AP*RV)^2
RV=AMAX1(RV+(PV-PO)/POP,0.0)
655 RV=1.-ZM(M)*RZ(M)
656 RV1=NC7=NC7+1
IF (NC7 (GT. 20) GO TO 670
X2=1.-B(M)*BP(M)*RV)^2
PO=X1*X2-(A*AP*RV)*RV**2
POP=X1*X2*(X1*PV*(B(M)-2.*BP(M)*RV))/(X2**2-(2.*A+3.*AP*RV)^2
RV=AMAX1(RV+(PV-PO)/POP,0.0)
IF (ABS(RV-RV1), GT. ACC) GO TO 653
670 PRINT 1109,RV,RL1,PO,POP,EL,RL,EL,T,M
STOP 670
C CALCULATIONS FOR EACH PHASE

C *** SOLID PHASE

700 EMU = 1./RHOS(M)/V-1.
RGE = RHOS(M)*EQSTG(M)*E
P = EMU*(EQSTC(M)+EMU*EQSTD(M)+EMU*EQSTS(M)+RGE)+RGE
GO TO 1000
C *** SOLID-LIQUID MIXED PHASE

750 FMAX = (E-ESO(M)/(ELO(M)-ESO(M)
IF (V .GT. FMAX) GO TO 990
C FIND T FOR V IN SL REGION
755 EPS = E-DEDV(M)*V
ES = EPS*DEDV(M)*V
Y2 = ES-ETS(M)
TF = (Y2*SQRT(Y2*Y2-Y3(M))/Y1(M)
N5=0
760 TFO = TF
N5=N5+1
IF (N5 (GT. 20) GO TO 780
ETA = VO(M)/VS
EMU = ETA-1.
ESP = CBT(M)-HDCT(M)/TF**2
ETAP = -ESP*ETA**2*EES(M)
RGE = RHOS(M)*EQSTG(M)*E
H = EOVO(M)*(TF-1.)-EMU*(EQSTC(M)+EMU*(EQSTD(M)+EMU*EQSTS(M)+RGE)
1.)*RGE
HP = EOVO(M)*(EQSTC(M)+EMU*(2.*EQSTD(M)+EMU*3.*EQSTS(M)+RGE)*ETAP)
1.)*RGE
1 = EQSTG(M)*RHOS(M)*ETA*ESP
TF = TF/HP
ES = EBS(M)*CBT(M)*TF*HDCT(M)/TF
VS = (ES*EPS)/DEDV(M)
IF (ABS(TF-TFO), GT. ACC) GO TO 760
P = EOVO(M)*(TF-1.)
GO TO 1000
780 PRINT 1107,TF,TF0,T,M
STOP 770
SUBROUTINE EQSTPF (Continued)

C *** LIQUID PHASE
C
C SOLVE FOR PLM, VLM, ON SL-L LINE
800 Y2 = E=EBL(M)
TF = (Y2+SQRT(Y2*Y2-Y3(M)))/Y1(M)
EZ = ES = ERS(M)+CBT(M)*TF+HDCT(M)/TF
C
GO TO 600 TO GET VS ON S-SL LINE
NPART = 3
GO TO 602
805 VLM= VS*(E-ES)/DEDV(M)
810 PLM = EOVO(M)*(TF-l.)
C
C SOLVE FOR PLB, VLB ON L-LV LINE
C
IF (NL .LE. 3) GO TO 815
IF (E ,GE. EC(M)) GO TO 820
IF (NL .EQ. 1) GO TO 812
RL » VC{M)/V
X1 a (1,-RL)/D1(M)/2.
X » SQRT{X1«X1+CC(M))
T = l.-((X-Xl)»»(l,/3.)-(X+Xl)»»(l,/3.))»3
C
GO TO 650 TO OBTAIN EL
NPART = 2
GO TO 650
812 T = TM(M)
EL = ELO(M)
C
BEGIN ITERATION LOOP TO FIND VLB ON L-LV BOUNDRY, GIVEN E
C
815 TL=T $ ETL=EL $ TU=1.0 $ ETU=EC(M)
TLAST = 0.5*(TU+TL)
C
USE PARABOLIC ESTIMATE OF SLOPES TO OBTAIN T FOR E
S2=S23=(TU-TL)/(ETU-ETL)
T = TL+((S2+2*S23-S2)*(E-EL)/(ETU-ETL))»(E-EL)
TLAST = 0.5*(TU+TL)
NC8=0 $ NPART=4
816 NC8=NC8+1
IF(T .GT. TU) T=0.1* TLAST*0.9*TU
IF(T .LT. TL) T=0.1* TLAST+0.9*TL
IF(NC8 .GT. 20) GO TO 827
C
GO TO 650 TO COMPUTE RL,EL,RV,EV FOR GIVEN T
C
GO TO 650
817 IF (ABS(E-EL) .LE. ACC*AMAX1(ABS(E),ELO(M))) GO TO 819
S12 = (T-TL)/(EL-ETL)
S23 = (TU-T)/ETU-EL)
S2 = S12*S23=(TU-TL)/(ETU-ETL)
TLAST = T
IF (EL .LT. E) GO TO 818
T = T*(S2*(S12-S2) .MP(E-EL)/(ETL-EL))»(E-EL)
ETU=EL i TU=TLAST $ GO TO 816
ETL=EL 5 TL=TLAST 1 GO TO 816
818 T = T*(S2+2*S23-S2) .MP(E-EL)/(ETU-EL))»(E-EL)
ETL=EL i TL=TLAST $ GO TO 816
819 VLB=VC(M)/RL
PLB=PC(M)*PV
GO TO 825
C
SOLVE FOR PLB ABOVE CRITICAL POINT ON V = VC LINE
820 VLB = VC(M)
RV = 1.
X1 = E=EO(M)+EPS1(M)*ZK0(M) •RV
X2 = EPS1(M)+ZK2(M) .MPRV=2.*ZK1(M)) •RV
T = (X1+SQRT(X1*X1-4.*EPS2(M)*X2))/2.*EPS2(M)
P0 = RV*T/1ZC(M)★(1.-B(M)-BP(M)*RV)*RV))-(ZK0(M)+ZK1(M)/T+ZK2(M))
1(T-1/T)√RV) √RV=PV0(M)
PLB = PC(M) •P0
EO STP337
EO STP338
EO STP339
EO STP340
EO STP341
EO STP342
EO STP343
EO STP344
EO STP345
EO STP346
EO STP347
EO STP348
EO STP349
EO STP350
EO STP351
EO STP352
EO STP353
EO STP354
EO STP355
EO STP356
EO STP357
EO STP358
EO STP359
EO STP360
EO STP361
EO STP362
EO STP363
EO STP364
EO STP365
EO STP366
EO STP367
EO STP368
EO STP369
EO STP370
EO STP371
EO STP372
EO STP373
EO STP374
EO STP375
EO STP376
EO STP377
EO STP378
EO STP379
EO STP380
EO STP381
EO STP382
EO STP383
EO STP384
EO STP385
EO STP386
EO STP387
EO STP388
EO STP389
EO STP390
EO STP391
EO STP392
EO STP393
EO STP394
EO STP395
EO STP396
EO STP397
EO STP398
EO STP399
EO STP400
EO STP401
EO STP402
EO STP403
EO STP404
EO STP405
EO STP406
EO STP407
EO STP408
EO STP409
EO STP410
EO STP411
EO STP412
EO STP413
EO STP414
EO STP415
EO STP416
EO STP417
EO STP418
EO STP419
EO STP420
EO STP421
EO STP422
EO STP423
EO STP424
EO STP425
EO STP426
EO STP427
EO STP428
EO STP429
EO STP430
EO STP431
EO STP432
EO STP433
EO STP434
EO STP435
EO STP436
EO STP437
EO STP438
EO STP439
EO STP440
EO STP441
EO STP442
EO STP443
EO STP444
EO STP445
EO STP446
EO STP447
EO STP448
EO STP449
EO STP450
EO STP451
EO STP452
EO STP453
EO STP454
EO STP455
EO STP456
EO STP457
EO STP458
EO STP459
EO STP460
EO STP461
EO STP462
EO STP463
EO STP464
EO STP465
EO STP466
EO STP467
EO STP468
EO STP469
EO STP470
EO STP471
EO STP472
EO STP473
EO STP474
EO STP475
EO STP476
EO STP477
EO STP478
EO STP479
EO STP480
EO STP481
EO STP482
EO STP483
EO STP484
EO STP485
EO STP486
EO STP487
EO STP488
EO STP489
EO STP490
EO STP491
EO STP492
EO STP493
EO STP494
EO STP495
EO STP496
EO STP497
EO STP498
EO STP499
EO STP500
EO STP501
EO STP502
EO STP503
EO STP504
EO STP505
EO STP506
EO STP507
EO STP508
EO STP509
EO STP510
EO STP511
EO STP512
EO STP513
EO STP514
EO STP515
EO STP516
EO STP517
EO STP518
EO STP519
EO STP520
EO STP521
EO STP522
EO STP523
EO STP524
EO STP525
EO STP526
EO STP527
EO STP528
EO STP529
EO STP530
EO STP531
EO STP532
EO STP533
EO STP534
EO STP535
EO STP536
EO STP537
EO STP538
EO STP539
EO STP540
EO STP541
EO STP542
EO STP543
EO STP544
EO STP545
EO STP546
EO STP547
EO STP548
EO STP549
EO STP550
EO STP551
EO STP552
EO STP553
EO STP554
EO STP555
EO STP556
EO STP557
EO STP558
EO STP559
EO STP560
EO STP561
EO STP562
EO STP563
EO STP564
EO STP565
EO STP566
EO STP567
EO STP568
EO STP569
EO STP570
EO STP571
EO STP572
EO STP573
EO STP574
EO STP575
EO STP576
EO STP577
EO STP578
EO STP579
EO STP580
EO STP581
EO STP582
EO STP583
EO STP584
EO STP585
EO STP586
EO STP587
EO STP588
EO STP589
EO STP590
EO STP591
EO STP592
EO STP593
EO STP594
EO STP595
EO STP596
EO STP597
EO STP598
EO STP599
EO STP600
EO STP601
EO STP602
SUBROUTINE EQSTPF (Continued)

825  RM = 1./VLM
826  RB = 1./VLB
827  Z1 = (PLM-PLB)/(RM-RB)
828  Z2 = (RR*PLM*RM*PLB)/(RM-RB)
829  P1 = Z1/V-Z2
830  Z3 = ALOG(PLM/PLB)/ALOG(RM/RB)
831  Z4 = (ALOG(RH)*ALOG(PLM)-ALOG(RM)*ALOG(PLB))/ALOG(RM/RB)
832  ALP2 = Z3*ALOG(1./V)-Z4
833  F = (PLM/(RM-1./VL0(M))-Z3*PLM/RM)/(Z1-Z3*PLM/RM)
834  F = AMIN1(1.,F)
835  F = EXP(F*ALOG(P1)+1.*F*ALP2)
836  GO TO 1000

827  PRINT 1110,T,TMIN,TMAX,TU,TL,E,ET,ETL,ETU
828  STOP 727

C C ***        LIQUID-VAPOR MIXED PHASE
850  RL = 1.+C1(M)*(1.-T)**1./3.*DI1(M)*R1(M)1.-T)
851  EL = EV-EPS1(M)*PV1(1./RV-1./RL)*(A2(M)/T+A1(M)-1.)
852  FM = (E-ELO(M))/(E0V(M)-E0L(M))
853  IF (V .GT. FM) GO TO 990
854  IF (V .LT. FM) GO TO 990
855  IF (V .GT. TU) T = T LAST*0.8999*TU
856  IF (T .LT. TL) T = 0.1**TLAST + 0.8999*TL
857  GO TO 870
858  GO TO 650 TO COMPUTE RL, EL, RV, EV FOR GIVEN T

870  T = TL*(E-ETL)*(TU-TL)/(ETU-ETL)
871  NC6 = NC6+1
872  IF (NC6 .GT. 20) GO TO 892
873  IF (T .GT. TU) T = T LAST + 0.8999*TU
874  IF (T .LT. TL) T = 0.1*T LAST + 0.8999*TL
875  GO TO 860 TO COMPUTE RL, EL, RV, EV FOR GIVEN T
876  GO TO 650
877  ET = (RL*V-1.)/(RL/RV-1.*(EV-EL))
878  IF (ABS(E-ET) .LT. ACC*AMAX1(ABS(E),E0L(M))) GO TO 890
879  TLAST = T
880  IF (ABS(E-ETL) .LT. S12(T-TL)/(ET-ETL)
881  IF (ABS(E-ETL) .LT. S2(T-TL)/(ETL-ETL)
882  S2 = S12 + S23(T-TL)/(ETL-ETL)
883  IF (ET .LT. E) GO TO 880
884  ETL = ET $ TL = T LAST $ GO TO 870
885  ETL = ET $ TL = T LAST $ GO TO 870
886  P = PC(M)*(PV-PVO(M))
887  GO TO 1000
888  PRINT 1108,T,TMIN,TMAX,TU,TL,E,ET,ETL,ETU
889  STOP 727

C C ***  VAPOR PHASE
900  RV = VC(M)/V
901  X1 = E-ELO(M)*EPS1(M)*ZK0(M)*RV
902  X2 = EPS1(M)*ZK2(M)*RV-2.*ZK1(M)*RV
903  T = (X1+SORT1(X1*X1-4.*EPS2(M)*X2))^(2.*EPS2(M))
904  P = PC(M)*(RV*T/(ZC(M)+1.*B(M)-BP(M)*RV))/RV-PV0(M)
905  ZK2(M)*T-1.*T*RV) RV=RV-PVO(M)
906  GO TO 1000
907  CONTINUE

C C  LIQUID-VAPOR MIXED PHASE
950  RL = 1.+C1(M)*(1.-T)**1./3.*DI1(M)*R1(M)1.-T)
951  EL = EV-EPS1(M)*PV1(1./RV-1./RL)*(A2(M)/T+A1(M)-1.)
952  FM = (E-ELO(M))/(E0V(M)-E0L(M))
953  IF (V .GT. FM) GO TO 990
954  IF (V .LT. FM) GO TO 990
955  GO TO 870
956  GO TO 650 TO COMPUTE RL, EL, RV, EV FOR GIVEN T
SUBROUTINE EQSTPF (Concluded)

C *** CUTOFF AT ZERO PRESSURE

990  P = 0.
1000  RETURN
1100  FORMAT(8A10)
1101  FORMAT(A10*1P7E10.3)
1102  FORMAT(1H*79X*5H IND=A2,5H, IN=12,* READ IN EQSTPF*)
1103  FORMAT(1H-* LOC=42 IN EQSTPF*5X,* EMUX,P,PP,EMU,M= *1P5E10.3///) EQSTP473
1104  FORMAT(1H-* LOC=82 IN EQSTPF*5X,* RV1,PO,POP,RV,M=1P5E10.3///) EQSTP475
1105  FORMAT(1H-* LOC=312 IN EQSTPF*5X,* TA,PG,PPV,PGP,T,PV,M= 1P3E10*EQSTP476
1106  FORMAT(1H-, LOC=620 IN EQSTPF *5X,* TEZ,M,EMUIA,EMUIB=1P5E10.3/EQSTP478
1107  FORMAT(1H-* LOC=780 IN EQSTPF *5X,* TF,TFO,T,M=1P4E10.3///) EQSTP480
1108  FORMAT(1H-* LOC=892 IN EQSTPF*5X,* T,TMIN,TMAX,TU,TL,E,ET,ETL, EQSTP481
1ETU *1P5E10.3/1P4E10.3///) EQSTP482
1109  FORMAT(1H-* LOC=670 IN EQSTPF*5X,* RV,RV1,PV,PO,POP,REV,RL,EL,T,MEQSTP483
1 = *1P5E10.3/1P5E10.3///) EQSTP484
1110  FORMAT(1H-* LOC=827 IN EQSTPF*5X,* T,TMIN,TMAX,TU,TL,E,ET,ETL, EQSTP485
1ETU *1P5E10.3/1P4E10.3///) EQSTP486
END
SUBROUTINE ESA(NCALL, IN, M, C, D, E, P, OPDR, DPDE)

ROUTINE COMPUTES PRESSURE FROM SIMPLE TWO-PHASE EQUATION OF STATE.

ESA HAS TWO PARTS, CORRESPONDING TO READING AND COMPUTING.

READ INPUT (NCALL=0). CALL IS FROM GENRAT.

CALL IS FROM HSTRESS USUALLY.

INPUT - NCALL, M, C, D, E

OUTPUT - PRINTS CARD IMAGES, ORGANIZES DATA INTO ARRAYS

READ INPUT DATA AND INITIALIZE ARRAYS

READ (IN, 1100) Al, Gl(M), F1(M), F2(M), Pl, R1, E1

WRITE (6, 1100) Al, Gl(M), F1(M), F2(M), Pl, R1, E1

WRITE (6, 1121) IDD, IN

READ (IN, 1100) A1, P2, R2, E2, P3, R3, E3

WRITE (6, 1100) A1, P2, R2, E2, P3, R3, E3

WRITE (6, 1121) IDD, IN

INITIALIZE COEFFICIENTS IN EXPANSION EQUATION

RO=HHOS(M)

F3(M) = (F2(M) - F1(M))/RO

F4(M) = (F2(M) - F1(M))/RO/RO

G2(M) = EQSTG(M) - G1(M)

G3(M) = G1(M)/RO

INITIALIZE -B- ARRAY

AO=EQSTC(M)/RO

A1=P1-R1*E1*(G2(M)+R1*G3(M))-R1*E1*(F3(M)+R1*F4(M))

A2=P2-R2*E2*(G2(M)+R2*G3(M))-R2*E2*(F3(M)+R2*F4(M))

A3=P3-R3*E3*(G2(M)+R3*G3(M))-R3*E3*(F3(M)+R3*F4(M))

REDEFINE A TO INCLUDE DENOMINATORS

RO=RHOS(M)

D01=RO-R1 $ D02=RO-R2 $ D03=RO-R3 $ D12=R1-R2

D13=R1-R3 $ D23=R2-R3

A0=AO/(D01*D02*D03) $ A1=A1/(D01*D01*D12*D13)

A2=A2/(1002*D02*D12*D23) $ A3=A3/(D03*D03*D03*D23)

B(1,M)=-AO*P1*R2*R3-R0*AO*P1*R2*R3-R0*P1*R2*R3-R0*P1*R2*R3-R0*P1*R2*R3


RETURN

CALCULATION OF PRESSURE AND SOUND SPEED

U=(D-RHOS(M))/RHOS(M)

PH=U*(EQSTC(M)+U*(EQSTC(M)+U*EQSTC(M))

GG1=EOSTG(M)*U*G1(M)

6F=1.-0.5*U*GG1

FF=F1(M)*U*F2(M)

200 IF (D<0.0) GO TO 300

235
SUBROUTINE ESA (Concluded)

\[ P = PH \cdot GF + (GG1 \cdot D + FF \cdot E) \cdot E \]
\[ DPDR = ((EQSTC(M) + U \cdot (2 \cdot EQSTD(M) + U \cdot 3 \cdot EQSTS(M))) \cdot GF \]
\[ 1 \cdot PH \cdot (0.5 \cdot EQSTG(M) + U \cdot G1(M)) + (G1(M) \cdot D + F2(M) \cdot E) \cdot E) / RHOS(M) \]
\[ 2 \cdot G61 \cdot E \]
\[ DPDE = GG1 \cdot D + 2 \cdot FF \cdot E \]
GO TO 350

C

*** EXPANSION EQUATION OF STATE

300 GG3 = D \cdot (G2(M) \cdot D \cdot G3(M))
FF = D \cdot (F3(M) \cdot D \cdot F4(M))
BTERMS = R(1, M) + D \cdot (B(2, M) + D \cdot (B(3, M) + D \cdot B(4, M)))
P = (D \cdot RHOS(M)) \cdot BTERMS + (GG3 + FF \cdot E) \cdot E
DPDR = (G2(M) \cdot 2 \cdot D \cdot G3(M) + (F3(M) + 2 \cdot D \cdot F4(M)) \cdot E) \cdot E
1 \cdot BTERMS + (D \cdot RHOS(M)) \cdot (B(2, M) + D \cdot (B(3, M) + D \cdot B(4, M)))
DPDE = GG3 + 2 \cdot FF \cdot E

350 CSQ = DPDR + P \cdot DPDE / D \cdot \cdot 2
IF (CSQ \geq GT, 0.) C = SQRT(CSQ)
RETURN

1100 FORMAT(A10,1P7E10.3)
1121 FORMAT (1H+,79X,5H IND=A2,5H, IN=I2, -ESA-)
END
SUBROUTINE EXPLODE

This subroutine for detonating flow has three functions and is divided into three corresponding parts.

1. Initialize the material variables at the time of reading material properties.
2. Initialize the coordinate arrays to simulate initiation.
3. Compute progress of detonation during the calculation.

NAMED COMMON

REAL MU,MUM
COMMON /EQS/ EQSTA (6) , EQSTC (6) ,EQSTD (6) ,EQ5TE (6) ,ECJSTG (6) ,EQSTH (6) ,ECSTG (6) ,EQSTS (6) ,EQSTU (6) ,ZUO (6) ,CWU (6) ,CZ (6)
COMMON /MELT/ EMELT (6) , SPH (6) , THERM (6)
COMMON /RHO/ RH0 (6) , RHCS (6)
COMMON /TSR/ TSR (6) , THERM (6)
COMMON /Y/ YO (6) , YAD0 (6) ,MU (6) ,MU0 (6) ,YADDM (6)
DIMENSION BURN (6) , OET (6) , UIST (6) , ECJ (6) , PCJ (6) , QEXPL (6) , VCJ (6)
DIMENSION EHL (1) , DHL (1) , PHL (1) , SHL (1) , FBUR (1) , X (1)

IF (NCALL = 2) 100 * 200 * 300

INITIALIZE MATERIAL VARIABLES

100 READ (IN) A1, QEXPL (M) , BURN (M) , OIST (M)
PRINT 1010, A1, QEXPL (M) , BURN (M) , DIST (M)
PRINT 1001 IN ET (M) = SQRT (2. * QEXPL (M) * EQSTG (M) * (EQSTG (M) + 2.))
EHL (1) = DET (M)
VCJ (M) = (EQSTG (M) + 1.) / (EQSTG (M) + 2.) * RHO (M)
ECJ (M) = 2. * (EQSTG (M) + 1.) * QEXPL (M) / (EQSTG (M) + 2.)
PCJ (M) = 2. * RHO (M) * QEXPL (M) * EQSTG (M)
IF (DIST (M) .EQ. 0.) PRINT 1102, QEXPL (M) , DET (M) , VCJ (M) , ECJ (M) , PCJ (M)
1130 FORMAT (* AMUR. HI, VOCN = *1P3E10.3)
RETURN

INITIALIZE CELL VARIABLES

200 CONTINUE
IF (DIST (M) .EQ. 0.) GO TO 270
Ox = X (J+1) - X (J)
IF (DX .LE. 0.) GO TO 250
XH = 0.5 * (X (J+1) * X (J+1))
TBURN = (ABS (XH - BURN (M)) - DIST (M) * DX) / OET (M)
IF (TBURN .GE. 0.) GO TO 250
FBURN = AMIN1 (1., TBURN * DTNH) / (DIST (M) * DX)
EHL (J) = QEXPL (M) * (ECJ (M) - QEXPL (M)) * FBURN
DHL (J) = RHO (M) / (1. - FBURN * (1. - VCJ (M) * RHO (M)))
PRL (J) = SHL (J) * PCJ (M) * FBURN
FBUR (J) = FBURN
250 IF (FBUR (J) .NE. 0.) PRINT 1300, J, M, EHL (J) , DHL (J) , PRL (J) , FBUR (J)
RETURN
270 EHL (J) = QEXPL (M)
FRUR (J) = 1.0
RETURN

COMPUTE DETONATION PROCESS.

300 CONTINUE
DX = X (J+1) - X (J)
XH = 0.5 * (X (J+1) * X (J+1))
DH = OHL (J)
TBURN = (ABS (XH - BURN (M)) - DIST (M) * DX / 2.) / OET (M)
FBURN = AMIN1 (1., AMAX1 (TIME - 0.5 * DTNH, TBURN) * DET (M) / (DIST (M) * DX), 1. - RHO (M) / DH) / (1. - VCJ (M) * RHO (M) * FBUR (J))
IF (FBURN .LT. 1.E-3) RETURN
HDV = 0.5 * (1. / OET (M) - 1. / DH)
POLD = PRL (J)
RETURN
SUBROUTINE EXPLODE (Concluded)

PHL(J) = EQSTG(M) * DH * (EHL(J) * POLD * HDV * QEXPL(M) * (FBURN - FBUR(J)) + QH * 2 * EXPLO163
1 * HDV) / (1 - EQSTG(M) * HDV * DH)
EHL(J) = EHL(J) * (PHL(J) * POLD * HDV * QEXPL(M) * (FBURN - FBUR(J)) + 2 * QH * HDV * EXPLO165
PHL(J) = AMAX1(PHL(J), PCLJ(M) * FBURN)
EHL(J) = AMAX1(EHL(J), ECJ(M) * FBURN)
FBUR(J) = FBURN
IF (FBURN .EQ. 1.) PRINT 1400, J, DH
1400 FORMAT(* DETONATION COMPLETED FOR J=*I5,* WITH DENSITY =*1PE12.4*) EXPLOJ70
RETURN
C
1000 FORMAT(A10,7E10.3)
1010 FORMAT(A10,1P7E10.3)
1001 FORMAT(1H++,79X,* IND=*, IN=*,I2,*, EXPLODE-*,*, ERG/G, CM, 1/CM*) EXPLO074
1100 FORMAT(* OUTPUT OF EXPLODE, DET=*,1PE10.3,*, VCJ=*,1PE10.3,*, ECJ=*,1PE10.3,*, PCJ=*,1PE10.3)
1012 FORMAT(10X,* EXPLODE-CON.
1 OUTPUT OF EXPLODE WITH ENERGY=*,1PE10.3,*, ERG=*,1PE10.3,*, CM=*,1/CM*) EXPLO078
1300 FORMAT(* EXPLODE, J=*,I3,*, M=*,I3,*, E=*,1PE10.3,*, D=*,F10.6,*, P=*, F=*,F6.3)
END
SUBROUTINE EXTRA

ROUTINE IS CALLED TO READ IN AUXILIARY INFORMATION FROM CARDS

INPUT = NONE

OUTPUT = ANY WORDS IN COMMON WHICH ARE READ FROM THE EXTRA CARDS

INTEGER NEMH, NETH
REAL MATL, NEM, NET, NEMH, NETH

COMMON AZERO(1), CEF(X), CXS, DAVG, OELTIM, OISCP(10), OLO, DRHO, DTMAX,
       DMIN, DTHN, OUX, EOLD, F, FAC, FIRST, J, JCYS, JINIT,
       JFIN, JREZON(15), JSTAR, JTS, LSMUB(30), MAXPR(30), NCYCS,
       NEDIT, NREZON, NSCRB(6), NSEPRT, NSPALL, NTEDT,
       NTEX, NTR(15), PUFCOM 2
REAL MATL, NEM, NET, NEMH, NETH

COMMON NEMH, NETH

COMMON IN=5, $  JO=6

READ (IN, 902) (A(I), I=1, 9)

IF (EOF(IN)) 19, 15

IF (A(I) .EQ. 2H $) GO TO 18

WRITE (JO, 902) (A(I), I=1, 9)

GO TO 20
SUBROUTINE EXTRA (Concluded)

19 WRITE (IO,903)
20 REWIND 7
    IF (NREC .LE. 0) GO TO 35
    DO 25 NRC=1,NREC
25 READ (7,NLIST)
35 CONTINUE
    RETURN
901 FORMAT(1H**,79X,** INPUT FROM **EXTRA** ROUTINE**)
902 FORMAT (A2,A8,7A10)
903 FORMAT (** EOF ENCOUNTERED BY EXTRA**)
END
SUBROUTINE FMELT

SUBROUTINE FMELT(LS, M, EN, FM, FG, X, MSAVE)
C
SUBROUTINE COMPUTES THE THERMAL STRENGTH
C AND MODULUS REDUCTION FACTORS FM AND FG
C
LS = -1 INITIALIZE FOR NOMINAL VALUES OF FMELT AND GMELT
C 0 INITIALIZE FOR STRENGTH (FM)
C 1 INITIALIZE FOR MODULUS (FG)
C 2 COMPUTE FOR STRENGTH
C 3 COMPUTE FOR MODULUS
C FM COMPUTE FOR BOTH STRENGTH AND MODULUS
C
M MATERIAL NUMBER
C EN DIMENSIONAL ENERGY
C X INPUT ARRAY FOR INITIALIZING PARAMETERS
C ZERO TO 3 PARABOLIC REGIONS MAY BE USED
C INPUT VALUES
C NO. 1 2 3 4 5 6 7 8
C ZERO MELT
C ONE MELT DF1
C TWO MELT E1 DF1 F1 DF2
C THREE MELT E1 DF1 F1 E2 DF2 F2 DF3
DIMENSION E(6,6),F(6,18),NREG(6,2),X(7)
C 50 IF (LS.GT. 0) GO TO 200
C
42 IF (LS .GT. 0) GO TO 200
C
40 LS=NREG(M,1)=NREG(M,2)=0
X(1)=EN $ X(2)=0.35 $ X(3)=0.15 $ X(4)=0.25 $ X(5)=-0.06
X(7)=0.
C 30 IF (MSAVE .EQ. M) GO TO 150
C
C 50 INITIALIZE IN REGION 1
C
100 IF (X(1) .GT. 0.)) GO TO 100
E(M,1+LS*3)=X(1)
NREG(M,LS+1)=1
RETURN
C 60 NIN = 9 *LS
IF (X(4) .NE. 0.) GO TO 100
IF (ABS(X(2)) .GT. 0.251) GO TO 500
RETURN
C 100 NR=1
IF (X(2) .LT. 1. ) X(2)*X(2)*X(1)
F(M,1+NIN) =1.
F(M,2+NIN)=X(4)-1. +4.*X(3)/X(2)
F(M,3+NIN)=-4.*X(3)/X(2)**2
E(M,1+LS*3) =X(2)
NIN = NIN+3
IF (X(7) .NE. 0.) GO TO 120
NR = 2
RETURN
C 120 NR=2
IF (ABS(X(3)) .GT. 0.25 *X(4)+1.E-4) GO TO 500
IF (ABS(X(3)) .GT. 0.25*(1.-X(4))+1.E-4) GO TO 500
RETURN
SUBROUTINE FMELT (Concluded)

IF (ABS(X(3)) .GT. 0.25*(1.-X(4))+1.E-4) GO TO 500
RETURN

C INITIALIZE FOR THE THIRD REGION

150 NR = 3
NIN = 9*LS + 6
EM = E(M,3+LS*3)
F(M,1+NIN) = X7-X5/(EM-X5)*(-X7+4.*X(1)*EM/(EM-X5))
F(M,2+NIN) = (-X7+4.*X(1)*EM+X5)/(EM-X5)
F(M,3+NIN) = -4.*X(1)/(EM-X5)**2
NREG(M,LS+1) = 3
MSAVE = 0
IF (ABS(X(1)) .GT. 0.25*X7+1.E-4) GO TO 500
RETURN

C ***************IK**********X
C COMPUTATION OF STRENGTH REDUCTION FUNCTION, FM

200 CONTINUE
IF (LS .NE. 3) GO TO 250
IF (NREG(M,2) .NE. 0) GO TO 350
250 NN = NREG(M,1)
IF (NN .LE. 0) OR (EN .LE. E(M,1)) GO TO 255
IF (EN .GT. 0) GO TO 260
255 FM = 1.0
GO TO 300
260 IF (NN .LE. 0) GO TO 265
IF (EN .LT. E(M,NN)) GO TO 275
265 FM = 0.
GO TO 300
275 N = 0
280 N = N + 1
IF (EN .GE. E(M,N) .AND. N .LT. NN) GO TO 280
NIN = 3 *(N-1)
FM = F(M,1+NIN) + (F(M,2+NIN) + F(M,3+NIN)*EN)*EN
300 IF (LS .NE. 3) GO TO 400, 320, 310
310 IF (NREG(M,2) .NE. 0) GO TO 350
320 FG = FM
GO TO 400

C COMPUTATION OF MODULUS REDUCTION FUNCTION, FG

350 N = NREG(M,2)
IF (NN .LE. 0) OR (EN .LT. E(M,4)) GO TO 355
IF (EN .GT. 0) GO TO 360
355 FG = 1.0
GO TO 400
360 IF (NN .LE. 0) GO TO 365
IF (EN .LT. E(M,NN+3)) GO TO 375
365 FG = 0.
GO TO 400
375 N = 0
380 N = N+1
IF (EN .GE. E(M,N+3) .AND. N .LT. NN) GO TO 380
NIN = 3*(N-1)+9
FG = FM + F(M,2+NIN) + F(M,3+NIN)**2
400 RETURN

500 PRINT 1500, NR
1500 FORMAT (33HERROR IN FMELT, SLOPE IN REGION 13,
1 51H IS POSITIVE BECAUSE CURVE OFFSET EXCEEDS (F1-F2)/4)
RETURN
END
SUBROUTINE GENRAT

SUBROUTINE GENRAT
C
C  READS INPUT DATA AND Initializes ARRAYS
C  * READS INPUT CARDS, EXCEPT FOR RADIATION INFORMATION
C  * COMPUTES COORDINATE LAYOUT
C  * Initializes DENSITY, ENERGY, YIELD, SOUND SPEED, SPALL
C  * STRENGTH, CONDITION INDICATORS, PARTICLE VELOCITY
C  * PRINTS INITIAL LAYOUT FOR NON-RADIATION PROBLEMS
C
C  INTEGER H, POROUS, PRESS, RINTER, SOLID, SPALL
C  REAL MATL, NEM, NET, NEMH, NETH
C  REAL T(J), X(J), Y(J), Z(J), EHL(J)
C  COMMON /ZERO(1), CEF, CKS, DAVG, DELTIM, DISCPT(10), DOLD, DRHO, DMAX,
C  1 DTMN, DTN, DTHN, DU, DX, EOLD, F, FAC, FIRST, J, JCYCS, JINIT,
C  2 JFIN, JRESSON(15), JMAX, JSTAR, JTS, LSUB(30), M, MAXPR(30), N, NCYC,
C  3 NEDIT, NPERN, NR, NRESSON, NSCRB(6), NSEPRAT, NSPAR, NTEDT,
C  4 NTEX, NTR(15), POLD, PG(20), R(30), RLAST, SLAST, SMAX, TEDIT(50),
C  5 TF, TIME, TJ, TRESSON, TS, T1(20), ULAST, UOLD, UZERO, UXLAST, XNOW, XOLD
C  6 XJDI(20), MS
C
C  HALFSTEP VALUES
C  COMMON DH, DHLAST, DUH, DHU, RH, RHLAST, SH, HLS, UH, UHLAST, XH, XHLAST
C  
C  CONDITION INDICATORS
C  COMMON INF, LINTER, MIRROR, NORMAL, POROUS, PRESS, RINTER, SOLID, SPALL
C
C  CELL LAYOUT
C  COMMON DXX(30), JBND(30), JMUT(30), NAUTO, MALT(6,2), NLAYER, NMTL,
C  1 THK(30)
C
C  NAMED COMMON
C
C  REAL MU, MUM
C  COMMON /EQS/, EQSTA(6), EQSTC(6), EQSTD(6), EQSTE(6), EQSTG(6),
C  1 EQST(6), EQSTN(6), EQSTT(6), EQSTV(6), C2W(6), C2(6)
C  COMMON /MELT/, EMELT(6,8), GMELT(6,8), SPH(6), THERM(6,8)
C  COMMON /RHO/, RH(6), RHOS(6)
C  COMMON /TSR/, TSR(6,30), EXMAT(6,20), TENS(6,3)
C
C  Y/ Y(6), YADD(6), MU, YM, YADD
C  COMMON /COORD/, HRESULT, H(301), CDELFIN, H(1000)
C  COMMON /COORD/X(200), X(200), XH(200), DH(200), DOL(200), DPLH(200), DPLEX(200),
C  1 EH(200), H(200,3), NEM(200), NET(200), PHL(200), RHL(200), SDT(200),
C  2 SHL(200), T(200), U(200), V(200), W(200), XH(200)
C  COMMON /NSC/A(5000)
C  COMMON /JOE/, JOE(100), JNUM(100), JTYPE(100), NAME2(40), JESIDZ,
C  1 MODULUS, NERR, NLJEDIT, NTAPE
C  COMMON /IND/, IEOS(6), INDK(20), NALPHA, NCMP(6), NFR(6), NPO(6),
C  1 NDS(6), NFR(6), NCON(6), NVAR(6)
C  COMMON /RAD/, RDSSOR(9), RSTOR(9), SSURM, SSSTOR, SSPEC, SSS, SSTOR, S1PL(4)
C  COMMON /PE/, PVCM, PVCMV, LS(100), PVCM, PVCMV
C  COMMON /ESC/, ESC(6,20)
C  DIMENSION DELFIN(30,5), DELX(30,5), TH(30,5), NCELLS(30,5), NzONES(30)
C  INTEGER HH
C
C  EQUIVALENCE (DELFIN, H(1)), (DELX, H(1(51)), (TH, H(301)),
C  1 (NCELLS, H(451)), (NZONES, H(601))
C
C  FORMAT (1H1,109A0/)
C  1 J DX X(J), U(J), Y(J), YHL(J), CHL(J), DHL(J)
C  2 T(J), ZHL(J), EHL(J), MATERIAL COND J
C  3 /102H CM CM SEC CM CM SEC CM SEC CM CMS/GM
C  4 M3 DYN/CM2 GM/CM2 ERG/GM
C  5 FORMAT (14,1PE10.3,2X,A9,3R2,15)
C  6 FORMAT (29H TIME TO COMPLETE GENRAT IS F10.3,9H SECONDS.)
C  7 FORMAT(A4, A5, 1X, 1PE10.3)
C  8 FORMAT(A4, A5, 1X, 7E10.3)
C  9 FORMAT(1A10,7E10.3)
C  10 FORMAT(2A10,10.3, A10,110, A10,10.3)
C  11 FORMAT(2A10,10.3, 2A10,10.3))
C  12 FORMAT(1A10,10.3))
C  13 FORMAT(1A10,10.3, A5, A2, A5, 12, 3A10, A7)
C  14 FORMAT(A10,10.3, A5, A2, A5, 12, 3A10, A7)
C  15 FORMAT(4A10,10.3, A5, A2, A5, 12, 3A10, A7)
C  16 FORMAT (1AOA)
C  17 FORMAT(2A10,10.3, 2A10,10.3))
C  18 FORMAT(2A10,10.3, A5, A2, A5, 12, 3A10, A7)
C  19 FORMAT(2A10,10.3, A5, A2, A5, 12, 3A10, A7)
C  20 FORMAT(2A10,10.3, A5, A2, A5, 12, 3A10, A7)
C  21 FORMAT(2A10,10.3, A5, A2, A5, 12, 3A10, A7)
C  22 FORMAT(2A10,10.3, A5, A2, A5, 12, 3A10, A7)
C  23 FORMAT(2A10,10.3, A5, A2, A5, 12, 3A10, A7)
C  24 FORMAT (1AOA)
C  25 FORMAT(2A10,10.3, 2A10,10.3))
SUBROUTINE GENRAT (Continued)

26  FORMAT(A1, R9, 7A10, A5, A2, A5, 12, 3A10, A7)       GENRAT  46
27  FORMAT(1H+, 79X, A5, A2, A5, 12, 3A10, A7)
28  FORMAT(8A10)                                      GENRAT  47
31  FORMAT(1H+, 25X, 26H*** SRF PUFF 8 *** /)         GENRAT  48
31  FORMAT(1H+)                                       GENRAT  49
50  FORMAT(A10, * BOUNDAKY AT *A5, * SURFACE*)        GENRAT  50
1062 FORMAT(2A4, 12, 110, 3(2A5, E10.3))               GENRAT  51
1064 FORMAT(2A5, 110, 3(2A5, E10.3))                  GENRAT  52
1073 FORMAT(2A10, 110, A10, E10.3, A10, 1011)         GENRAT  53
1075 FORMAT(8E10.3)                                   GENRAT  54
1090 FORMAT(A1, A9, 2A10, E10.3, 2(A7, 311), A7, 13, A8, 12) GENRAT  55
62  FORMAT(2A4, 12, 110, 3(2A5, E10.3), A5, A2, A5, 12, 3A10, A7) GENRAT  56
64  FORMAT(2A5, 110, 3(2A5, E10.3), A5, A2, A5, 12, 3A10, A7) GENRAT  57
68  FORMAT(A10, 110, A10, 1015, A5, A2, A5, 12, 3A10, A7) GENRAT  58
67  FORMAT(10H **** ABORT FOLLOWING LINE DOES NOT FIT YIELD, MELT, VISCOSITY, OR SPALL CATEGORIES/8A10) GENRAT  59
96  FORMAT (31H ERROR JFIN EXCEEDS 301, JFIN=14)      GENRAT  60
97  FORMAT (*1 DATA BANK WITH HEADING -- *A10,--- ON FILE*) GENRAT  61
98  FORMAT (*0 DATA BANK WITH HEADING -- *A10,--- ON FILE*12/) GENRAT  62
100 DO 101 I = 1, 456                                  GENRAT  63
101 AZERO(I)=0.                                        GENRAT  64
103 X(I)=0.                                            GENRAT  65
105 A(I)=0.                                            GENRAT  66
109 EQSTA(I)=0.                                        GENRAT  67
111 EMELT(I)=0.                                        GENRAT  68
113 T65R(I)=0.                                         GENRAT  69
115 YO(I)=0.                                           GENRAT  70
117 EOS(I)=0.                                          GENRAT  71
119 IF (EOF(5)) 153, 154                                GENRAT  72
121 LVAR(I)=0.                                         GENRAT  73
122 LVAR(I)=0.                                         GENRAT  74
123 LVAR(I)=0.                                         GENRAT  75
124 LVAR(I)=0.                                         GENRAT  76
125 LVAR(I)=0.                                         GENRAT  77
126 LVAR(I)=0.                                         GENRAT  78
127 LVAR(I)=0.                                         GENRAT  79
128 LVAR(I)=0.                                         GENRAT  80
129 LVAR(I)=0.                                         GENRAT  81
131 LVAR(I)=0.                                         GENRAT  82
132 LVAR(I)=0.                                         GENRAT  83
133 LVAR(I)=0.                                         GENRAT  84
134 LVAR(I)=0.                                         GENRAT  85
135 LVAR(I)=0.                                         GENRAT  86
136 LVAR(I)=0.                                         GENRAT  87
137 LVAR(I)=0.                                         GENRAT  88
138 LVAR(I)=0.                                         GENRAT  89
139 LVAR(I)=0.                                         GENRAT  90
140 LVAR(I)=0.                                         GENRAT  91
141 LVAR(I)=0.                                         GENRAT  92
142 LVAR(I)=0.                                         GENRAT  93
143 LVAR(I)=0.                                         GENRAT  94
144 LVAR(I)=0.                                         GENRAT  95
145 LVAR(I)=0.                                         GENRAT  96
146 LVAR(I)=0.                                         GENRAT  97
147 LVAR(I)=0.                                         GENRAT  98
148 LVAR(I)=0.                                         GENRAT  99
149 LVAR(I)=0.                                         GENRAT  100
150 LVAR(I)=0.                                         GENRAT  101
151 LVAR(I)=0.                                         GENRAT  102
152 LVAR(I)=0.                                         GENRAT  103
153 LVAR(I)=0.                                         GENRAT  104
154 LVAR(I)=0.                                         GENRAT  105
155 LVAR(I)=0.                                         GENRAT  106
156 LVAR(I)=0.                                         GENRAT  107
157 LVAR(I)=0.                                         GENRAT  108
158 LVAR(I)=0.                                         GENRAT  109
159 LVAR(I)=0.                                         GENRAT  110
160 LVAR(I)=0.                                         GENRAT  111
161 LVAR(I)=0.                                         GENRAT  112
162 LVAR(I)=0.                                         GENRAT  113
163 LVAR(I)=0.                                         GENRAT  114
164 LVAR(I)=0.                                         GENRAT  115
165 LVAR(I)=0.                                         GENRAT  116
166 LVAR(I)=0.                                         GENRAT  117
167 LVAR(I)=0.                                         GENRAT  118
168 LVAR(I)=0.                                         GENRAT  119
169 LVAR(I)=0.                                         GENRAT  120
244
**SUBROUTINE GENRAT (Continued)**

155 READ (5,30) (A(I),I=1,8)
156 IF (EOF(5)) 152,156
157 DECODE (80,26,ITITLE) IND, (DISCPT(I),I=3,10)
158 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
159 READ (5,30) (A(I),I=1,8)
160 IF (IND .EQ. IDD) GO TO 158
161 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
162 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
163 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
164 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
165 IF (IND .EQ. IDD) GO TO 158
166 IF (IND .EQ. IDD) GO TO 158
167 IF (IND .EQ. IDD) GO TO 158
168 IF (IND .EQ. IDD) GO TO 158
169 IF (IND .EQ. IDD) GO TO 158
170 IF (IND .EQ. IDD) GO TO 158
171 IF (IND .EQ. IDD) GO TO 158
172 IF (IND .EQ. IDD) GO TO 158
173 IF (IND .EQ. IDD) GO TO 158
174 IF (IND .EQ. IDD) GO TO 158
175 IF (IND .EQ. IDD) GO TO 158
176 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
177 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
178 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
179 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
180 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
181 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
182 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
183 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
184 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
185 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
186 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
187 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
188 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
189 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
190 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
191 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
192 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
193 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
194 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
195 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
196 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
197 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
198 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
199 READ (IN,26) A(I), (TEDIT(I+NZ),I=1,NT)
200 READ (IN,1090) A(I), (TEDIT(I+NZ),I=1,NT)
SUBROUTINE GENRAT (Continued)

1  NPOR(M), A3, NDS(M), NPR(M), NYAM, A4, NVAR(M), A5, NCON(M)  GENRAT  196
2  , NIND, IND, NIN, IN, N5T  GENRAT  197
1  IF (IN .EQ. 4) OR (IND .EQ. 1DD) GO TO 205  GENRAT  198
IN=4  GENRAT  199
CALL REDR(MATL(M,1), MATL(M,2), IN,2)  GENRAT  200
1  IF (IND .EQ. 1HT) GO TO 200  GENRAT  201
READ(IN,1090)IND,A1,A2,A3,A4,A5,N1,N2,N3,A6,N4,N5,NYAMT,A7,A8,N6  GENRAT  202
WRITE (6,90) IDD, A1, A2, A3, A4, A5, N1, N2, N3, A6, N4, N5, NYAMT, A7, A8, N6,  GENRAT  203
1  NIND, IND, NIN, IN, N5T  GENRAT  204
205  RHO(M)=RHOS(M)  GENRAT  205
1  IF (NCMF(M) .NE. 0) GO TO 2055  GENRAT  206
C  ** READ EQST VARIABLES ****  GENRAT  207
READ(IN,1020)A1,EQSTC(M), EQSTD(M), EQST(E), EQST(G), EQSTH(M),  GENRAT  208
1  EQSTS(M), EQSTV(M)  GENRAT  209
WRITE (6,20) A1, EQSTC(M), EQSTD(M), EQST(E), EQST(G), EQSTH(M),  GENRAT  210
1  EQSTS(M), EQSTV(M), NIND, IDD, NIN, IN, NFT, NGT, NHT, NIT  GENRAT  211
1  IF (A1 .EQ. 10H EQSTX=  ) READ (IN,1020) A2, EQSTA(M), A3, A4, A5, A6,  GENRAT  212
1  A7, A8  GENRAT  213
1  IF (A1 .EQ. 10H EQSTX=  ) WRITE (6,20) A2, EQSTA(M), A3, A4, A5, A6,  GENRAT  214
1  A7, A8, NIND, IDD, NIN, IN  GENRAT  215
LS = -1  GENRAT  216
EMELT(M)=0.1*EQST(E)  GENRAT  217
CALL FMELT(LS,M, EMELT(M), A1, A2, X, MS)  GENRAT  218
1  IF (NCMF(M) .EQ. 0) GO TO 2059  GENRAT  219
C  **** READ COMPOSITE DATA ****  GENRAT  220
2055 CONTINUE  GENRAT  221
LS=-1  GENRAT  222
CALL REBAR(LS, IN, J, I, M, N, H(J,3), RHOS(M), DOLD, EXMAT(M,3), SY, SZ,  GENRAT  223
1  TXY, EH, PHL(J), EX, EY, EZ, EXY, F, O, O, ESC, FS, COM(1), COM(2), COM(6),  GENRAT  224
2  COM(7), YO(M), COM(8), 1PRT)  GENRAT  225
NVAR(M)=MAXO(NVAR(M),7)  GENRAT  226
GO TO 245  GENRAT  227
2059 CONTINUE  GENRAT  228
C  **** READ FRACTURE DATA ****  GENRAT  229
NFRM=NFR(M)+1  GENRAT  230
GO TO (210,206,207,208,208,207) NFRM  GENRAT  231
206 CONTINUE  GENRAT  232
207 CONTINUE  GENRAT  233
READ(IN,1020)A1,(TSR(M,1), N1=1,7)  GENRAT  234
WRITE (6,20) A1, (TSR(M,1), N1=1,7), NIND, IDD, NIN, IN  GENRAT  235
1  IF (NFR(M) .EQ. 0) GO TO 210  GENRAT  236
1  IF (NFR(M) .EQ. 2) NVAR(M)=MAXO(NVAR(M),18)  GENRAT  237
1  IF (NFR(M) .EQ. 5) NVAR(M)=MAXO(NVAR(M),11)  GENRAT  238
READ(IN,1020)A1, (TSR(M,1), N1=6,14)  GENRAT  239
WRITE (6,20) A1, (TSR(M,1), N1=6,14), NIND, IDD, NIN, IN  GENRAT  240
GO TO 210  GENRAT  241
C  READ FOR SHEAR BAND MODEL.  GENRAT  242
208 CALL SHEAR2(LSUB(15), IN,M)  GENRAT  243
NVAR(M)=MAXO(NVAR(M),5)  GENRAT  244
LSUB(15)=1  GENRAT  245
1  IF (NFR(M) .EQ. 4) GO TO 207  GENRAT  246
C  **** READ POROUS DATA ****  GENRAT  247
210 IF (NPOR(M) .EQ. 0) GO TO 230  GENRAT  248
NPORM = NPOR(M)  GENRAT  249
GO TO (21,212,225,225,227) NPORM  GENRAT  250
211 READ(IN,1020)A1, RHO(M)  GENRAT  251
WRITE (6,20) A1, RHO(M)  GENRAT  252
CALL POREGST(0, IN, M, EXMAT(M,3), RHO(M), A2, A3, A4, A5, A6, CZQ(M), CWQ(M),  GENRAT  253
1, A7, A8, EQSTC(M), EQSTD(M), EQST(E), EQST(G), EQSTH(M), EQSTS(M), A11, A12, YO(M)  GENRAT  254
GO TO 230  GENRAT  255
212 IF (NPOR(M) .GT. 2) GO TO 225  GENRAT  256
CALL PORHOLT(0, IN, M, EXMAT(M,3), RHO(M), DOLD, A1, A2, A3, A4, A5, A6, A7,  GENRAT  257
1, EQSTC(M), A9, YO(M), RHOS(M), A10)  GENRAT  258
GO TO 230  GENRAT  259
225 READ(5,1020)A1, RHO(M)  GENRAT  260
WRITE (6,20) A1, RHO(M)  GENRAT  261
CALL PEST(LSUB(14), 5, A1, A2, A3, A4, A5, M, EXMAT(M,3), RHO(M), A6, RHOS(M)  GENRAT  262
1, A7, A8, A9, A10, A11, A12, A13, EQSTC(M), EQSTD(M), EQST(E), EQST(G),  GENRAT  263
2, EQSTS(M), EQSTH(M), EQSTV(M), EQSTH(M), EQSTE(M), EQSTN(M), EQSTV)  GENRAT  264
3, EQSTA(M))  GENRAT  265
246
SUBROUTINE GENRAT (Continued)

NVAR(M)=MAXO(NVAR(M),5)
GO TO 230

227 READ(5,1021)A1,RH0(M),A2,MU(M)
WRITE(6,21)A1,RH0(M),A2,MU(M)
M0(M)=1.333*MU(M)
CALL EXPLODE(M,IN,M,NVAR,M),EQSTC(M),EQSTG(M),RHOS(M),SHL(1),SHL(2),SHL(3),SHL(4),NEM(1),
2 K,J,NET(1))

C NYAM IS THE NUMBER OF CARDS
C

230 IF (NDS(M) .EQ. 0) GO TO 235
236 CALL EQSTPF(0,IN,M,M0(M))
WRITE(6,20)A1,IN,M,NPRM
GO TO 235

233 CALL EP0(M)
235 IF (NPRM .EQ. 0) GO TO 245
NPRM = NPRM
GO TO (236,237,238,239,240,241,245) NPRM

236 CALL EXPLODE(1,IN,M,EXMAT(M,3),A1,A2,A3,A4,A5,A6,A7,A8,EXMAT(M,4)
GO TO 245

237 CALL ESA0(0,IN,M)
GO TO 245

238 CALL EQSTPF(0,IN,M)
GO TO 245

239 CALL HYPO0(0,IN,M,EXMAT(M,3),RHOS(M))
GO TO 245

240 CALL GRAY0(0,IN,M)
GO TO 245

241 CALL EQSTAB(0,IN,XN,YN,ZN)
GO TO 245

C NYAM IS THE NUMBER OF CARDS

245 IF (NYAM .EQ. 0) GO TO 260
DO 275 NY=1,NYAM
READ (IN,30)(X(I),I=1,8)
DECODE (10,24,X)(A(I),I=1,10)
DO 250 I=1,10
IF (A(I) .EQ. 1.0) GO TO 260
IF (A(I) .EQ. 1.1) .AND. A(I+1) .EQ. 1.2) GO TO 260
IF (A(I) .EQ. 1.3) .OR. A(I+1) .EQ. 1.3) GO TO 260
IF (A(I) .EQ. 1.4) .OR. A(I+1) .EQ. 1.4) GO TO 260
IF (A(I) .EQ. 1.5) .OR. A(I+1) .EQ. 1.4) GO TO 260
IF (A(I) .EQ. 1.6) .AND. A(I+1) .EQ. 1.6) GO TO 260
IF (A(I) .EQ. 1.7) .AND. A(I+1) .EQ. 1.7) GO TO 260
IF (A(I) .EQ. 1.8) .AND. A(I+1) .EQ. 1.8) GO TO 260
CONTINUE

250 PRINT 67, (X(I),I=1,8)
GO TO 298

252 DECODE(60,1020,X)A1,(TENS(M,1),I=1,3)
WRITE(6,20)A1,TENS(M,1),I=1,3), (T(I),I=1,4), NIND,IDD,NIN,NF
GO TO 275

253 DECODE(60,1020,X)A1,CZQ(M),CQW(M),C2(M)
WRITE(6,20)A1,CZQ(M),CQW(M),C2(M), (T(I),I=1,4), NIND,IDD,NIN,NF
GO TO 275

256 DECODE(60,1020,X)A1, YOS,MU(M),YADD(M),EXMAT(M,1),EXMAT(M,4)
WRITE(6,20)A1,YOS,MU(M),YADD(M),EXMAT(M,1),EXMAT(M,4), (T(I),I=1,2)

1, NIND,IDD,NIN,NF,NNT,JNT,NJT,JNT
IF (NDS(M) .EQ. 5) YADD(M)=YADD(M)/(RHOS(M)X(1+0.5*YOS/MU(M)))
IF (NPO(M) .EQ. 0) YOH(M)=YOS

C TEST FOR COULOMB FRICTION MODEL
C

257 IF (EXMAT(M,1) .EQ. 0) GO TO 275
C

265 DECODE(60,1020,X)A1,SPH(M)
WRITE(6,20)A1,SPH(M)
GO TO 275

267 LS = 0
GO TO 273

272 LS = 1

273 DECODE(60,1020,X)A1,(A(I),I=1,7)
WRITE(6,20)A1,(A(I),I=1,7), NIND,IDD,NIN,NK
SUBROUTINE GENRAT (Continued)

IF (MS .EQ. M) GO TO 2732
DO 2731 L = 1, 7
IF (LS .EQ. O) EMELT(M, L) = A(I)
IF (LS .EQ. 1) EMELT(M, L) = A(I)
2731 CONTINUE
GO TO 2733
2732 IF (LS .EQ. O) EMELT(M, 8) = A(I)
IF (LS .EQ. 1) EMELT(M, 8) = A(I)
2733 CALL EMELT(LS, M, EMELT(M), A1, A2, A, MS)
GO TO 275
256 DECODE(80, 1020, X) A1, (THERM(M), L = 1, 5)
WRITE (6, 20) A1, (THERM(M), L = 1, 5), NIND, IDD, NIN, IN
275 CONTINUE
C
**** READ IN EDGE VARIABLES
280 IF (NCON(M) .GT. 0) AND. MATFL .EQ. 0) CALL DEPOS(1, IN)
ESC(M, 1) = RHO(M) * ESC(M, 2) = EQSTC(M)
ESC(M, 3) = EQSTD(M) * ESC(M, 4) = EQSTG(M)
ESC(M, 5) = MU(M) * ESC(M, 6) = YADD(M)
ESC(M, 7) = RHOS(M) * ESC(M, 8) = EQSTG(M)
ESC(M, 9) = EMELT(M)
THERM(M, 6) = EMELT(M)
THERM(M, 8) = EQSTG(M)
CONTINUE
C
**** END OF M-LOOP ****
WRITE (6, 69)
C
**** READ IN ZONING VARIABLES ****
C
DO 291 L = 1, 30
JBLN(L) = 0
291 THK(L) = 0.
IN = 5
READ (5, 66) A1, NLAYER, A2, (JMAT(L), L = 1, 10)
WRITE (6, 66) A1, NLAYER, A2, (JMAT(L), L = 1, 10), NIND, IDD, NIN, NLT
IF (NLAYER .LE. 10) GO TO 292
READ (5, 80) (JMAT(L), L = 11, NLAYER)
WRITE (6, 80) (JMAT(L), L = 11, NLAYER)
292 INF = INF = 0
IF (JMAT(1) .LT. 0) INF = 1
IF (JMAT(NLAYER) .LT. 0) INF = 1
JMAT = IABS(JMAT(1))
JMAT = IABS(JMAT(NLAYER))
READ (5, 30) (X(1), I = 1, 8)
DECODE(4, 1062, X) A1
IF (A1 .NE. 4H THK) GO TO 293
DECODE(80, 1019, X) A1, A2, (THK(L), L = 1, 7)
WRITE (6, 19) A1, A2, (THK(L), L = 1, 7)
IF (NLAYER .GT. 7) GO TO 292
READ (5, 1075) (THK(L), L = 8, NLAYER)
WRITE (6, 75) (THK(L), L = 8, NLAYER)
2921 IF (A2 .NE. 5H INCH) GO TO 399
C
CONVERSION OF THK(L) FROM INCHES TO CM
DO 2922 L = 1, NLAYER
2922 THK(L) = 2.54 * THK(L)
GO TO 399
293 DECODE(80, 1062, X) A1, A2, NZONES(L), NCELLS(L, 1), A3, A4, TH(L, 1), A5, A6,
1 DELX(L, 1), A7, A8, DELFIN(L, 1)
DO 300 L = 1, NLAYER
IF (L .GT. 1) READ (5, 1062) A1, A2, NZONES(L), NCELLS(L, 1), A3, A4, TH(L, 1),
1 A5, A6, DELX(L, 1), A7, A8, DELFIN(L, 1)
WRITE (6, 62) A1, A2, NZONES(L), NCELLS(L, 1), A3, A4, TH(L, 1), A5, A6, DELX
1 (L, 1), A7, A8, DELFIN(L, 1), NIND, IDD, NIN, IN, NLT
NZON = NZONES(L)
IF (NZON .EQ. 1) GO TO 2951
DO 295 N = 2, NZON
READ (5, 1064) A1, A2, NCELLS(L, N), A3, A4, TH(L, N), A5, A6, DELX(L, N),
1 A7, A8, DELFIN(L, N)
295 WRITE (6, 64) A1, A2, NCELLS(L, N), A3, A4, TH(L, N), A5, A6, DELX(L, N),
1 A7, A8, DELFIN(L, N), NIND, IDD, NIN, IN, NLT
2951 IF (A5 .NE. 5H INCH) GO TO 300
C
CONVERSION OF TH(L, N), DELX(L, N) FROM INCHES TO CM
DO 2952 N = 1, NZON
248
SUBROUTINE GENRAT (Continued)

DO 395 L=1,NLAYER
   IF (JMAT(L).EQ.0) GO TO 393
   JBOUND(L-NULL)=JBOUND(L)
   JMAT(L-NULL)=JMAT(L)
   GO TO 395
393 NULL=NULL+1
395 CONTINUE
   Nlayer=NlayerNULL
396 IF (JFIN.LE.201) GO TO 399
   WRITE (6,96) JFIN
   PRINT 30, (A(I), I=1,8)
   IF (EOF(5)) 100, 398
C **** READ RADIATION SOURCE DATA ****
399 IF (MATFL.EQ.0) CALL DEPS0(2,IN)
C **** INITIALIZE THE J-ARRAY VARIABLES ****

DO 601 I=1,2400
   CHL(I)=0.
   J=1
   LMAX=1
   DO 630 L=1,NLAYER
      M=JMAT(L)
      YOM=Y0(M)
      IF (NPOR(M).NE.0) GO TO 602
      IF (NPR(M).EQ.4) GO TO 602
      DET=EXMAT(M,3)
      CJ=AMAX1(DET,SORT((EQTC(M)+1.333*MU(M))/RHOS(M)),5,E4)
      HH=1.SOLID $ CJ=EXMAT(M,3)
      IF (RH0(M).EQ.RHOS(M)) HH = SOLID
500 HH=0
501 HH=POISSON $ CJ=EXMAT(M,3)
502 IF (NPOR(M).EQ.1) CALL EXPLODE(2,IN,M,EHL,DHL,DOLD,PHL,SHL,NEM,X, GENRAT 495
SUBROUTINE GENRAT (Continued)

1  J, A1, A2, A3)
IF (NVAR(M) .LE. 0) GO TO 604
LVAR(J)=LMAX
LMAX=LMAX+NVAR(M)
604 CONTINUE
IF (NPOR(M) EQ. 3) COM(LMAX+3)=1.-RHO(M)/RHOS(M)
IF (J .NE. J1) H(J,2)=NORMAL
H(J,3)=2
T(J)=TENS(M,1)
YHL(J)=YOM
ZHL(J)=HDL(J)*(X(J+1)**NALPHA-X(J)**NALPHA)
GO TO (610,610,605,606,610,607,605,610) NDSM1
605 NEM(J)=TSR(M,21) $ GO TO 610
606 NEM(J)=TSR(M,19) $ GO TO 610
607 NEM(J)=YHL(J)
610 CONTINUE
H(J1,2)=RINTER
IF (J1 .EQ. 1) GO TO 620
IF (X(J1) .GT. X(J1-1)) H(J1,2)=SPALL
CONTINUE
T(JN)=TENS(M,3)
H(JN,2)=LINTER
J1=JN+1
CONTINUE
ZHL(JFIN-1)=0.
H(1,2)=H(JFIN,2)=SPALL
IF (NVAR .EQ. 1) H(1,2)=INF
IF (NVAR .EQ. 1) H(JFIN,2)=INF
IF (INFF .EQ. 1) ZHL(INFF-1)=2HL(1)
ACTIVATE THIS ROUTINE, DTMAX IS NEGATIVE OF NUMBER OF CELLS
DESIRED IN LAYER NUMBER(-NREZON)
IF (DTMAX .GT. 0. .OR. NREZON .GE.O)  GO TO 635
JB=JBND(-NREZON)
X1=0.
IF (NREZON .EQ. -1) GO TO 632
JB1=JBND(-NREZON-1)
X1=X(JB1)
632 DTMAX=(X(JB)-X1)/(CHL(JB-1)*DTMAX)
NREZON=-30
CONTINUE
635 CONTINUE
DTNH=1.E-12
CHECK FOR END OF DATA DECK AND CALL FOR ADDED READS
INSERT CARD HERE READING EXTRA
READ 30, A1
IF (EOF(5)) 650, 640
640 IF (A1 .EQ. 10H H-DATA ) GO TO 642
IF (A1 .EQ. 10H EXTRA ) GO TO 645
GO TO 398
642 CALL HDATA(H)
GO TO 638
645 CALL EXTRA
650 CONTINUE
IF (MATFL) 815, 700, 800
C DEPOSITION EDIT
700 CALL DEPOS(3, IN)
GO TO 900
C INITIALIZE VELOCITY
800 JFIN2=JBND(MATFL)
IF(UZERO .EQ. 0.) JFIN2=2
DC 810 J=1, JFIN2
810 U(J)=UZERO
DTNH=0.02*AMIN1((X(JFIN2)-X(JFIN2-1))/CHL(JFIN2-1),
(X(JFIN2)-X(JFIN2+1))/CHL(JFIN2+1))
JSTAR=JFIN2+3 $ SDURM=1. $ GO TO 818
815 IF (MATFL+2) 817, 816, 8151
8151 H(1,2)=MIRROR $ JSTAR=3 $ SDURM=1. $ U(1)=0.5*UZERO
GO TO 818
816 JSTAR=3 .SDURM=1. $ H(JFIN,2)=PRESS
GO TO 818
817 JSTAR=JFIN $ SDURM=1. $ H(JFIN,2)=PRESS
C VELOCITY EDIT
818 IF (H(1,2) .EQ. SPALL) GO TO 819
A2=5HFRONT

C
SUBROUTINE GENRAT (Concluded)

A1=10H  UNKNOWN
IF (H(1,2) .EQ. MIRROR) A1=10H  MIRROR
IF (H(1,2) .EQ. INF) A1=10H  INFINITE
IF (H(1,2) .EQ. PRESS) A1=10H  PRESS
IF(INFF .EQ. 1) U(JFIN+1)=U(1)
PRINT 50,A1,A2

819  IF (H(JFIN,2) .EQ. SPALL) GO TO 8195
A2=5HREAR  $  A1=10H  UNKNOWN
IF(H(JFIN,2) .EQ. INF) A1=10H  INFINITE
IF(H(JFIN,2) .EQ. PRESS) A1=10H  PRESS
PRINT 50,A1,A2

8195  WRITE (6,16) (DISCPT(I),I=1,10)
      IF (EHL(J) .GT. 1.) JSTAR=MAX0(JSTAR,J)
DO 820  J=1,JFIN

820  A(J)=X(J+1)-X(J)
L=K=J1=1
825  J2=MIN0(JFIN-1,50*K,JBND(L))
M=JMST(L)
WRITE (6,17) (J,A(J),X(J),U(J),YHL(J),CHL(J),DHL(J),T(J),ZHL(J),
1 EHL(J),MATL(M,1),(H(J,1),I=1,3),J,J=J1,J2)
IF (J2 .EQ. JFIN-1) GO TO 900
J1=J2+1
IF (J2 .NE. 50*K) GO TO 830
K=K+1  $  WRITE (6,16) (DISCPT(I),I=1,10)

830  IF (J2 .NE. JBND(L)) GO TO 825
L=L+1  $  WRITE (6,69)  $  GO TO 825
900  CALL SECOND(TWIX)  $  DUR=TWIX-FIRST
WRITE (6,18) DUR
WRITE (6,41)
IF (JCYCS .LE. 0 .OR. LSUB(7) .EQ. 1 ) GO TO 100
C  ****  PREPARE FOR STORAGE OF HISTORIES  ****
C
RETURN
END
SUBROUTINE GRAY(NPART, IN, AMU, EMELT, D, E, CH, P, DD, DE, IH)

GRAY 3-PHASE EOS OF ROYCE AT LLL. REF UCRL-51121
GRAY MODIFIED IN THE LIQUID-VAPOR REGION BY YOUNG AT LLL.
GRAY REF. UCRL-51575

DIMENSION A(10), ALFLS(10), AYBLS(10), C(10), CE2(10), CTA2(10),
1 CTB1(10), CTB2(10), CSLS(10), C2LS(10), CSLS(10), DLS(10), D1LS(10),
2 DLC(10), DLC(10), E00(10), G0(10), GPLS(10), PCCLS(10), RPLS(10),
3 S(10), TH(10), TM0(10), VB(10), VO(10), XJ(10), ZJ(10),
4 DIMENSION TEMP(50), PRES(50), VMN(50), VMX(50), EMN(50),
5 DIMENSION ALFLS(L), AYBLS(L), C(L), CE2(L), CTA2(L),
6 CTB1(L), CTB2(L), CSLS(L), C2LS(L), CSLS(L), DLS(L), D1LS(L),
7 DLC(L), DLC(L), E00(L), G0(L), GPLS(L), PCCLS(L), RPLS(L), S(L),
8 S(L), TH(L), TM0(L), VB(L), VO(L), XJ(L), ZJ(L),
9 DIMENSION TEMP(L), PRES(L), VMN(L), VMX(L), EMN(L),
10 IF (NPART < 6) GO TO 40

C **** READ DATA AND PRINT ****
READ 1000, A1, AMU, EMELT, D, E, CH, P, DD, DE, IH
1 A2, CTB2, CSLS, C2LS, CSLS, DLS, D1LS, DLC, DLC,
2 A3, DLC(10), E00(10), G0(10), GPLS(10), PCCLS(10), RPLS(10), S(10),
3 A4, TH(10), TM0(10), VB(10), VO(10), XJ(10), ZJ(10),
4 PRINT 1001, A1, JMX(M)
5 FORMAT(A10, 1P7E10.3)
6 READ 1001, A1, JMX(M)
7 PRINT 1001, A1, JMX(M)
8 FORMAT(A1, O10)
9 IMAX=JMX(M)
10 DO 30 I=1, IMAX
11 READ 1000, A1, TEMP(I, M), VMX(I, M), VMN(I, M), EMX(I, M), EMN(I, M), PRES(I, M)
12 PRINT 1000, A1, TEMP(I, M), VMX(I, M), VMN(I, M), EMX(I, M), EMN(I, M), PRES(I, M)
13 1 CONTINUE
14 RETURN
C C **** COMPUTE PRESSURE, SOUND SPEED, (DP/DD)E, AND (DP/DE)V
15 D1=D
16 E1=E
17 NLOOP=0
18 NLOOP=NLOOP+1
19 IF (NLOOP < 3) GO TO 90
20 PR1=P
21 D2=D1+.001*D1
22 GO TO 100
23 PR2=P
24 D=D1
25 EX=E1+.001*E1
26 100 X=(VO(M)-1)/D)/VO(M)
27 V=1/D
28 IF (V*LE. 1.04*VO(M)) GO TO 145
29 EN=EMX(2, M)
30 IF (E*GE. 1.2*EN) GO TO 140
31 C USE CRITICAL POINT AND TIE LINES TO REPLACE VAN DER WAALS LOOPS
32 C LIQUID-VAPOR PHASE.
33 VXM1=VM1+VMN(1, M)
34 EM1=EM1+EMN(1, M)
35 EQ=EM1*(EN-EMN(2, M))*(V-VXM1)/(VXM(2, M)-VMN(2, M))
36 JMAX=JMX(M)
37 DO 110 J=2, JMAX
38 VX=VMX(J, M)
39 EX=EMX(J, M)
40 VN=VMN(J, M)
41 EN=EMN(J, M)
42 EQ=EN*(EX-EN)*(V-VN)/(VX-VN)
43 110 CONTINUE
44 RETURN
C
SUBROUTINE GRAY (Continued)

IF (E .LT. EQ) GO TO 105
EH=EN*(EMM1-EN)/(VMM1-VN)
IF (E .GE. EH) GO TO 140

105  EQM=EQ
VM1=VN
EXM1=EX
EMM1=EN

110  CONTINUE

J=JMAX
EL=EX*VX/V
IF (E .GE. EL) GO TO U0

115  NAME=10H LIQ-VAPOR
GO TO 120

140  IF (X .LT. XJ(M)) GO TO 60
C  START COMPUTATIONS FOR SOLID-LIQUID STATES (S)

145  EO=(C(M)*X)**2/2/((1.*S(M)*X)*((1.*S(M)*X)/3.+CE2(M)*X)**X)

1  EO0=M(1.*GO(M)**X)
G=GO(M)-A(M)**X
PL=(C(M)/S(M)*X)**2*G*D*(1.-X-0.5*G*X) + G*E*D
IF (X .GE. 0.) GO TO 150
TM=TM0(M)/(1.-X)**2*(1.*(CTB1(M)-2.*X)*X/CTA2(M)**X)
GO TO 155

150  TM=TM0(M)*((1.*CTB1(M)-A(M)*X)**2/2/((1.*S(M)*X)**X))

155  CONTINUE

DT=DSLS(M)*(CTB1(M)-0.5*RPLS(M)+0.5*GPLS(M)**TMDT)
C  BRANCH POINT
IF (E .GE. EM1) GO TO 300
C

C  ***  SOLID EQUATION OF STATE  ***

200  QUAD=Q.**RPLS(M)**2+2.*GPLS(M)**(E-EO)
IF (QUAD .LT. 0.) GO TO 800
T=-(Q**RPLS(M)+SQRQ(QUAD))/GPLS(M)
P=Q**RPLS(M)+GPLS(M)**T**2.*D/2.
C  BRANCH POINT
IF (E .GE. EM2) GO TO 400
C
SUBROUTINE GRAY (Concluded)

C ***** MELT EQUATION OF STATE
ENU*(E-EM1)/(EM2-EM1)
ENUS=ENU*(DSLS(M)-ALFLS(M))
RNU=3.*RPLS(M)*ENUS
QUAD=RNU**2.*GPLS(M)*(E-EO*ENNOL*EM1)
IF (QUAD*LT.0.) GO TO 800
T=(-RNU*SQRT(QUAD))/GPLS(M)
PC=D**(0.5*(0.666667-G))/GAT
P=P1+P2*PCCLS(M)
IM=6R M
GO TO 700

400 EGG=EO+TM*(28.7*RPLS(M)+DSLS(M)+64.017*GAT)
EGG=EO+TM*64.017*ALFLS(M)
IF (EGG .LT. EMT) GO TO 500

C ****** BRANCH POINT
IF (E .GT. EGG) GO TO 500

C ****** LIQUID EQUATION OF STATE
GAT=GPLS(M)-ALFLS(M)/TM
QUAD=(3.*RPLS(M)-9.593*ALFLS(M))
IF (QUAD .LT. 0.) GO TO 800
T=(3.*RPLS(M)-9.593*ALFLS(M)+SQRT(QUAD))/GAT
PC=D**(0.5*(0.666667-G))/GAT
P=P1+PC*PCCLS(M)
GO TO 700

C ****** HOT LIQUID EQUATION OF STATE
GAT=GPLS(M)-ALFLS(M)/TM
QUAD=(3.*RPLS(M)-9.593*ALFLS(M))
IF (QUAD .LT. 0.) GO TO 800
T=(3.*RPLS(M)-9.593*ALFLS(M)+SQRT(QUAD))/GAT
PC=D**(0.5*(0.666667-G))/GAT
P=P1+PC*PCCLS(M)
GO TO 700

C ****** LIQUID-VAPOR EQUATION OF STATE
ZH=Z
PP=P1*P2*P2CLS(M)
IH=6R V
GO TO 700

600 Z=Z
FE=0.5*Z/((TH(M)-Z)/(TH(M)-Z))**2*(Z**2+TH(M))
1 =-(2.5*Z/((TH(M)-Z)/(TH(M)-Z))**2+TH(M))**3
QUAD=(3.*RPLS(M)+2.*(C3LS(M))**2-16.*C3LS(M)**2)
DLS(M)**2+E*AYBS(M)**Z-C1LS(M)**FE
1 =-(DLS(M)**2+C3LS(M)**2+E*AYBS(M)**Z-C1LS(M)**FE)
IF (QUAD .LT. 0.) GO TO 800
T=(3.*RPLS(M)+2.*C3LS(M)**2-16.*C3LS(M)**2)/C3LS(M)**2+E*AYBS(M)**Z-C1LS(M)**FE
PP=P1*P2*P2CLS(M)
IH=6R V
GO TO 700

700 IF (NLOOP .LT. 3 .AND. CH .NE. 1.) GO TO 50
PR3=P
DPDD=(PR2-PR1)/(D2-D1)
DPDE=(PR3-PR1)/(E3-E1)
CH2=DPDD*P1*DPDE/D1**2*1.333*AMU/D1
IF (CH2 .GE. 1.333*AMU/D1)
E=E1
D=D1
IF (E .GE. EMT) P=AMAX1(P,0.)
RETURN

800 CONTINUE
IM=6R Z
P=0.
RETURN
END
SUBROUTINE HAFSTEP

C  * CALLED BY HYDRO TO COMPUTE X, U, D, E FOR THE
  * HALFSTEP POINT BETWEEN J AND J+1
C  INPUT - J, M.
C  OUTPUT - UHL, DHL, EHL.

INTEGER H, POROUS, PRESS, RINTER, SOLID, SPALL
REAL MATL, NEM, NET, NEMH, NETH
C  MISCELLANEOUS
COMMON AZERO(1), CEF, CKS, DAVO, DELTM, DISCPT(10), DOLD, DRHO, DTMAX,
  1 DTMIN, DTN, DTHM, DU, DX, EOLD, F, FAC, FIRST, J, JCYCS, JINIT,
  2 JFIN, JREZON(15), JSMAX, JSTAR, JTS, LSUB(30), M, MAXPR(30), N, NCYCS,
  3 NEDIT, NPERN, NR, NREZON, NSCRB(6), NSPRAF, NSPALL, NTEDT,
  4 NTEX, NTR(15), NOLD, P(20), R(30), RLAST, SLAST, SMAX, TEDIT(50),
  5 TF, TIME, TJ, TREZON, TS, T(20), ULAST, UOLD, UZERO, XLAST, XNOW, XOLD
  6 , XJIDT(20), MS
C  HALFSTEP VALUES
COMMON DH, DHLAST, DUH, EH, PH, RH, RHLAST, SH, SLAST, UH, UHLAST, XH, XHLAST
  1 , NEMH, NETH
C  CONDITION INDICATORS
COMMON INF, LINTER, MIRROR, NORMAL, POROUS, PRESS, RINTER, SOLID, SPALL
C  CELL LAYOUT
COMMON DX(30), JBOUND(30), JMAT(30), NAUTO, MATL(6, 2), NLLAYR, NMTRLS,
  1 THK(30)
C  COORDINATE ARRAYS
COMMON/COORD/X(200), X0(200), CHL(200), DHL(200), DPDD(200), DPDE(200),
  1 EHL(200), H(200, 3), NEM(200), NET(200), PHL(200), RHL(200), SDT(200),
  2 SHL(200), T(200), U(200), YHL(200), ZHL(200)
COMMON /NAMED COMMON
REAL MU, MUM
COMMON /EQLS/ EQS(6), EQSTC(6), EQSTD(6), EQSTE(6), EQSTO(6),
  1 EQSTH(6), EQSTN(6), EQSTV(6), CQO(6), CWO(6), C2(6)
COMMON /MELT/ EMELT(6, 8), GMELT(6, 8), PHM(6, 8), THERM(6, 8),
  1 THERM(6, 8)
COMMON /RHOS/ RHOS(6), RHOS(6),
COMMON /T/ TS(6, 30), Tremain(6, 20), TEN(6, 3),
COMMON /Y/ YO(6), YADD(6), MU(6), MUM, YADM
COMMON /IND/ INDS(6), INDP(20), NALPHA, NCMP(6), NFR(6), NPER(6),
  1 NDS(6), NPER(6), NCON(6), NVAR(6)
COMMON /RAD/ SSTOP(9), SSTART(9), SDURM, SSTOPM, NSPEC, SSJ, JSS, IPLOT(4)
  1 , XMAX(4), XMIN(4), YMAX(4), YMIN(4), IA(7), ITITLE24, NARZ, TARZ
  1 , REAL MATL, NEM, NET, NEMH, NETH
  1 , COMMON AZERO(1), CEF, CKS, DAVO, DELTM, DISCPT(10), DOLD, DRHO, DTMAX,
  1 DTMIN, DTN, DTHM, DU, DX, EOLD, F, FAC, FIRST, J, JCYCS, JINIT,
  2 JFIN, JREZON(15), JSMAX, JSTAR, JTS, LSUB(30), M, MAXPR(30), N, NCYCS,
  3 NEDIT, NPERN, NR, NREZON, NSCRB(6), NSPRAF, NSPALL, NTEDT,
  4 NTEX, NTR(15), NOLD, P(20), R(30), RLAST, SLAST, SMAX, TEDIT(50),
  5 TF, TIME, TJ, TREZON, TS, T(20), ULAST, UOLD, UZERO, XLAST, XNOW, XOLD
  6 , XJIDT(20), MS
C  HALFSTEP VALUES
C  COMMON DH, DHLAST, DUH, EH, PH, RH, RHLAST, SH, SLAST, UH, UHLAST, XH, XHLAST
  1 , NEMH, NETH
C  CONDITION INDICATORS
C  COMMON INF, LINTER, MIRROR, NORMAL, POROUS, PRESS, RINTER, SOLID, SPALL
C  CELL LAYOUT
C  COMMON DX(30), JBOUND(30), JMAT(30), NAUTO, MATL(6, 2), NLLAYR, NMTRLS,
C  THK(30)
C  COORDINATE ARRAYS
C  COMMON/COORD/X(200), X0(200), CHL(200), DHL(200), DPDD(200), DPDE(200),
C  EHL(200), H(200, 3), NEM(200), NET(200), PHL(200), RHL(200), SDT(200),
C  SHL(200), T(200), U(200), YHL(200), ZHL(200)
C  NAMED COMMON
REAL MU, MUM
C  COMMON /EQLS/ EQS(6), EQSTC(6), EQSTD(6), EQSTE(6), EQSTO(6),
C  EQSTH(6), EQSTN(6), EQSTV(6), CQO(6), CWO(6), C2(6)
C  COMMON /MELT/ EMELT(6, 8), GMELT(6, 8), PHM(6, 8), THERM(6, 8)
C  COMMON /RHOS/ RHOS(6), RHOS(6),
C  COMMON /T/ TS(6, 30), Tremain(6, 20), TEN(6, 3),
C  COMMON /Y/ YO(6), YADD(6), MU(6), MUM, YADM
C  COMMON /IND/ INDS(6), INDP(20), NALPHA, NCMP(6), NFR(6), NPER(6),
C  NDS(6), NPER(6), NCON(6), NVAR(6)
C  COMMON /RAD/ SSTOP(9), SSTART(9), SDURM, SSTOPM, NSPEC, SSJ, JSS, IPLOT(4)
C  XMAX(4), XMIN(4), YMAX(4), YMIN(4), IA(7), ITITLE24, NARZ, TARZ
C
DX=X(J+1)-X(J) $ EOLD=EHL(J)
DOLD=DHL(J)
IF (NALPHA .GT. 1) GO TO 20
DH(J) = DH = DOLD = ZHL(J)/(DX+0.5*DTNH*(U(J+1)-U(J)))
GO TO 25
20 DHL(J)=DH=DOLD+ZHL(J)*((X(J+1)+0.5*DTNH*(U(J+1)-U(J)))*NALPHA-(X(J)+
  1 0.5*DTNH*(U(J)))*NALPHA)
25 IF (NPR(M).EQ.7) GO TO 200
NSC=MAX1(1, 100.*ABS((DHEND-DOLD))/Ealem+DOLD)
NASC=MIN((NSC, 10)
DDH = (DHEND-DOLD)/NSC
SSC=0. $ IF (NSPEC = 0). AND. SDURM .LT. 1.) SSC=SSCALH(J)/NSC
IF (NSC .EQ. 1) GO TO 50
PRINT 1030, NSC, J, N
1030 FORMAT (* SUBCYCLING IN HAFSTEP, NSC=3, FOR J=3, N=3)*
DTNS = DTN
DTNHS = DTNH
DTN = DTNHS/NSC
50 DC 120 NS = 1, NSC
51 DHL(J) = DH = DOLD+DDH
52 HDV=0.5*(1./DOLD-1./DH)
53 RHOHD=ZHL(J)*FACT(RHOL(J)-SHL(J))
54 EH=EOLD+HDV*FACT(2.*RHOHD+DDPD(J)*(U(J+1)-U(J)+DOLD)*DOLD+DPPD(J))
55 IF (NPERP(A) .LT. 2) GO TO 70
56 ZH(6) = (SHL(J)-PHEL(J)-DTN(J))*(2.*HDV*(U(J+1)-U(J)))*(DTNH+DTN)/(X(J+1)-
  1 X(J))/(DOLD+DOLD))
C CYLINDRICAL CASE
60 EZ=1.5*(SLH(J)-PHEL(J))*(2.*HDV*(U(J+1)-U(J)))*(DTNH+DTN)/(X(J+1)-
  1 X(J))/(DOLD+DOLD)
C SPHERICAL CASE
65 EZ=1.5*(SHL(J)-PHEL(J))*(2.*HDV*(U(J+1)-U(J)))*(DTNH+DTN)/(X(J+1)-
  1 X(J))/(DOLD+DOLD)

255
SUBROUTINE HAFSTEP (Concluded)

70  CONTINUE
00      CALL HSTRESS TO COMPUTE STRESS VARIABLES
00      CALL HSTRESS
00      RHL(J)=RH
00      EHL(J)=EH
00      IF (NPR(M).EQ.1.OR. NPR(M).EQ.4) GO TO 90
00      RHL(J)=RH=(RH+DPDE(J)*(EOLD+SSC-EH+FAC*HDV*RHOLD))/(1.-HDV*DPDE(J))
00       1=FAC
00      EHL(J)=EH=(RHOLD+SHL(J)+FAC*(RH-SHL(J)))*HDV+EQSTG(M)*EH+EQSTC(M)/RHO(M)
00      IF (DH.EQ.0.D0LD) GO TO 80
00      DPDDA=EQSTG(M)*EH+EQSTC(M)/RHO(M)
00      IF (DH .EQ. DOLD) GO TO 80
00      DPDD(J)=(-RH-RHOLD-2*DPDE(J))*SSC)/(DH-DOLD)
00        80
00      IF (DSPEC .EQ. 1) GO TO 140
00      DOLD = DH $ EOLD = EH
00      CONTINUE
00      DTN = DTNS $ DTNH = DTNHS
00      CONTINUE
00      RETURN
00      END
SUBROUTINE HDATA

INTEGER H
DIMENSION H(200,3)
READ 1000,A1,J1,I1,K1,A2,J2,I2,K2
PRINT 1000,A1,J1,I1,K1,A2,J2,I2,K2
H(J1,I1)=K1
IF (J2 .EQ. 0) RETURN
H(J2,I2)=K2
RETURN
1000 FORMAT(2(A10,2I5,5X,R5))
END
SUBROUTINE HSTRESS

THIS ROUTINE CONTROLS SWITCHING BETWEEN EQUATIONS OF STATE
COMPUTES R,S,P FOR THE HALFSTEP POINT BETWEEN J AND J+1
INPUT - J, M, DOLD, EOLD, UH, DH, EH.
OUTPUT - RH, SHL, PHL, YHL, H, C.

INTEGER H, POROUS, PRESS, RINTER, SOLID, SPALL
REAL MATL, NEM, NET, NEMH, NETH

MISCELLANEOUS

COMMON AZERO(1), CEF, CKS, DAVG, DELTIM, DISCP(10), DOLD, DRHO, DMAX,
1 DMIN, DTNH, DU, DX, EOLD, F, FAC, FIRST, J, JCYCS, JINIT,
2 JFIN, JREZON, JSTAR, JTS, LSUB(30), N, MAXPR(30), N, NCYCS,
3 NEDIT, NPERN, NR, NRE20, NSCRB(6), NSPALL, NJFIN, JREZONC15, JSMAX,
4 NTEDT, NTEX, NTR(15), POLD, P6(20), R(30), RLAST, SLAST, SMAX, TEDIT(50),
5 TF, TIME, TJ, TRES, TS(20), ULAST, UOLD, UZERO, XLAST, XNOW, XORL
1 , , XJEDIT(20), MS
2 HALFWAVE VALUES
3 COMMON DH, DHLAST, DUH, PH, RH, RHLAST, SH, SHLAST, UH, UHLAST, XH, XHLAST
1 , , NEMH, NETH

CONDITION INDICATORS

COMMON INF, LINTER, MIRROR, NORMAL, POROUS, PRESS, RINTER, SOLID, SPALL

CELL LAYOUT

COMMON DXX(30), JBOUND(30), JMATH(30), NAUTO, MATL(6,2), NAYER, NMTRLS,
1 THK(30)
2 Coordinate Arrays
3 COMMON/COORD/X(200), XO(200), CHL(200), DHL(200), DPD(200), DPDE(200),
1 EHL(200), H(200), NEM(200), NET(200), PHL(200), RHL(200), SDT(200),
2 SHL(200), T(200), U(200), YHL(200), ZHL(200)
3 Named Common
4 REAL MU, MUM
5 COMMON /EQS/ EQSTA(6), EQSTC(6), EQSTD(6), EQSTE(6), EQSTG(6),
1 EQSTH(6), EQSTN(6), EQSTS(6), EQSTV(6), CZO(6), CWG(6), C2(6)
5 COMMON /EQTH/ EMLT(6,8), EMELT(6,8), SPH(6), THERM(6,8)
6 COMMON /RHS/ RH(6), RHOS(6)
7 COMMON /EMLT/ T(6,30), EXMT(6,20), TENS(6,3)
8 COMMON /Y/ YO(6), YADD(6), MU(6), MUM
9 COMMON /Ind/ I(6), I(20), NALPHA, NCOMP(6), NFR(6), NPOD(6),
1 NDS(6), NPER(6), NCOMP(6), NVAR(6)
2 COMMON/RAD/ SSTOR(9), START(9), SDOH, SDO congratulations, NSPEC, SSJ, SJS, IPLOT(4)
2 COMMON/PE/ LMAX, LVTG, LVAR(200), CM(4000)
3 COMMON/ESC/ ESC(6,20)
4 DATA MM/0/

ABORT FOR NEGATIVE DENSITY

IF (DH .GT. 0.) GO TO 25
WRITE (6,4990) N,J,DH,TIME $ LSDIJ(7)=1 $ RETURN

COMPUTE THERMAL STRENGTH REDUCTION AND OLD DEVIATOR STRESS

25 F=1.
1 IF (N .EQ. 0.) GO TO 30
I IF (MM .EQ. 1000*N+N) GO TO 30
I IF (THERM(1) .EQ. 0.) GO TO 27
EMELT(1) =THERM(1) +THERM(1) -THERM(M,6) $ EXP(-TIME+0.5*DTNH)
1 /THERM(M,2)
27 MM=1000*M
1 IF (THERM(3) .EQ. 0.) GO TO 30
EGSTH(1) =THERM(1) +THERM(1) -THERM(M,8) $ EXP(-TIME+0.5*DTNH)
1 /THERM(M,4)
30 IF (EH .LT. EMELT(M)) IE=1
1 IF (H(J,3) .EQ. SR .AND. IE .EQ. 1) H(J,3)=5R .E
I IF (EMELT(M,1) .GT. 1.0) GO TO 34
I IF (THERM(1) .EQ. 0.) GO TO 33
CALL TMELT(0,J,MU,F,G0)
I IF (EMELT(M,1) .EQ. 0.) CALL TMELT(1,MU,F,G0)
I GO TO 34
33 CALL TMELT(4,MU,F,G0,F,G0)
34 IF (E .EQ. 0.) H(J,3) =SR .M
I IF (EXMATH(M,4) .EQ. 0.) MUM=MU(M)+DH-RHOS(M)*EXMATH(M,4)*F
1 T(J)=TENS(M,1)*F
YADD=MADD(M)
1 CJ=32G(M)
CW=CMQ(M)
SUBROUTINE HSTRESS (Continued)

DUH=U(J+1)-U(J)
DUHM=DUH
DT=0.5*(DTN+DTNH)
IF (NALPHA .GT. 1) DUHM=-2.*DX/DH*(DH-DOLD)/(DTN+DTNH)

C
C STRAIN CALCULATIONS AND DEFINITION OF DEVIATOR STRESSES FROM
C FROM PREVIOUS CYCLE. STRAINS AND STRESS DEVIATORS ARE
C POSITIVE IN TENSION.

TXY=0.
ROT=0.
K=1
DROT=0.
EV=-2.*(DH-DOLD)/CDH+DOLD)
SDH=SHL(J)-PHL(J)
SX=-SDH
IF (N .LE. 1) EXEN=EYEN=EZEN=0.
IF (NSCRB(6) .EQ. 0 .OR. MOD(N,NEDIT) .NE. 1) GO TO 45
DXO=XO(J+1)-XO(J)
EXEN=(DX-DXO)/DXO
IF CNALPHA .EQ. 1) GO TO 45
XOS=XO(J+1)+XO(J)
EYEN=(X(J+1)+X(J)-XOS)/XOS
CONTINUE
GO TO (50,60,70)NALPHA

PLANAR GEOMETRY
EX = EV
EY = 0.
EZ = 0.
SY=SDH/2.
SZ = SY
GO TO 80

Cylindrical Geometry
EX=DUH*DT/DX
EY=(EV-EX)/2.
EZ = EY
SY=-SDT(J)
SZ=-(SX+SY)
GO TO 60

Spherical Geometry
EX=DUH*DT/DX
EY=(EV-EX)/2.
EZ = EY
SY=SDH/2.
SZ = SY
EZEN=EYEN
CONTINUE

PRINT 81,N,J,TCX,TOY,TCZ,PHL(J),ECXEN,ECYEN,ECZEN
FORMAT(*
N,J=«2I4,*
TCX,TOY,TCZ,PHL=*1P4E10.3,* ENG. STRAINS
1 ECX,ECY,ECZ=«3E10.3)

** ROUTES FOR COMPOSITE, POROUS, AND FRACTURE MODELS **
100 IF (NCOMPCS+NFRM+NPORE .EQ. 0) GO TO 200
100 IF (NCOMPCS .EQ. 0) GO TO 130

ROUTE FOR COMPOSITE MODEL
-- REBAR --
L=LVAR(J)
CALL REBARC(J,1,5,J,J,M,N,H(J,1),DH,DOLD,SX,SY,SZ,TXY,EH,PHL(J),EX,
1EY,EZ,EXY,F,0.,0.,ESC,COM(L),COM(L+1),COM(L+3),NEM(J),NETC(J),
2 YHL(J),COM(L+2),0)
SDH=-SX
SDT(J)=-SY
GO TO 400

ROUTE FOR POROUS MODEL
130 IF (NPORE .EQ. 0) GO TO 160
IF(NPORE .EQ. 3) GO TO 135
IF (H(J,1) .GE. SOLID) GO TO 200
IF (F .GT. 0.) GO TO 135
H(J,1)=SOLID $ GO TO 200

HSTRESS 47
HSTRESS 48
HSTRESS 49
HSTRESS 50
HSTRESS 51
HSTRESS 52
HSTRESS 53
HSTRESS 54
HSTRESS 55
HSTRESS 56
HSTRESS 57
HSTRESS 58
HSTRESS 59
HSTRESS 60
HSTRESS 61
HSTRESS 62
HSTRESS 63
HSTRESS 64
HSTRESS 65
HSTRESS 66
HSTRESS 67
HSTRESS 68
HSTRESS 69
HSTRESS 70
HSTRESS 71
HSTRESS 72
HSTRESS 73
HSTRESS 74
HSTRESS 75
HSTRESS 76
HSTRESS 77
HSTRESS 78
HSTRESS 79
HSTRESS 80
HSTRESS 81
HSTRESS 82
HSTRESS 83
HSTRESS 84
HSTRESS 85
HSTRESS 86
HSTRESS 87
HSTRESS 88
HSTRESS 89
HSTRESS 90
HSTRESS 91
HSTRESS 92
HSTRESS 93
HSTRESS 94
HSTRESS 95
HSTRESS 96
HSTRESS 97
HSTRESS 98
HSTRESS 99
HSTRESS 100
HSTRESS 101
HSTRESS 102
HSTRESS 103
HSTRESS 104
HSTRESS 105
HSTRESS 106
HSTRESS 107
HSTRESS 108
HSTRESS 109
HSTRESS 110
HSTRESS 111
HSTRESS 112
HSTRESS 113
HSTRESS 114
HSTRESS 115
HSTRESS 116
HSTRESS 117
HSTRESS 118
HSTRESS 119
HSTRESS 120
HSTRESS 121

259
SUBROUTINE HSTRESS (Continued)

135 CONTINUE
HSTRESS 122
NPORM = NPORM (M)
GO TO (140, 145, 150, 155) NPORM
HSTRESS 124
C
HSTRESS 125
C -- PORFREQ --
HSTRESS 126
140 CALL POREQST (1, 5, M, CHL (J)), DH, DOLD, EH, EOLD, F, PHL (J), CZJ, CWJ, H (J), 1),
1 DPDE (M), EQSTC (M), EQSTD (M), EQSTG (M), EQSTS (M), MUM, RHOS (M),
2 YADD, NDS (M), NPR (M), J)
GO TO 310
HSTRESS 127
145 CONTINUE
HSTRESS 128
C
HSTRESS 129
C -- PORHOLT --
HSTRESS 130
CALL PORHOLT (1, 5, M, CHL (J)), DH, DOLD, EH, EOLD, F, PHL (J), H (J), 1),
1 DPDE (J), EQSTC (M), MUM, YADD, RHOS (M), DT)
GO TO 310
HSTRESS 131
150 NPRM = NPR (M) + 1
HSTRESS 132
C
HSTRESS 133
C -- PEST --
HSTRESS 134
L = LVAR (J)
MUM = 1.333 * MUM
CALL PEST (2, 5, NPRM, H (J, 1), J), T (J), DT, M, CHL (J), DH,
1 DOLD, RHOS (M), COM (L), PHL (J), COM (L+1),
2 COM (L+2), EH, EOLD, F, EQSTC (M), EQSTD (M), EQSTG (M),
3 EQSTS (M), MUM, YADD, COM (L+3), COM (L+4), CZJ,
4 CWJ, EQSTH (M), EQSTEC, EQSTG (M),
5 EQSTG (M), DPDE (J), DPDE (J), N)
MUM = 0.75 * MUM
GO TO 300
HSTRESS 135
HSTRESS 136
157 SX = SX - PHL (J)
SY = SY - PHL (J)
SZ = SZ - PHL (J)
HSTRESS 137
C
HSTRESS 138
C -- CAP1 --
HSTRESS 139
C SX, SY, SZ ARE TOTAL STRESSES POSITIVE IN TENSION
HSTRESS 140
159 CALL CAP1 (LSUB (8), IN, M, N, H (J, 1), DH, DOLD, EH, EX, EY, EZ, EXY, F,
1 PHL (J) = (SX + SY + SZ) / 3.
SDH = SX - PHL (J)
SDT (J) = SY - PHL (J)
GO TO 400
HSTRESS 141
HSTRESS 142
HSTRESS 143
HSTRESS 144
HSTRESS 145
HSTRESS 146
HSTRESS 147
HSTRESS 148
HSTRESS 149
HSTRESS 150
HSTRESS 151
HSTRESS 152
HSTRESS 153
HSTRESS 154
HSTRESS 155
HSTRESS 156
HSTRESS 157
HSTRESS 158
HSTRESS 159
HSTRESS 160
HSTRESS 161
HSTRESS 162
HSTRESS 163
HSTRESS 164
HSTRESS 165
HSTRESS 166
HSTRESS 167
HSTRESS 168
HSTRESS 169
HSTRESS 170
HSTRESS 171
HSTRESS 172
HSTRESS 173
HSTRESS 174
HSTRESS 175
HSTRESS 176
HSTRESS 177
HSTRESS 178
HSTRESS 179
HSTRESS 180
HSTRESS 181
HSTRESS 182
HSTRESS 183
HSTRESS 184
HSTRESS 185
HSTRESS 186
HSTRESS 187
HSTRESS 188
HSTRESS 189
HSTRESS 190
HSTRESS 191
HSTRESS 192
HSTRESS 193
HSTRESS 194
HSTRESS 195
HSTRESS 196

260
SUBROUTINE HSTRESS (Continued)

GO TO 400

-- SHEAR1 --
160 IF(H(J,3)-2)177,181,183
181 IF(NFR(M) .EQ. 3) GO TO 183
STENS=AMAX1(SX,SY,SZ)
IF(-STENS .LT. TSR(M,5)*TSR(M,9) .AND. -STENS+PHL(J) .LT. TSRCM) HSTRESS
1 GO TO 177
183 LS=2
IF(NFR(M) .EQ. 0) LS=3)
STENS=AMAX1(SX,SY,SZ)
IF (N .EQ. 0) GO TO 200
Q0 TO 183
181 IF(NFR(M) .EQ. 3)Q0 TO 183
STENS=AMAX1(SX,SY,SZ)
LS=LSUB(12)
IF (LS .NE. 0 .AND. MOD(N,NEDIT) .EQ. 0) LS=2
L=LVAR(J)
CALL SHEAR2CLS(J)
HSTRESS
183 SDH=-SX
HSTRESS
212
GO TO 400

C

-- BFRACT --
195 IF(H(J,3) .GT. 2)1GO TO 197
STENS=AMAX1(SX,SY,SZ)
IF (N .EQ. 0) GO TO 200
Q0 TO 183
1
183
LSUB(12)=1
GO TO 400

C

**** ROUTES FOR PRESSURE CALCULATION ****

200 NPRM = NPR(M)+1
GO TO (270,220,230,240,250,255,260,270) NPRM

C

EQUATION OF STATE FOR EXPLOSION PRODUCTS
220 IF (NEM(J) .GE. 0.999999) GO TO 270
QH=0.  $ IF(DHM+LT. .LT. 1.) QH=(CZJ*DUHM-CWJ*CHL(J))*DUHM
L=LVAR(J)
CALL EXPL0DE(3,5 M,EHL,DHL,D0LD,J PHL,J SHL,NEM,J X,J QH,TIME,DT)
EH=EHL(J)
GO TO 305

C

SIMPLE, EXTENDED EQUATION OF STATE
230 CALL ESA(1,M,CHL(J),EH,PHL(J),DPDD(J),DPDE(J))
GO TO 300

C

PHILCO-FORD EQUATION OF STATE
240 CALL EQSTPFCI,J 5 M,CHL(J),EH,PHL(J))
GO TO 300

C

VARIABLE MODULI EQN. OF STATE
(IMPLEMENTED FOR PLANAR AND SPHERICAL CASES ONLY)
250 EPS=EMU+ALOG(CX(J)+U(J)*DTNH/2.)/COM(L))
NET(J)=EPS
CALL HYPOCl,IN,M,CHL(J)^H.EMU.COMCL+l),EPS
GO TO 400

C

3-PHASE EQUATION OF STATE OF ROYCE
260 CALL GRAYd, IN,M,MUM,EMELT(M,J,J,J,DH,EH,PHL(J),DPDD(J))
GO TO 305

C

MIE-GRUNEISEN AND PUFF EXPANSION EQUATIONS OF STATE
270 CALL EQST(EH,DH,PHL(J),M,CHL(J),DPDD(J),DPDE(J))

261
**SUBROUTINE HSTRESS (Continued)**

300 IF (MUM .GT. 10 .AND. YHL(J)*F .GT. 0. AND. NPR(M,1) .NE. 1) GO TO 310

310 IF (SDH .EQ. 0.) GO TO 312

320 SDH=SDH-2.0*MUM*(EX-EV/3.)

330 CALL BANDRLX(H(J,3),SDH,YHL(J),DRHO,COEF,N,J,M,NEM(J),NET(J),DT,

340 CALL BANDLX(H(J,3),SDH,YHL(J),DRHO,COEF,N,J,M,NEM(J),NET(J),DT,

350 MUM=(MUM+TSR(M,19)*(DH/RHO(M)-1.))

360 CALL STRESS(LSUB(13),0,H(J,3),M,J,N,DH,DOLD,RHOS(M),SDH,MUM,F,DT,

370 SZ=-SZ

380 IF (NALPHA .EQ. 2) GO TO 323

390 IF (H(J,1) .EQ. SOLID .AND. (DH /RHOS(M)-1.) .GT. 0.) SHL(J)

400 IF (H(J,1) .EQ. SOLID .AND. (DH /RHOS(M)-1.) .GT. 0.) SHL(J) =

410 CONTINUE
SUBROUTINE HSTRESS (Concluded)

410 RH=SHL(J) $ CEF=CHL(J)
     IF (DUHM .GE. -1.) GO TO 450
     CF=CWJ-CZJ*DUHM/CHL(J)
     IF (CF .GT. 1.0) CEF=CHL(J)*((2.*CF+0.5/CF)-DUH/2).
     GO TO 470
     450 CF=C2(M)
     470 RH=SHL(J)-CF*CHL(J)*DUHM*DH
     IF (RH.LT.0. . AND. F.L.E.0. .AND. NSEPRAT.EQ.0) RH=SHL(J)=PHL(J)=0.

C SPALL PROVISIONS
     IF (NFR(M) .GT. 0) GO TO 550
     SHLSV=SHL(J)
     PHLSV=PHL(J)
     IF (RH .LT. T(J)) GO TO 515
     IF (NALPHA .EQ. 2) GO TO 505
     IF (PHL(J) -SDH/2. .LT. T(J)) GO TO 520
     GO TO 550
     505 IF (PHL(J) +SDT(J) .LT. T(J)) GO TO 525
     IF (2.*PHL(J)-SDT(J)-SHL(J).LT.TCJ)) G0 TO 530
     GO TO 550
     515 SHL(J)=PHL(J)=RH=0.
     GO TO 535

C SPALL BY LATERAL STRESS, NALPHA=1,3
     RF=(3.*PHL(J)-SHL(J))/2./(EQSTC(M)+1.333*MUM)
     DP=EQSTC(M)*RF
     DS=MUM*RF
     Q=RH-SHL(J)
     SHL(J)=SHL(J)-DP+DS/2.
     RH=SHL(J)+Q
     PHL(J)=PHL(J)-DP
     SHL(J)=PHL(J)+Q
     GO TO 535

C SPALL BY THETA STRESS, NALPHA=2
     RF=(PHL(J)+SDT(J))/(EQSTC(M)+1.333*MUM)
     DP=EQSTC(M)*RF
     DS=MUM*RF
     PHL(J)=PHL(J)-DP
     Q=RH-SHL(J)
     SHL(J)=SHL(J)-DP+DS/2.
     RH=SHL(J)+Q
     PHL(J)=PHL(J)+Q
     SDT(J)=SDT(J)+DS/2.
     GO TO 535

C SPALL BY Z STRESS, NALPHA=2
     RF=(2.*PHL(J)-SDT(J)-SHL(J))/(EQSTC(M)+1.333*MUM)
     DP=EQSTC(M)*RF
     DS=MUM*RF
     PHL(J)=PHL(J)-DP
     Q=RH-SHL(J)
     SHL(J)=SHL(J)-DP+DS/2.
     SDT(J)=SDT(J)+DS/2.
     GO TO 535

PRINT 4992,J,N,SHL(J),PHL(J),RH,Q,SDT(J),DP,DS,SHLSV,PHLSV
4992 FORMAT(* ---J=x13, * N=x13, * SHL,PHL,RH,Q=x1P4E11.3, * SDT,DP,DS=x1P3E11.3)
     IF (H(J,2) .GT. 0 . AND. H(J,2) .LT. 77B) GO TO 550
     IF (H(J,2) .NE. NORMAL) GO TO 550
     H(J,2)=NSPALL=NSPALL+1
     CONTINUE
     550 CONTINUE
     RETURN

4990 FORMAT(2OH STOP IN HSTRESS, N=14,4H, J=14,4H, D=1PE10.3,  
1 7H, TIME=1PE10.3)
     END

HSTRESS 349
HSTRESS 350
HSTRESS 351
HSTRESS 352
HSTRESS 353
HSTRESS 354
HSTRESS 355
HSTRESS 356
HSTRESS 357
HSTRESS 358
HSTRESS 359
HSTRESS 360
HSTRESS 361
HSTRESS 362
HSTRESS 363
HSTRESS 364
HSTRESS 365
HSTRESS 366
HSTRESS 367
HSTRESS 368
HSTRESS 369
HSTRESS 370
HSTRESS 371
HSTRESS 372
HSTRESS 373
HSTRESS 374
HSTRESS 375
HSTRESS 376
HSTRESS 377
HSTRESS 378
HSTRESS 379
HSTRESS 380
HSTRESS 381
HSTRESS 382
HSTRESS 383
HSTRESS 384
HSTRESS 385
HSTRESS 386
HSTRESS 387
HSTRESS 388
HSTRESS 389
HSTRESS 390
HSTRESS 391
HSTRESS 392
HSTRESS 393
HSTRESS 394
HSTRESS 395
HSTRESS 396
HSTRESS 397
HSTRESS 398
HSTRESS 399
HSTRESS 400
HSTRESS 401
HSTRESS 402
HSTRESS 403
HSTRESS 404
HSTRESS 405
HSTRESS 406
HSTRESS 407
HSTRESS 408
HSTRESS 409
HSTRESS 410

263
SUBROUTINE HYDRO

SUBROUTINE CONTROLS THE MAIN CALCULATION CYCLE
* CONTAINS 6 PATHS -
  1. NORMAL - COORDINATES WITHIN MATERIAL
  2. INTERFACE - INTERFACE BETWEEN MATERIALS
  3. INTERFACE SPLA - SEPARATED INTERFACE BETWEEN MATERIALS
  4. MIRROR - FIRST COORDINATE FOR A SYMMETRIC IMPACT
  5. PRESSURE - PRESSURE HISTORY APPLIED AT FRONT (J=1)
  6. LEFT INTERFACE - DUMMY PATH
* CALLS HAFSTEP FOR HALFWAVE CALCULATIONS AT EACH COORDINATE
* CHECKS FOR SPLA AND RECOMBINATION
* COMPUTES MINIMUM PERMITTED TIME STEP FOR NEXT CYCLE

INPUT - DTNH, DTN, FIRST, NCYCS.

NAMED COMMON
REAL MU, MUM
COMMON /EQL1/ EQTA(6,6), EQSTC(6), EQSTD(6), EQSTE(6), EQSTG(6),
1 EQSTH(6), EQST(6), EQSTV(6), EQTV(6), CQZ(6), CQW(6), CQX(6)
COMMON /MELL/ EMELT(6,6), SPM(6,6), THERM(6,6)
COMMON /RHO/ RHG(6,6), RHOS(6)
COMMON /TISM/ TSR(6,30), EXMA(6,30), TENS(6,3)
COMMON /Y/ YO(6), YADD(6), YADD, YADD
COMMON /IND1/ EOS(6), INDX(20), NALPHA, NCM(6), NFR(6), NFRP(6),
1 NDS(6), NFRP(6), NCON(6), NVAR(6)
COMMON /RAD/ SSTP(9), STARTN, SDUN, SSTOPM, NSPEC, SSJ, JSS, IPILOT(4)

SUBROUTINE HYDRO

SUBROUTINE HYDRO

INTEGER H, POROUS, PRESS, RINTER, SOLID, SPALL
REAL MATL, NEM, NET, NEMH, NETH

MISCELLANEOUS
COMMON AZERO(1), CEF, CKS, DAVG, DELTIM, DISCPT(10), DOLD, DRHO, DTMAX,
1 DTMN, DTN, DTNH, DU, DX, EOLD, F, FAC, FIRST, J, JYCS, JINT,
2 JFIN, JREZON(15), JMAX, JSTAR, JS, LSUB(30), M, MAXPR(30), N, NCYCS,
3 NEDIT, NPERR, NREZON, NSCRB(6), NSEPRAT, NSPALL, NTEDT,
4 NTEX, NTRN(15), POLD, P6(30), R(30), RSLAST, SLAST, SMAX, TEDIT(50),
5 TF, TIME, TJ, TREZON, TS, T6(20), ULAST, UOLD, UZERO, XLAST, XNOW, XOLD
1 XJEDIT(20), MS

INTEGER H, POROUS, PRESS, RINTER, SOLID, SPALL
REAL MATL, NEM, NET, NEMH, NETH

CONDITION INDICATORS
COMMON INF, INTEN, MIRROR, NORMAL, POROUS, PRESP, RINTER, RINTER, SOLID, SPALL
CELL LAYOUT
COMMON DXX(30), JBND(30), JMAT(30), NAUTO, MATL(6,2), NLayer, NMTLRS,
1 THK(30)

COORDINATE ARRAYS
COMMON /COORD/ X(200), YO(200), CHL(200), DHL(200), DPDD(200), DPDE(200),
1 EHL(200), H(200,3), NEM(200), NET(200), PHL(200), RHL(200), SDT(200),
2 SHL(200), T(200), U(200), YHL(200), ZHL(200)

DO IF (N .EQ. 1) ISPALL = 0
1 DT = DTMIN = 1

**** ROUTINE TO RESET DTNH FOR SPALL CLOSURE ******
IF (NLAYER .LE. 1 OR. ISPALL .EQ. 0) GO TO 62
NLMO = NLAYER - 1
DO 80 LLL = 1, NLMO
1 JB = JBND(LLL)
IF (JB GT JSTAR) GO TO 82
1 IF (H(JB+1,2) NE. SPALL) GO TO 80
1 IF (U(JB), EQ. U(JB+1)) GO TO 80
1 DTSP = (X(JB+1)-X(JB))/(U(JB)-U(JB+1))
1 IF (DTSP .GT. DTNH) DTSP = DTSP .LT. 0.001)
1 DTNH = AMIN1(DTNH,AMAX1(DTSP,0.2*DTNAT)) - 0.001*DTNAT
1 DTMN = DTMN .LE. 0.2*DTNAT
1 NCYCS = 1
80 CONTINUE
82 CONTINUE

C CUTER HYDRO LOOP
DO 1000 NN = 1, NCYCS
1 TIME = TIME + DTNH

264
SUBROUTINE HYDRO (Continued)

I SPALL = 0
FAC = FLOAT(MINO(N-1, 20))/20.
LL=0
DO 900 J=JMNT, JFIN
C CHECK FOR THE APPROPRIATE PATH
10 XOLD=X(J) $ UOLD=U(J)
IF (H(J,2) .EQ. NORMAL) GO TO 100
IF (H(J,2) .EQ. LINTER) GO TO 900
IF (H(J,2) .EQ. RINTER) GO TO 200
IF (H(J,2) .EQ. SPALL) GO TO 300
IF (H(J,2) .EQ. MIRROR) GO TO 500
IF (H(J,2) .EQ. PRESS) GO TO 600
C
100 IF (NSPEC.GT.0 .OR. ABS(U(J)-U(J+1)).GT.0.001 .OR. EHL(J).GT.1. .OR. NPR(J).EQ.1) GO TO 102

101 X(J)=X(J)+DTNH*U(J) $ GO TO 102
CALL HAFSTEP
C VELOCITY CALCULATION
120 IF (NALPHA-2) 125,130,135
C PLANAR CASE
125 U(J)=U(J)+4.5*(X(J+1)+X(J)+DTNH*U(J))

GO TO 140
C CYLINDRICAL CASE
130 XBAR1=SQRT((X(J-1)**2+X(J)**2)/2.)
XBAR2=SQRT((X(J)**2+X(J+1)**2)/2.)
U(J)=U(J)+4.*DTNH*(RHLAST*XBAR1-RH*XBAR2+(PHL(J-1)+SDT(J-1))*(X(J)-XBAR1)+(PHL(J)+SDT(J))*(XBAR2-X(J)))/2

GO TO 140
C SPHERICAL CASE
135 XBAR1 = ((X(J-1)**3+X(J)**3)/2.)**(1./3.)
XBAR2 = ((X(J)**3+X(J+1)**3)/2.)**(1./3.)
U(J) = (U(J)+ZHL(J)/3.+U(J)*ZHL(J)/3.-2.*DTNH**2*(RH*XBAR2**2-RHLAST*XBAR1**2+0.5*(SHL(J-1)-3.*PHL(J-1))**2))/(ZHL(J-1)+ZHL(J))*3.

GO TO 140
C COORDINATE CALCULATION
140 X(J)=X(J)+0.5* DTNH*(U(J)+UOLD)
DT=(X(J+1)+U(J+1)**2-X(J))/CEF
GO TO 800
C INTERFACE
C LEFT VALUES ARE IN (J-1) CELLS AND RIGHT VALUES ARE IN (J) CELLS
200 IF (X(J-1) .LT. X(J)) GO TO 300
MLAST=M $ LL=LL+1 $ M=MAT(LL)
CALL HAFSTEP
C CHECK STRESS AND SET INDICATORS FOR SPALL
J1=J-1
IF (R(MLAST).GT. T(J1)) GO TO 205
H(J,2)=SPALL $ R(MLAST)=T(J1)=0. $ SPALL=1
IF (MAXPR(J).LE. 0) GO TO 205
PRINT 5230, N,NN,LL,TIME
205 J1=J-1 $ J2=J-2
UOLD=U(J1)
IF (NAPLA=2) 210,212,215
C PLANAR CASE
210 U(J)=(U(J1)+ZHL(J)-2.*DTNH*(RH-RHLAST))=/ 

1 (ZHL(J)+ZHL(J-1))
GO TO 218
C CYLINDRICAL CASE
212 XBAR1=SQRT((X(J-1)**2+X(J)**2)/2.)
XBAR2=SQRT((X(J)**2+X(J+1)**2)/2.)
U(J)=(U(J1)+ZHL(J)+U(J)*ZHL(J)+4.*DTNH*RHLAST*XBAR1-RH*XBAR2+ 

1 (PHL(J)+SDT(J)*(X(J)-XBAR1)+(PHL(J)+SDT(J))*(XBAR2-X(J)))/ 

2 (ZHL(J)+ZHL(J-1))
GO TO 218
C SPHERICAL CASE
215 XBAR1 = ((X(J-1)**3+X(J)**3)/2.)*((1./3.)
XBAR2 = ((X(J)**3+X(J+1)**3)/2.)*((1./3.)
U(J) = (U(J1)+ZHL(J)/3.+U(J)*ZHL(J)/3.-2.*DTNH 

1 *(RH*XBAR2**2-RHLAST*XBAR1**2+0.5*(SHL(J)-3.*PHL(J)) /** 

265
SUBROUTINE HYDRO (Continued)

2 \((XBAR1*X(J))+(X(J)-XBAR1))+0.5*(SHL(J)-3.*PHL(J))\times(XBAR2+X(J))^{HYDRO}
3 \((XBAR2-X(J)))/(ZHL(J)+ZHL(J2))^{HYDRO}

218 U(J1)=U(J)
X(J)=X(J1)+X(J)+5.*DTNH\times(U(J)+UOLD)
R(LL)=(RH\times ZHL(J2)+RHLAST\times ZHL(J)+0.5\times (UOLD-UOLD)\times ZHL(J2))^{HYDRO}
1 \((ZHL(J)/DTNH)/(ZHL(J)+ZHL(J2)))^{HYDRO}
DT=(X(J1)+U(J1))/DTNH-X(J))/CFE

22D CONTINUE
GO TO 8DD

C
C INTEFACE SPALL

C

30D IF (J_EQ. JINIT) GO TO 330
MLAST=M

C LEFT SIDE

J1=J-1
XL0LD=X(J-1) $ U=U(J-1)
GO TO 320

C PLANAR CASE

305 U(J1)=U(L0LD+2.*DTNH=RHLAST/ZHL(J-2)
GO TO 320

C CYLINDRICAL CASE

310 XBAR1=SQRT((X(J-2)**2+X(J-1)**2)/2.)
U(J1)=U(J1)+4.**DTNH*(RHLAST*XBAR1*(PHL(J)-2)+SDT(J-2))
1 \((X(J-1)-XBAR1))/ZHL(J-2))^{HYDRO}

GO TO 320

C SPHERICAL CASE

315 XBAR1 = ((X(J-2)**2+X(J-1)**2)/2.)**(1./3.,)
U(J1) = U(L0LD+2.*DTNH=RHLAST+XBAR1**2-D.5*(SHL(J-2)-3.*PHL(J-2)
1 \times(XBAR1+X(J-1))/X(J-1)-XBAR1))/ZHL(J-2))^{HYDRO}

GO TO 320

C RIGHT SIDE

330 LL=LL+1 $ M=MAT(LL)
R(LL)=X
GO TO 332

IF (NSPEC.GT.0 .OR. ABS(U(J)-U(J+1)) \times 0.001 .OR. EHL(J) .GT. 1.)
GO TO 332

1 \times OR. ABS(RHLAST) .GT. 1. .OR. NPR(M).EQ.1) GO TO 332

331 U(J)=U(J) $ RH=RHL(J) $ X=X*0.5\times(X(X+1)+X(J)+DTNH=U(J))
X(J)=X(J)+DTNH=U(J) $ DT=1.
IF (N=2 .AND. J .EQ. JINIT) X(J)=MAX1(XOLD, D.)
IF (N=2 .AND. J .EQ. JINIT) X(J)=MAX1(XOLD, D.)

GO TO 334

D=X+DHL(J) $ EH=EHL(J)
IF (J_EQ. JINIT) 80D, 335

332 CALL HAFSTEP
IF (RHL(J) .GT. T(J)) GO TO 334

334 U=U(J) $ X=X+X
IF (N=2) 3341, 3342, 3343

3341 U(J)=U(L0LD+2.*DTNH=RHL(J)
GO TO 3343

334 C CYLINDRICAL CASE

3342 XBAR2=SQRT((X(J)**2+X(J+1)**2)/2.)
U(J)=U(L0LD+4.*DTNH=RXBAR2+RHL(J)+SDT(J))\times(XBAR2-X(J)))/ZHL(J)
GO TO 3344

334 C SPHERICAL CASE

3343 XBAR2 = ((X(J)**2+X(J+1)**2)/2.)**(1./3.,)
U(J) = U(L0LD+2.*DTNH=RXBAR2+X(J+1)**2+0.5\times(SHL(J)-3.\times PHL(J))\times
1 \times(XBAR2+X(J))/X(J)-XBAR2)/ZHL(J)\times3.

3344 X(J)=X+0.5\times(DTNH=U(J)+UOLD)
IF (N=2 .AND. J .EQ. JINIT) X(J)=MAX1(XOLD, 0.)
IF (N=2 .AND. J .EQ. JINIT .AND. X(J) .EQ. J .EQ. D.) U(J)=0.
DT=(X(J+1)+U(J+1))/DTNH-X(J))/CEF

C CHECK FOR RECOMBINATION

IF (J_EQ. JINIT) GO TO 8DD

335 IF (X(J) .LE. X(J-1)) GO TO 365
ISPELL=ISPALL+1 $ GO TO 8DD

C RESET ARRAY VARIABLES AND GO TO INTERFACE ROUTE

365 H(J,2)=RINTER \times X(J)**XOLD $ X(J-1)**XOLD $ U(J)=UOLD

PRINT 1365, N, J, TIME
U(J-1)=UOLD
DT=D.1=AMIN1(DT, DTP)
IF (DT .LT. 0.) DT=1.
IF (DT .GT. DMIN) GO TO 205
SUBROUTINE HYDRO (Continued)

DTMIN=DT $ JTS=J $ GO TO 205

C********** MIRROR AT FRONT SURFACE
C
500  LL=LL+1 $ M=JMAT(4LL)
IF (J .GE. JFIN-1) GO TO 600
CALL HAFSTEP
R(LL)=RHL(J)
X(J)=X(J)+DTNH*U(J)
DT=(X(J+1)+DTNH*U(J+1)-X(J))/CEF
IF (R(LL) .GT. T(JFIN-1)) GO TO 600
H(J,2)=SPALL
R(LL)=T(JFIN-1)=0. $ GO TO 600

C********** PRESSURE BOUNDARY AT FRONT SURFACE
C
600  LL=LL+1
IF (J .EQ. JFIN) GO TO 650
M=JMAT(4LL)
IF (T6(1) .EQ. 0.) GO TO 602
R(LL)=P6(1)*EXP((TIME-DTNH)/T6(1))
CALL HAFSTEP
RHAF=P6(1)*EXP((TIME-0.5*DTNH)/T6(1))
GO TO 603

602  R(LL)=SIGMAT(1,TIME-0.5*DTNH)
CALL HAFSTEP
RHAF=SIGMAT(1,TIME-0.5*DTNH)
CONTINUE
IF (NALPHA-2) 605,610,615

PLANAR CASE
U(J)=UOLD-2.-(RH-RHAF)/ZHL(J)*DTNH
GO TO 620

CYLINDRICAL CASE
XBAR2=SGRT((X(J)*2+X(J+1)*2)/2.)
U(J)=U(J)+4.*DTNH*(RHAF*X(J)-RH*RHL(J+1))+T(JFIN-1))*(XBAR2-X(J))
1./ZHL(J)
GO TO 620

SPHERICAL CASE
XBAR2=((X(J)*2+X(J+1)*2)/2.)**(1./3.)
U(J)=U(J)+2.*DTNH*(RHAF*X(J)**3-RH**XBAR2**3+0.5*(T(JFIN-1)-3.*PHL(J))**3)
1./ZHL(J)
GO TO 620

C********** PRESSURE BOUNDARY AT OUTER SURFACE
C
650  IF (T6(2) .EQ. 0.) GO TO 652
R(LL)=P6(2)*EXP((TIME-DTNH)/T6(2))
RHAF=P6(2)*EXP((TIME-0.5*DTNH)/T6(2))
GO TO 654

C CYLINDRICAL CASE
U(J)=UOLD=2.-(RH-RHAF)/ZHL(J)*DTNH
GO TO 675

C SPHERICAL CASE
XBAR1=SGRT((X(J-2)*2+X(J-1)*2)/2.)
U(J)=U(J)+4.*DTNH*(RHAF*X(J-2)-RH*RHL(J-2))+T(JFIN-1))*(XBAR1-X(J))
1./ZHL(J)
GO TO 675

C INFINITE BOUNDARY AT FRONT SURFACE.
C
GO TO 800
SUBROUTINE HYDRO (Concluded)

700 IF (J.EQ.JFIN) GO TO 720
LL=LL+1 $ M=JM(N)(LL)
IF (ABS(U(J)-U(J+1)) .LT. .001 .AND. EHL(J) .LT. 1. .AND.
1 NPR(M) .NE. 1) GO TO 101
DS=SQRTRHOS(M)*EQSTC(M)+1.333*MU(M)))*U(JFIN+1)-U(J)+1)
DP=EQSTC(M)/(EQSTC(M)+1.333*MU(M)))*DS
SDH=SHL(JFIN+1)-PHL(JFIN+1)+DS+DP
PHL(JFIN+1)+1)=PHL(JFIN+1)+DP
U(JFIN+1)+1)=UJLD
IF (ABS(SDH) .GT. .001 .AND. EHL(J) .LT. 1) GO TO 101
DS=SQRT(RHOS(M)*(EQSTC(M)+1.333*MU(M)))*U(J-1)-U(J)
DP=EQSTC(M)/(EQSTC(M)+1.333*MU(M)))*DS
SDH=SHL(J-1)-PHL(J-1)+DS+DP
PHL(J-1)=PHL(J-1)+DP
U(J-1)+1)=UJLD
IF (ABS(SDH) .GT. .001 .AND. EHL(J) .LT. 1) GO TO 101
SMAX CALCULATION
IF (SHL(J) .GT. SMAX) 820,822
SMAX=SHL(J) $ JSMAX=J
TIME STEP CALCULATION
IF (DT .LT. 0.1) DT=1.
IF (DT .LT. DTMIN) GO TO 826
DTMIN=DT $ JTS=J
DT=DT $ JTS=J
JSTAR CALCULATION
IF (ABS(U(J)) .LT. 1.E-3 .AND. EHL(J) .LT. 1.) 851,900
IF (J .LT. JSTAR) 852,900
JSTAR=J-1
GO TO 990
END OF HYDRO INNER LOOP
900 CONTINUE
JSTAR=JFIN-1
DTN=DTNH
JTS=JTS+1000*ISPALL
END OF HYDRO OUTER LOOP
1000 CONTINUE
1002 RETURN
1365 FORMAT(* RECOMBINATION AT CYCLE *14,*, J=*14,*, TIME=*1PE10.3)
5115 FORMAT(* SPALL AT N, NN=*214,*, FOR J=*14,*, NSPALL=*14,*, TIME=*1PE10.3)
END

C INFINITE BOUNDARY AT REAR SURFACE.

C END OF HYDRO PATHS
C CYCLE RESET
XLAST=XJLD $ ULAST=UJLD $ RLAST=RJ $ DLAST=DJ
EHLAST=EH $ SHLAST=SH
SMAX CALCULATION
IF (SHL(J) .GT. SMAX) 820,822
SMAX=SHL(J) $ JSMAX=J
TIME STEP CALCULATION
IF (DT .LT. 0.1) DT=1.
IF (DT .LT. DTMIN) GO TO 826
DTMIN=DT $ JTS=J
DT=DT $ JTS=J
JSTAR CALCULATION
IF (ABS(U(J)) .LT. 1.E-3 .AND. EHL(J) .LT. 1.) 851,900
IF (J .LT. JSTAR) 852,900
JSTAR=J-1
GO TO 990
END OF HYDRO INNER LOOP
900 CONTINUE
JSTAR=JFIN-1
DTN=DTNH
JTS=JTS+1000*ISPALL
END OF HYDRO OUTER LOOP
1000 CONTINUE
1002 RETURN
1365 FORMAT(* RECOMBINATION AT CYCLE *14,*, J=*14,*, TIME=*1PE10.3)
5115 FORMAT(* SPALL AT N, NN=*214,*, FOR J=*14,*, NSPALL=*14,*, TIME=*1PE10.3)
END

SUBROUTINE HYDRO
SUBROUTINE HYPO

SUBROUTINE HYPO(INDE, IN, M, CJ, DH, EMU, EMUD, EPS, EPSO, J, P, SDH)

THIRD VERSION OF VARIABLE MODULUS MODEL INCLUDING AN INTEGRAL DEFINITION OF LOADING SURFACES FOR P AND SDH AND DIFFERENTIAL RELATIONS ONLY FOR UNLOADING.

ROUTINE IS WRITTEN FOR 1-DIMENSIONAL PLANAR AND SPHERICAL FLOW. FOR 1-D CYLINDRICAL FLOW, SDH IS INTERPRETED AS 2/3 EFFECTIVE STRESS AND RADIAL AND TANGENTIAL DEVIATOR STRESSES ARE COMPUTED FROM SDH IN HSTRESS.

SUBROUTINE COMPUTES PRESSURE P AND DEVIATOR STRESS SDH AND SOUND SPEED CJ.

SUBROUTINE IS CALLED TWICE

INDE=0 CALLED FROM GENRAT
READ MATERIAL PROPERTY DATA AND INITIALIZE VARIABLES
INDE=1 CALLED FROM HSTRESS
COMPUTE PRESSURE, DEVIATOR STRESS, AND SOUND SPEED.

REAL K0, K1, K2, K0UN, K1UN, K2UN, KY
DIMENSION GO(6), Gl(6), G2(6), GU(6), KY(6), EMUY(6)

IF(INDE.GT.0)GO TO 200

**READ AND INITIALIZE**

READ(1N,1000) A1, GO(M), Gl(M), G2(M), GU(M), KY(M), EMUY(M)
WRITE(6,1000) A1, GO(M), Gl(M), G2(M), GU(M), KY(M), EMUY(M)
READ(1N,1000) A1, K0(M), K1(M), K2(M), K0UN(M), K1UN(M), K2UN(M)
WRITE(6,1000) A1, K0(M), K1(M), K2(M), K0UN(M), K1UN(M), K2UN(M)
G0(M) = 1.333*GO(M)
G1(M) = 1.333*Gl(M)
G2(M) = 0.667*G2(M)
GU(M) = 1.333*GU(M)
K2(M) = 0.5*K2(M)
K1UN(M) = 0.5*K1UN(M)
K2UN(M) = 0.5*K2UN(M)
RETURN

**COMPUTATION OF STRESS**

200 DMU=EMU-EMUO
DEPS=EPS-EPSO
DEVIATOR STRESS
SDH=AMIN1(SDH+GU(M)*DEPS, EPS*(G0(M)+G1(M)*EMU+G2(M)*EPS))
PRESSURE
IF(EMU.GT.EMUY(M))GO TO 220
BULK=K0UN(M)+K1UN(M)*EMUO+K2UN(M)*(EPS+EPSO)
P=AMIN1(P+DMU*BULK+KY(M)*EMUY(M)+EMU*K0(M)+K1(M)*EMU+K2(M)*EPS))
GO TO 300
220 P=AMIN1(P+DMU*KY(M)+EMU(M)+EMU)
BULK=KY(M)
300 EMU0=EMU
EPS0=EPS
RETURN

1000 FDRMAT(A10,1P7E10.3)
END
SUBROUTINE PEST

SUBROUTINE PESTLS, IN, NPNRM, H, J, T, DT, M, C, D, DDLOE, RHOS, RHQ, P, PST, PEST 2
1 AST1, E, E0L, F, E0ST, E0ST, E0STM, MUM, YADD, PV, V, NT, CZJ, PEST 3
2 CJ, EGSTHM, EGSTEM, EGSTM, EGSTM, MUM, YADD, PV, V, NT, CZJ, PEST 4
C PEST 5
C PEST 2, VERSION OF DEC 1976 PEST 6
C WRITTEN AT STANFORD RESEARCH INSTITUTE BY L. SEAMAN AND R.E. TOLKHEIM PEST 7
C CODE PROVIDES EQUATIONS OF STATE FOR POROUS AND SOLID MATERIALS PEST 8
C UNDER COMPRESSION(C), TENSION(T) AND RECOMPRESSION(R) BY RATE- PEST 9
C INDEPENDENT AND RATE-DEPENDENT MODELS. INITIALIZATION FOR ALL MODELS PEST 10
C IS INCLUDED. PEST 11
C C INDICATORS OF MODELS TO BE CHOSEN FOR STATIC(S) AND DYNAMIC(D) PEST 12
C CONDITIONS FOLLOW:
C KCS OR KRS: R LINEAR-MODEL OF COMPRESSION
C 1 PCREGST PEST 15
C 2 PORHOLT
C 3 CARROLL-HOLT
C 4 HERRMANN P-ALPHA
C 5 HENDRON
C C KTS: RATE-INDEPENDENT TENSION
C 1 VARIABLE STRENGTH
C 2 FRACTURE MECHANICS
C 3 CARROLL-HOLT
C C KCD OR KRD: COMPRESSION WITH RATE EFFECTS
C 1 NO RATE DEPENDENCE
C 1 LINEAR VISCOUS VOID COMPRESSION
C 3 PORHOLT
C 4 BUTCHER P-ALPHA-TAU
C C KTD: TENSION WITH RATE EFFECTS
C 1 NO RATE DEPENDENCE
C 1 N.A.G. DUCTILE FRACTURE
C 2 BRIE FRACTURE AND FRAGMENTATION
C C INDICATORS(X) ARE READ IN THREE-DIGIT PAIRS FOR S AND D CONDITIONS:
C KCS, KTS, KRS = OXO OXO KCD, KTD, KRD = OXO OXO
C C INDICATORS H AND IH
C S SOLID
C P POROUS-PRESSURE
C T POROUS-TENSION
C Q POROUS-RECOMPRESSION
C C 2 FRAGMENTATION
C R RECOMPRESSION AFTER FRAGMENTATION
C C INTEGER H, OUT
REAL MUM, MUP, KIC
DIMENSION KCS(4), KCD(4), KTS(4), KTD(4), KRS(4), KRD(4)
DIMENSION NPNM(6), NREG(4)
DIMENSION TPH(4, 3), DADP(4, 3), K1C(4), TEMP(8)
DIMENSION AK(4), MUP(4), YADD(4, 3), TER(4, 3)
DIMENSION RHOP(4, 3), COSQ(4, 5, 3), CI(4, 5, 3)
DIMENSION PORA(4, 5, 3), PORS(4, 5, 3), POC(4, 5, 3)
DIMENSION EPS(4, 3), DEL(4, 3), ALE(4, 3), APC(4, 3)
DATA SMF/1.88/, EP/1.E-6/, IDD/1H/, OUT/6/, JQ1/7H-PEST/7/, JQ2/7
1 10H-PREGST/-, JQ3/10H-CARROLL/-, JQ4/5HHOLT/-, JQ5/10H-HERRMANN
L 1, JQ6/10H-P-ALPHA/-, JQ7/10H-VARIA ST/-, JQ8/7THRENGT/-
J 2, JQ9/10H-FRACRT/-. JQ10/6H MPE/-, JQ11/10H-LIINEAR V/, JQ12/10H
3 9HIC void/-, JQ13/10H-DYMIC/-, JQ14/8HPORHOLT/-, JQ15/10H
4 10H-PHOHOLT/-, JQ16/8HBUCHTER/-, JQ5/5HSTAT/-, JQ6/5HRATE/-
5 , JQ17/10H-DUCIL /, JQ18/9HFRACTURE/-
C C *** ZEROING OF ARRAYS
C C IF (LS-1) 1,8,1000
1 DO 5 I = 1, 6
5 NPM(I) = 0.
DO 50 I = 1, 4
AK(I)*MUP(I)=K1C(I)=0.
50 NREG(I) = 0.
DO 51 I = 1, 12
51 TPH(I) = DADP(I) = EPS(I) = DEL(I) = ALE(I) = APC(I) = 0.
DO 52 I = 1, 60
52 YADD(I)=COSQ(I)*C(I(I)=PORA(I)=PORS(I)=POC(I)=0.
C 720
DO 53 I=1,72
53 RHOP(1)=0.
DO 54 I=1,96
54 TERR(I)=0.
MP=0 $ DPDDJ=DPDEJ=0. $ LS=1 $ CJ=1.
MP=MP+1
NPM(M) = MP
PEST
C  *****************************************
C READING OF INPUT DATA
C *****************************************
C C *** READ DATA USED BY ALL MODELS.
C C READ(IN,935)A1,KCS(MP),KTS(MP),KRS(MP),A2,KCD(MP),KTD(MP)
C 1 ,KRD(MP)
C WRITE(6,935)A1,KCS(MP),KTS(MP),KRS(MP),A2,KCD(MP),KTD(MP)
C 1 ,KRD(MP)
C WRITE(6,960)IDD,IN,JQ1
C
C READ(IN,919)A1,AK(MP),A2,MUP(MP),A3,YZERO,A4,RHOPCMP, 6, 1)
C WRITE(6,920)A1,AK(MP),A2,MUP(MP),A3,YZERO,A4,RHOP(MP,6, 1)
C WRITE(6,960)IDD,IN,JQ1
C ALF0=RH0S/RH0P(MP,6,1)
C IF (AK(MP) .GT. 0. .AND. AK(MP) .LE. EQSTCM*RHOPCMP,6,1)
C /RH0S) GO TO 20
C IF (AK(MP) .GT. 0.) GO TO 10
C IF AK IS TOO LARGE, IT IS REDUCED TO THE MAXIMUM PERMITTED.
C AK(MP)=EQSTCM*RH0P(MP,6,1)/RHOS
C WRITE(6,950) AK(MP), MUP(MP)
C WRITE(6,960)IDD,OUT,JQ1
C YADDM=0.666667*YZERO  $ MUP(MP)=1.333333*MUP(MP)
C J2=J0H0P0M,  $  J3=J4=1H
C N=1
C KCSM=KCS(MP)  $  KCDM=KCD(MP)  $  KTSM=KTS(MP)  $  KTDM=KTD(MP)
C IF (KTSM .EQ. 0) J3=5HTENS,   $   IF (KRSM .EQ. 0) J4=5HREC0M
C GO TO 501
C
C ** READ FOR RATE-INDEPENDENT COMRESSIVE MODEL.
C GO TO (490,510,520,530,540,550)KCSM
C CONTINUE
C
C *** READ AND INITIALIZE FOR POREST.
C READ(IN,939)A1,NREG(MP)
C WRITE(6,940)A1,NREG(MP)
C WRITE(6,960)IDD,IN,JQ2,IDD,JQS,J2,J3,J4
C READ(IN,909)A1,tRHOPCMP,I^J, 1=1
C WRITE(6,910)A1,(RHOPCMP,I,N),1=1,5)
C WRITE(6,960)IDD,IN,JQ2
C DO 498 1=1,5
C COSQ(MP, I ,N) = 0.15
C 496   C1(MP, I,N) = 0.15
C 501   DECODE (3,915,TEMP) A1, A2
C 502   IF (A1 .EQ. 1HC .AND. (A2 .EQ. 1 HO .OR. A2 .EQ. 1H0)) GO TO 502
C 503   IF (A1 .EQ. 1HC .AND. A2 .EQ. 1H1) GO TO 503
C 504   GO TO 504
C 505   DECODE (80,910,TEMP) A1, (COSQ(MP, I,N),1=1,5)
C 506   WRITE(6,910)A1,(COSQ(MP, I,N),1=1,5)
C 507   WRITE(6,960)IDD,IN,JQ2
C 508   GO TO 501
C 509   DECODE (80,910,TEMP) A1, (C1(MP, I,N),1=1,5)
C 510   WRITE(6,910)A1,(C1(MP, I,N),1=1,5)
C 511   WRITE(6,960)IDD,IN,JQ2
C 512   GO TO 501
C 513   CZJ = COSQ(MP,5,1)
C 514   C2J = C1(MP,5,1)

271
SUBROUTINE PEST (Continued)

NP=NREG(MP)
DECOD(80,920,TEMP)A1,P1
WRITE(6,920)A1,P1
WRITE(6,960)IDD,IN,JQ2
POR(A,MP,1,N)=P1 $ PORB(MP,1,N) = PORC(MP,1,N) = 0.
505 NQ=1,NP
READ(IN,919)A1,P2,A2,DRH0,A3,YADDMP,MP,NQ,0.
WRITE(6,920)A1,P2,A2,DRH0,A3,YADDMP,MP,NQ,0.
WRITE(6,960)IDD,IN,JQ2
IF (NQ .GE. NP) GO TO 5045
IF (RH0P(MP,NP+1,N).GT. RH0S) GO TO 5045
RH0P(MP,NP+1,N)=RH0S*(1.+TSQ(0,P2,0.,EQSTCM,EQSTD,EGSTSM,1
EQSTHM,EQSTEM,RHOS,EQSTNM,0.,EGSTVM,EQSTNM,NCYC))
WRITE(6,932)RH0P(MP,NP+1,N)
WRITE(6,960)IDD,OUT,JQ2
5045 DRH0=RH0P(MP,NQ+1,N)-RH0P(MP,NQ,N)
AA=P2-P1-4.*DELP*RH0P(MP,NQ+1,N)/DRH0
POR(A,MP,NQ+1,N)=P1+RH0P(MP,NQ+1,N)/DRH0*AA
BB=P2-P1-4.*DELP*(RH0P(MP,NQ+1,N)+RH0P(MP,NQ,N))/DRH0
PORB(MP,NQ+1,N)=-RH0P(MP,NQ+1,N)*RH0P(MP,NQ,N)/DRH0*BB
PORC(MP,NQ+1,N)=4.*DELP*(RH0P(MP,NQ+1,N)+RH0P(MP,NQ,N))/DRH0)**2
YADDMP(MP,NQ,N)=YADDMP(MP,NQ,N)/DRH0
505 P1=P2
YADDMP(MP,NP+1,N)=0.
RH0P(MP,N,N)=RH0P(MP,NP+1,N)
GO TO 600
510 CONTINUE
C ** READ AND INITIALIZE FOR PORHOLT.
** C
READ(IN,919)A1,RH0P(MP,1,N)
WRITE(6,920)A1,RH0P(MP,1,N)
READ(IN,919)A1,RH0P(MP,5,N),A2,DRH0,A3,PY,A4,YADDMP,MP,1,N)
WRITE(6,920)A1,RH0P(MP,5,N),A2,DRH0,A3,PY,A4,YADDMP,MP,1,N)
WRITE(6,960)IDD,IN,JQ15,IDD,JQ5,J2,J3,J4
IF (RH0P(MP,5,N).LT.100.) GO TO 512
P2 = RH0P(MP,5,N)
RH0P(MP,5,N)=RH0S*(1.+TSQ(0,P2,0.,EQSTCM,EQSTD,EGSTSM,EGSTHM,EQSTEM,RHOS,EQSTNM,0.,EGSTVM,EQSTNM,NCYC))
WRITE(6,932)RH0P(MP,5,N)
WRITE(6,960)IDD,OUT,JQ15
512 RH0P(MP,2,N)=RH0P(MP,1,N)*PY/AK(MP)+1.
RH0P(MP,3,N)=RH0S/(1.-RH0S*PY/RH0P(MP,2,N)/EGSTCM)
ALFE=RH0P(MP,3,N)/RH0P(MP,2,N)
R=RH0P(MP,3,N)/RH0S
POR(A,MP,1,N)=ALFE*(ALFE*RH0P(MP,2,N)/EGSTCM*DRH0-R/RH0S)
R1=POR(A,MP,1,N)/RH0P(MP,5,N)-RH0P(MP,2,N)
PORB(MP,1,N)=(RH0P(MP,5,N)-RH0P(MP,3,N))/
(10.00*(RH0P(MP,5,N)-RH0P(MP,2,N))*2-1
YADDMP(MP,1,N)=YADDMP(MP,1,N)/RH0P(MP,5,N)-RH0P(MP,2,N)
WRITE(6,932)RH0P(MP,5,N)
WRITE(6,960)IDD,OUT,JQ2
520 CONTINUE
C ** READ AND INITIALIZE FOR CARROLL-HOLT.
** C
READ(IN,919)A1,YCH,A3,EPs(MP,N),A4,TER(MP,7,N)
WRITE(6,920)A1,YCH,A3,EPs(MP,N),A4,TER(MP,7,N)
WRITE(6,960)IDD,IN,JQ3,JQ4,JQ5,J4
IF (A1 .EQ. 10H) YCH =
GO TO 525
PY = YCH
IF (ABS(EPS(MP,N)) .LT. 1.) GO TO 526
P2 = EPS(MP,N)
R1=-RHO(MP,1,N)/RH0S
C ** PY AND PC KNOWN
* C
RH0P(MP,5,N)=RH0S*(1.+TSQ(0,P2,0.,EGSTCM,EQSTD,EGSTSM,EGSTHM,EQSTNM,0.,EGSTVM,EQSTNM,NCYC))
BB = BBMIN = (RH0P(MP,5,N)/RH0S-1.)*EQSTCM/PY
ALFA = 1./(-R1
DEL(MP,N) = PY/(EGSTCM*ALFA(-R1-RH0P(MP,1,1)/RH0S))
IF (.EQ. 1.) BBMIN = AMAX1(BB,1.,DEL(MP,N))
BBMIN = AMAX1(BBMIN,0.24627*ALFA**2+2.8512*ALFA-1.9633)
IF (BB.GT.1.) BBMIN GO TO 521
BB = BBMIN
RH0P(MP,5,N)=RH0S*(1.+BB*PY/EGSTCM)
EO = 1./BB
SUBROUTINE PEST (Continued)

521 E0 = RV**BB

5215 B0 = ALOG(E0)/ALOG(RV+E0)

5215 E2= E1 = (RV+E0)**BB

IF (ABS(E0-E1) .LT. 1.E-05*E1) GO TO 524

522 NW = 0

522 E2 = E1 = (RV+EO)**BB

522 IF (ABS(B2-B1) .LT. 1.E-5 .OR. AW .GE. 10.) GO TO 524

523 E0 = E1 = B0 = B1 = E2 = B2

524 EPSCMP.N) = E2

DELMP.N) = (1.-RHOP(MP,1,N))/RHOS)/ALOG(EPS(MP,1,N))

525 CALL EQST(0,RHOPCMP,5,N),P2,M,1.,A1,A2)

526 DEL(MP,N) = 0.66667*YCH/EQSTCM

526 IF (YCH .LT. 0.) EPS(MP,N) = AMAX1(EPS(MP,N),ABS(DEL(MP,N)))

527 EPS(MP,N) = 1.+EPS(MP,N)

550 CONTINUE

C * ** READ INPUT AND INIT FOR HERRMANN P-ALPHA. ** C PEST

READ(IN,919)A1,PC,A3,PY

WRITE(6,920)A1,RHOP(MP,1,1),A2,PC,A3,PY

WRITE(6,960)IDD,OUT,JQ3,JQ4

P0RA(MP,1,N) = PY  $  P0RC(MP,1,N) = PC

GO TO 600

C ** READ AND INIT FOR HENDRON. ** C PEST

GO TO 600

C ** READ AND INIT FOR TBS. ** C PEST

GO TO 600

600 IF (N .GE. 2) GO TO 640

J2=5HTENS  $  J3=J4=1H

C ** READ FOR RATE-INDEPENDENT TENSION MODEL. ** C PEST

IF (KTSM .EQ. 0 .AND. KCSM .EQ. 3) GO TO 610

GO TO (615,620,520) KTSM

C ** REPEAT CARROLL-HOLT ARRAY FOR N=2. ** C PEST

610 ALE(MP,2) = -ALE(MP,1) $ EPS(MP,2) = EPS(MP,1)

DEL(MP,2) = -DEL(MP,1) $ TER(MP,7,2) = TER(MP,7,1)

APC(MP,2) = 1./1. -ALE(MP,N))

RHOP(MP,5,N) = RHOS/APC(MP,2)

WRITE(6,932)RH0P(MP,5,N)

WRITE(6,960)IDD,OUT,JQS,JQ4,JQS,J2

GO TO 600

615 CONTINUE

C ** READ AND INIT FOR VARIABLE STRENGTH. ** C PEST

READ(IN,919)A1,TER(MP,5,N),A2,TER(MP,7,N)
SUBROUTINE PEST (Continued)

WRITE(6,920)A1,TER(MP,5,N),A2,TER(MP,7,N) PEST 302
WRITE(6,960)1DD,IN,JQ7,JQ8,JQS,J2 PEST 303
GO TO 600 PEST 304

620 CONTINUE PEST 305

C ** READ AND INIT FOR KIC. ** C PEST 306
READ(IN,919)A1,KIC(MP),A2,TER(MP,7,N) PEST 307
WRITE(6,920)A1,TERCMP,S,N^AZ.TERCMP.T,N) PEST 308
WRITE(6,960)1DD,IN,JQ7,JQ8,JQS,J2 PEST 309
GO TO 600 PEST 310

640 CONTINUE PEST 311
IF (N .EQ. 3) GO TO 700 PEST 312
N = 3 PEST 313
J2=5HRECPM PEST 314

C *** READ FOR RATE-INDEPENDENT RECOMPRESION MODEL. *** C PEST 315
IF (KICM .GT. 0) GO TO 660 PEST 316

C ** REPEAT ARRAYS KRS=KCS. ** C PEST 317
GO TO (641,645,647,648) KCSM PEST 318

C ** PORQST. ** C PEST 319
641 NPP = NP+1 PEST 320
DO 642 NQ = 1,NPP PEST 321
PORA(MP,NQ,3)=P0RA(MP,NQ,1) $ YADDP(MP,NQ,3)=YADDP(MP,NQ,1) PEST 322
PORB(MP,NQ,3)=PORB(MP,NQ,1)  $ PORC(MP,NQ,3)=PORC(MP,NQ,1) PEST 323
GO TO 700 PEST 324

C ** PORHOLT. ** C PEST 325
645 PORA(MP,1,3)=P0RA(MP,1,1) $ PORB(MP,1,3)=PORB(MP,1,1) PEST 326
RHOP(MP,5,3)=RHOP(MP,5,1) $ RHOP(MP,2,3)=RHOP(MP,2,1) PEST 327
RHOP(MP,3,3)=RHOP(MP,3,1) $ YADDP(MP,1,3)=YADDP(MP,1,1) PEST 328
GO TO 700 PEST 329

C ** CARROLL-HOLT MODEL. ** C PEST 330
647 APC(MP,3)=APC(MP,1) $ EPS(MP,3)=EPS(MP,1) PEST 331
DEL(MP,3)=DEL(MP,1) $ RHOP(MP,5,3)=RHOP(MP,5,1) PEST 332
RHOP(MP,1,3)=RHOP(MP,1,1) PEST 333
GO TO 700 PEST 334

C ** HERRMANN P-ALPHA MODEL. ** C PEST 335
648 PORA(MP,1,3)=P0RA(MP,1,1) $ PORC(MP,1,3)=PORC(MP,1,1) PEST 336
RHOP(MP,1,3)=RHOP(MP,1,1) PEST 337
GO TO (490,510,520,530,540,550) KRS M PEST 338

C ** READ FOR RATE EFFECTS IN COMPRESSION. ** C PEST 339
700 N = 1 PEST 340
J2=5HCOMP, $ J3=J4=1H PEST 341
IF (KTDM .EQ. 0) J3=5HTENS, $ IF (KRD M .EQ. 0) J4=5HRECPM PEST 342
IF (KCDM .LE. 0) GO TO 750 PEST 343
GO TO (750,720,730,740) KCDM PEST 344

720 CONTINUE PEST 345

C ** READ AND INIT FOR LINEAR VISCOUS VOID(C) OR DUCTILE FRACTURE(T) PEST 346
READ(IN,909)A1,(TER(MP,1,N),I=1,7) PEST 347
WRITE(6,910)A1,(TER(MP,1,N),I=1,7) PEST 348
IF (N .EQ. 1 .OR. N .EQ. 3) WRITE(6,960)1DD,IN,JQ11,JQ12,JQR,J2 PEST 349
1 J2,J3,J4 PEST 350
IF (N .EQ. 2) WRITE(6,960)1DD,IN,JQ17,JQ18,JQR,J2 PEST 351
IF (TER(MP,8,N) .EQ. 0) WRITE(6,961)1DD,IN,JQ11,JQ12,JQR,J2 PEST 352
1 J2,J3,J4 PEST 353
730 CONTINUE PEST 354

C ** READ AND INIT DYNAMIC PORHOLT. ** C PEST 355
READ(IN,919)A1,THP(MP,N) PEST 356
WRITE(6,920)A1,THP(MP,N) PEST 357

274
SUBROUTINE PEST (Continued)

740 CONTINUE
C
750 N = N+1
C
755 READ AND INIT DYNAMIC BUTCHER P-ALPHA-TAU.
C
760 READ(IN,919)A1,TPH(MP,N)
C
765 WRITE(6,920)A1,TPH(MP,N)
C
770 DADP(MP,N)=-ALFO/AK(MP)*(1.-AKCMP)*ALFO/EQSTCM)
C
780 WRITE(6,960)IDD, IN,JQ13,JQ14,JQR,J2,J3,J4
790 N = N+1
C
800 READ FOR RATE EFFECTS IN TENSION.
C
810 GO TO(700,755,770,900) N
C
820 J2=SHTENS
C
830 IF (KTDM .GT. 0) GO TO (750,720,760) KTDM
C
840 IF (KCDM .EQ. 0) GO TO 750
C
850 IF (KCDM .GT. 2) GO TO 756
C
860 REPEAT LINEAR VISCOUS VOID FOR DUCTILE FRACTURE.
C
870 TER(MP,1,2)=TER(MP,1,1)
C
880 TER(MP,3,2)=TER(MP,3,1)
C
890 TER(MP,5,2)= -TER(MP,5,1)
C
900 TER(MP,7,2) = TER(MP,7,1)
C
910 GO TO 750
C
920 C READ BRITTLE FRACTURE AND FRAGMENTATION.
C
930 CONTINUE
C
940 CONTINUE
C
950 GO TO 750
C
960 C READ FOR RATE EFFECTS IN RECOMPRESS!
C
970 J2=SHRECOM
C
980 IF (KRDM .GT. 0) GO TO 800
C
990 IF (KCDM .EQ. 0) GO TO 900
C
1000 GO TO(900,720,730,740) KRDM
C
1010 RETURN
C
1020 FORMAT(8A10)
C
1030 FORMAT(A10,7E10.3)
C
1040 FORMAT(A10,1P7E10.3)
C
1050 FORMAT(1X,2A1)
C
1060 FORMAT(4(A10,E10.3))
C
1070 FORMAT(4(A10,E1PE10.3))
C
1080 FORMAT(* PY=*1PE10.3, * PC=*1PE10.3, * EPS=*1PE10.3)
C
1090 FORMAT(* ABSOLUTE VALUE OF CONSOLIDATION PRESSURE WAS CHANGED TO
1091 BE WITHIN ALLOWABLE RANGE*)
1092 FORMAT(/)
C
1093 FORMAT(2(A10,E16.12,12),12)
C
1094 FORMAT(A10,1PE10.3)
C
1095 FORMAT(A4,A10,1PE10.3)
C
1096 FORMAT(*)
COMPUTATION OF PRESSURE DURING WAVE PROPAGATION.

1000 MP = NPM(M)
IHH = H

** COMPUTE BULK AND SHEAR MODULI APPROPRIATE TO CURRENT E AND D. **

TF = 1. + E*EQSTGM*RHOS/EQSTCM
DREF = D*TF
RUV1 = ABS(RUV) $ ALFD1 = 1./1. - RUV1
IF (RUV1 LT. 1. AND DREF/RHOS LT 1. - ABS(RUV)) GO TO 2000
IF (NCYC LE. 1) ALFS = RHOS/RHOP(MP,6,1)
IF (F .EQ. 0.) GO TO 1800
IF (H .EQ. 5R S OR H .EQ. 5R M) GO TO 1800
RHOPV = RHOS/(TF + (RHOP(MP,5,1) - RHOS)*F
MP = NPM(M)
IH = H
COMPUTE BULK AND SHEAR MODULI APPROPRIATE TO CURRENT E AND D.
TF = 1. + E*EQSTGM*RHOS/EQSTCM
DREF = D*TF
DREF = AMAX1(DREF, RHOP(MP,6,1)/TF)
ALF = AMAX1(1.0, RHOPV/RHOM)
ELK = (EQSTCM/AK(MP) - ALFZ)/(ALFZ - 1.)
ELG = (1. - MUP(MP)*F/MUM)/1. - 1./ALF2)
BULK = EQSTCM*F/(ALF + ELK*(ALF - 1.))
MUM = AMAX1
C = SQRT((BULK + MUM)/D)
IF (NCYC .EQ. 0) PRINT 2300,D,BULK,MUM,C,F,ELK,ELG,RHOPV,MP,6,1,9/12/79 13

** COMPUTE PRESSURE FROM ELASTIC RELATIONS. **

PEL = P + BULK*(D - DOLD)/(0.5*(D + D0LD)) + EQSTQMXRHOS/EQSTCM*(E - EOLD)
BRANCH TO TENSILE OR COMPRESSIVE ROUTES.

IF (PEL LT. 0.) GO TO 1500
COMPRESSION PATH.
KCRS = KCS(MP)
N = 1
IF (H .EQ. 5R S AND H .EQ. 5R M) GO TO 1090
KCRS = KRS(MP)
IF (KRS(MP) .EQ. 0) KCRS = KCS(MP)
IF (KCRS GT. 1500) GO TO 1090

** CALCULATION OF COMPACTION CURVE. **

NC = 0
PST = 0.
IF (DREF .GT. RHOP(MP,5,N)) GO TO 1109
1105 NC = NC+N1
IF (DREF .GT. RHOP(MP,NC,N)) GO TO 1105
PST = F*(PORA(MP,NC,N) + PORB(MP,NC,N)/DREF + PORC(MP,NC,N)/DREF**2)
NQ = MAXO(NC-1)
CZJ = COSQ(MP,NQ,N) $ CWJ = C1(MP,NQ,N)
YADDM = YADDP(MP,NQ,N)
YADD = YADDP(MP,NQ,N)
IF (PST LT. PST) GO TO 1300
PST = PST
IH = 5R S $ PJ = PS H = 5R S $ RVV = 0.
GO TO 1900

** CHECK FOR CONSOLIDATION IN LAST POROUS REGION. **

IF (DREF .GT. RHOS) GO TO 1300
IF (RUV LT. 1. AND DREF/RHOS LT 1. - ABS(RUV)) GO TO 2000
IF (NCYC,LE. 1) ALFS = RHOS/RHOP(MP,6,1)
IF (EQSTCM .LT. 1.) GO TO 2000
COMPUTE PRESSURE FROM ELASTIC RELATIONS.
PEL = P + BULK*(D - DOLD)/(0.5*(D + D0LD)) + EQSTQMXRHOS/EQSTCM*(E - EOLD)
BRANCH TO TENSILE OR COMPRESSIVE ROUTES.

IF (PEL LT. 0.) GO TO 1500

** PORHOLT MODEL. **

1109 GO TO (1100,1120,1140,1160,1180)
1090 GO TO (1100,1120,1140,1160,1180) KCRS
1109 GO TO (1100,1120,1140,1160,1180) KCRS
1100 NC = 0
1105 NC = NC+1
1109 GO TO (1100,1120,1140,1160,1180) NPRM
1110 CALL EQST(E,D,PS,M,CJ,DPDDJ,DPDEJ)
1112 CALL ESA(1.5,M,CJ,D,PS,DPDDJ,DPDEJ)
1114 CALL EQSTPF(1.5,M,CJ,D,PS)
1116 CALL (PS,JT.PST) GO TO 1300
PST = PS
IH = 5R S $ PJ = PS H = 5R S $ RVV = 0.
GO TO 1900

** PORHOLT MODEL. **

1120 DREF = AMAX1(DREF, RHOP(MP,1,N))
SUBROUTINE PEST (Continued)

1126 CALL EQST(0., DS, PS, M, CJ, DPDDJ, DPDEJ)
GO TO 1134

1128 CALL ESA(1.5, M, CJ, DS, 0., PS, DPDDJ, DPDEJ)
GO TO 1134

1130 CALL EQSTPF(1.5, M, CJ, DS, 0., PS)
PST = PS/ALFS*F
GO TO 1108

C
C ** CARROLL-HOLT MODEL.

1140 BNEW = 1.0
IF (DREF .GT. RHOPCMP(MP, 5, N)) GO TO 1143
BNEW = BP = DREF/RHOS
IF (BNEW .GT. 2.-1./APC(MP,N)) BNEW = 1.+0.5*(BP-1./APC(MP,N))
NW = 0
B1 = BP+DEL(MP,N)*ALOG(EPS(MP,N)-BNEW)
BNEW = AM(N1(BNEW+(B1-BNEW)/(1.+DEL(MP,N)/(EPS(MP,N)-BNEW)),0.99999)
NW = NW+1
AW = NW
IF (ABS(BNEW-B1) .GT. L.E-6 .AND. AW .LT. 10.) GO TO 1141
DS = DREF/BNEW
GO TO (1145, 1147, 1149) NPRM

1145 CALL EQST(0., DS, PS, M, CJ, DPDDJ, DPDEJ)
GO TO 1155

1147 CALL ESA(1,5, M, CJ, DS, 0., PS, DPDDJ, DPDEJ)
GO TO 1155

1149 CALL EQSTPF(1,5, M, CJ, DS, 0., PS)
PST = PS*BNEW*F
GO TO 1108

1160 CONTINUE

C
C ** HERRMANN P-ALPHA.

1170 CALL EQST(E, DS, PS, M, CJ, DPDDJ, DPDEJ)
GO TO 1178

1172 CALL ESA(1,5, M, CJ, DS, E, PS, DPDDJ, DPDEJ)
GO TO 1178

1174 CALL EQSTPF(1,5, M, CJ, DS, E, PS)
PST = PS/ALFS
IF (PEL .LT. PST) GO TO 1300
PJ = PST
GO TO 1300

1180 CONTINUE

1300 PJ = PEL
IF (PST .LT. PEL) PJ = PST

C
C ** COMPUTE RELATIVE VOID VOLUME (RVV)

PST = TSQE(1, PJ*RHOS/D, EQSTGM*RHOS*E, EQSTCM, EQSTDM, EQSTSM, EQSTGM,
SUBROUTINE PEST (Continued)

1 EQSTHM, EQSTEM, RHOS, EQSTNM, E, EQSTVM, EQSTAM, NCYC) PEST 601
IF (PJ .NE. 0.) RVV=AMAX1(1.-PJ/PTH, 0.) PEST 602
IF (PJ .EQ. 0.) RVV=AMAX1(0., 1.-D/PTH) PEST 603
ALFS=1./(1.-RVV) PEST 604
IF (AST1 .EQ. 0.) AST1 = ALFS PEST 605
IF (PEL .GT. PST) GO TO 1310 PEST 606
IF (1H .NE. 5R S) GO TO 1900 PEST 607
RVV = 0. $ H = 5R S $ GO TO 1900 PEST 608

C *** DYNAMIC PRESSURE. C PEST 609
C C PEST 610
1310 KCRD=KCD(MP) PEST 611
IF (H .EQ. 5R S .AND. KRD(MP) .NE. 0) KCRD = KRD(MP) PEST 613
IF (KCRD .GT.1) GO TO 1320 PEST 614
C ** NO RATE-DEPENDENCE. C PEST 615
C IN: C PEST 616
1320 PELS=TSQG(1, PEL*RHOS/D, EQSTHM*RHOS, EQSTNM*EQSTVM, EQSTAM, NCYC) PEST 618
C C PEST 620
C C PEST 621
1340 VVE = 1.-1./ALFL PEST 622
DV = DV1 = 1./D-DOLD PEST 623
NLOOP=MAX1(1., -DV*EQSTCM*D/AMAX1(PST,P)*ALF+0.8, -4.*TER(MP,1,N)*DT PEST 625
1 *(P-PST)) PEST 627
VOLD = 1./DOLD $ VSO = (1.-RVV1)/DOLD PEST 629
NTRY = 0 PEST 630
RVVL = RVV1 PEST 631
PTHL = PTHO = PST1*AST1 PEST 633
PSO = AMAX1(P,PST1)/(1.-RVV1) PEST 636
IF (PST1 .LT. 0.) PSO=PTHL=PTHO=0. PEST 637
IF (1.-RVV1 -1.*/AST1 .LT. 0. .AND. PSO .GT. PTHO) GO TO 13401 PEST 638
RVPO = -1./(DOLD*EQSTCM) PEST 639
DRVP = 0. PEST 640
GO TO 13403 PEST 641
13401 RVPO = (1.-RVV1-1./AST1)/DOLD/(PSO-PTHL) PEST 642
DRVP = (RVV*VVE)/D/(PELS-PTH)-RVPO PEST 643
13403 VSTHO = 1./(DOLD*AST1) PEST 644
IF (PST1 .LE. 0. .OR. PST1 .GT. P) PTHL=PTHO=PTH PEST 645
DVSTH = (1.-RVV)/D-VSTHO PEST 646
DVDP = (VVE/D-RVV1/DOLD)/(PELS-PSO) PEST 647
DPTH = PTH-PTH0 PEST 648
1341 DELV = DV/NLOOP $ VH = VOLD $ DTN = DELV/DV*DT PEST 649
A1 = TER(MP,1,N)*DTN PEST 650
C BEGIN DO LOOP FOR SUBCYCLING PEST 651
DO 1347 NL = 1,NLOOP PEST 652
VH = VH+DELV $ RATIO = (VH-1,/DOLD)/DVO PEST 653
RVP = RVPO+DRVP*RATIO PEST 654
VSTH = VSTHO+DVSTH*RATIO PEST 655
PTH = PTH0-DPTH*RATIO PEST 656
C FIRST ESTIMATE OF PRESSURE IN SOLID PEST 657
DP = AMAX1(0., PSO-PTHL) PEST 658
XG = 1. $ IF (DP .GE. 0. ) XG = EXP(A1*DP) PEST 659
PLC = PTH0 $ PUP = PELH = AMAX1(P,PST1)/(1.-RVV1)+(PELS-AMAX1(P, P) PEST 660
1 PST1)/(1.-RVV1))$RATIO PEST 661
PSA = PELH $ ZG = RVVL*VH PEST 662
IF (PTH0 .GT. PELH) GO TO 1346 PEST 663
PSJ = (DELV+VSO-VSTH+RVP+PSO-DELV-RVVL*VH*(XP*G*(1.+A1/2.* PEST 664
IF (PTTH .GT. PELH) GO TO 1346 PEST 665
PSJ = (DELV-VSO+VSTH+RVP+PSO-DELV+RVVL*VH*XP*G*(1.+A1/2.)* PEST 666
NC = 0 PEST 667
1342 NC = NC+1 PEST 668
DP = (AMAX1(0., PSJ-PTHH)+AMAX1(0., PSO-PTHL))/2. PEST 669
ZG = RVVL*VH $ IF (DP .GE. 0. ) ZG = EXP(A1*DP) PEST 670
DELVA = VSTH-VSO+RVP*(PSJ-PTHH)+DVDP*(PSJ-PSO)+ZG-RVVL*VH PEST 671
PSA = PSJ PEST 672
AC = NC PEST 673
IF (ABS(DLVA-DELV) .LT. 1.E-5*VH .OR. (PSJ .LE. PTHH .AND. AC PEST 674
1 .GT. 1.)) GO TO 1346 PEST 675
IF (NC .GE. 10) GO TO 1348 PEST 676
C
SUBROUTINE PEST (Continued)

IF (DELVA .GT. DELV) PLO = AMAX1(PSA, PLO)
IF (DELVA .LT. DELV) PUP = AMIN1(PSA, PUP)
C
MAKE 2ND ESTIMATE OF PRESSURE IN THE SOLID
IF (MOD(NC, 2) .EQ. 0) GO TO 1343
PJO = (AMAX1(PSA, PLO))
GO TO 1344
C
INTERPOLATION ESTIMATE OF PRESSURE IN SOLID
1343 PSJ = PSA+(DELV-DELVA)/(DELVB-DELVA)*(PSB-PSA)
GO TO 1344
C
CONCLUSION OF LOOP
1346 RVVL = ZG/VH $ PTHL = PTHH $ PSA = PSO = AMAX1(PTHH, AMIN1
1 (PEL, PSA))
VSO = VH-ZG $ ENT = ENT*VOLD/VH
GO TO 1346
C
PROVISION FOR ABORT FOR ITERATION FAILURE
1347 PJ = (1. - RVVL)*PSA $ RVL = RVVL $ GO TO 1900
1348 NTRY = NTRY+1 $ IF (NTRY .GE. 5) GO TO 1349
1349 WRITE(6, 2349) M, P, DV, DELVA, DELVB
GO TO 1346
C
PORHOLT MODEL - DYNAMIC.
1380 ALFD = TPH(MP, N)*ALFLD + AST1 + ALFSR*(DT-TPH(MP, N))
1 MP, N)*ALFLD-AST1+ALFSR)*EXP(-DT/TPH(MP, N))
1382 DS = ALFD*D
GO TO (1385, 1390, 1395) NPRM
1385 CALL EQST(E, DS, PS, M, CJ, DPDDJ, DPDEJ)
GO TO 1400
1390 CALL ESA(1, 5, M, CJ, DS, E, PS, DPDDJ, DPDEJ)
GO TO 1400
1395 CALL EQSTPF(1, 5, M, CJ, DS, E, PS)
GO TO 1400
1400 PJ = AMIN1(PEL, AMAX1(PST, PSA))
PSJ = TSGE(1, PJ)*RHOS/D, EQSTGM*RHOS, E, EQSTM, EQSTDM, EQSTM, EQSTGM,
1 EQSTHM, EQSTEM, EQSTM, EQSTNM, EQSTEM, NCY)
IF (PJ .NE. 0.) RVVL = AMAX1(0., 1. - PJ/PS1)
IF (PJ .EQ. 0.) RVVL = AMAX1(0., 1. - D/PS1)
GO TO 1900
C
BUTCHER P-ALPHA-TAU
1440 BT = TPH(MP, N)*(ALFLD-ALFD1)/DADP(MP, N)/(PEL-PST)
ALFD = (ALFLD-ALFD1)*BT/DT+ALFD1*EXP(DT/TPH(MP, N))
1 BT/DT
IF (ALFD .LT. ALFD) ALFD = ALFD
IF (ALFD .GT. ALFD) ALFD = ALFD
GO TO 1382
C
TENSILE PATH.
1500 KTSS = KTS(MP)
1501 IF (KTSS .EQ. 0) KTSS = KC(MP)
N = 2
GO TO (1520, 1540, 1560) KTSS
C
VARIABLE STRENGTH.
1520 PTH = TER(MP, 5, N)*F
PST = D*PTH*(1./RHOS+EQSTGM*E/EQSTCM)/(1.+PTH/EGSTM)
GO TO 1600
C
FRACTURE MECHANICS.
1540 GO TO 1520
C
CARROLL-HOLT THRESHOLD STRESS.
1560 PST = DEL(MP, N)*ALGD(EPS(MP, N)-BNEW)
SUBROUTINE PEST (Continued)

BNEW = AMIN1(BNEW+(B1-BNEW)/(1.0+DEL(MP,N)/(EPS(MP,N)-BNEW)),0.9999) PEST 751

19999) PEST 752
NW = NW+1 PEST 753
AW = NW PEST 754
IF (ABS(BNEW+B1) .GT. 1.E-6 .AND. AW .LT. 10.) GO TO 1565 PEST 755
DS = DREF/BNEW PEST 756
GO TO (1570,1572,1574) NPRM PEST 757

1570 CALL EQST(0.,DS,PS,M,CJ,DPDDJ,DPDEJ) PEST 758
GO TO 1580 PEST 759

1572 CALL ESA(1.5,M,CJ,PS,DS,0.,PS,DPDDJ,DPDEJ) PEST 760
GO TO 1580 PEST 761

1574 CALL EQSTPF(1.5,M,CJ,DS,0.,PS) PEST 762

1580 PST = PS*BNEW*F PEST 763
IF (PST .GT. PS) GO TO 1600 PEST 764
PST = PS PEST 765
IH = 5R S PEST 766
IF (PEL .GT. PST) GO TO 1600 PEST 767
PJ = PS PEST 768
H = 5R S $ RVV = 0. PEST 769
GO TO 1900 PEST 770

1600 PJ = PEL PEST 771
IF (H .NE. 5R S) H = 5R T PEST 772
IF (PEL .LT. PST) H = 5R T PEST 773
IF (PEL .LT. PST) PJ = PST PEST 774

C ** COMPUTE RELATIVE VOID VOLUME. (RVV) ** C PEST 775

C PEST 776

C PEST 777

C PEST 778

C PEST 779

C PEST 780

C PEST 781

C PEST 782

C PEST 783

C PEST 784

C PEST 785

C PEST 786

C PEST 787

C PEST 788

C PEST 789

C PEST 790

C PEST 791

C PEST 792

C PEST 793

C PEST 794

C PEST 795

C PEST 796

C PEST 797

C PEST 798

C PEST 799

C PEST 800

C PEST 801

C PEST 802

C PEST 803

C PEST 804

C PEST 805

C PEST 806

C PEST 807

C PEST 808

C PEST 809

C PEST 810

C PEST 811

C PEST 812

C PEST 813

C PEST 814

C PEST 815

C PEST 816

C PEST 817

C PEST 818

C PEST 819

C PEST 820

C PEST 821

C PEST 822

C PEST 823

C PEST 824

C PEST 825
SUBROUTINE PEST (Continued)

VH = VH+DELV $ RATIO = (VH-1./DOLD)/DV0
RVP = RVP+DRVP*RATIO
VSTH = VSTH+DVSTH*RATIO
PTHH = PTHH+DP/PTHH*RATIO
C
FIRST ESTIMATE OF PRESSURE IN SOLID

C
C
1622 PLO = PTHH $ PUP = PELH = AMIN1(P,PST1)/(1.-RVV1)*(PELS-AMIN1(P, PST1)/(1.-RVV1))*RATIO

C
C
1623 NC = NC+1

C
C
1625 PSJ = PSJ+(DELV-DELVA)/(DELVB-DELVA)*(PSB-PSA)

C
C
1627 PSB = PSA $ DELVB = DELVA

C
C
1630 RVVL = (ZG+ZN)/VH $ PTHL = PTHH $ PSA=PSO=AMIN1(PTHH,AMAX1

C
C
1632 CONTINUE

C
C
1635 IF (RVV .GT. TER(MP,7,N)) GO TO 2000

C
C
1640 NTRY = NTRY+1

C
C
1660 GO TO 1800

C
C
1800 GO TO (1805,1810,1815) NPRM

C
C
1805 CALL EQST(E,D,PS,M,C,DPDDJ,DPDEJ)

C
C
1810 CALL ESA(1,5,M,C,D,E,PS,DPDDJ,DPDEJ)

C
C
1815 CALL EQSTPF(1,5,M,C,D,E,PS)

C
C
1840 IF (H .NE. 5*NRE+5) GO TO 1850

C
C
1850 NLOOP = MAX1(3.,-2.**NTRY/(DV*EQSTCM*D/AMIN1(PST,P)/ALF+0.8))

C
C
1900 GO TO 1900

C
C
1960 GO TO 1900

C
C
1891 C
C
C SOLID AND PORSOUS MELT AND SOLID BEHAVIOR

C
C
1892 C
C
C
1893 C
C
C
1894 C
SUBROUTINE PEST (Concluded)

IF (F .EQ. 0.) GO TO 1850
PST=PEL
GO TO 1860
1850 PJ=PST=PEL=AMAX(0.,PS)
IF (PJ .GT. 0.) GO TO 1855
PTH=TSQ(1.,PJ*RHOS/D, EQSTGM,RHOS,E, EQSTCM, EQSTDM, EQSTSM, 
1 EQSTGM, EQSTHM, EQSTEM, RHOS, EQSTNM, E, EQSTVM, EQSTAM, NCYC)
RVV=AMAX(0., 1. - D/PTH)
H = 5R M
GO TO 1860
1855 H=5R M $ RVV=0.
1860 IF (PEL .LT. 0.) GO TO 1500
C ** ENDING ROUTINE.
C IF (H .EQ. 5R M .OR. H .EQ. 5R S) GO TO 1905
1900 DPDDJ=DPDEJ=0.
1905 P=PJ
PST1=PST $ AST1=ALFS
RETURN
C FRAGMENTATION.
2000 P=PST1=TJ=0.
RVV = -ABS(RVV)
AST1 = 1/(1.+RVV)
H = 5R Z
RETURN
2300 FORMAT(* D,BULK,MUM,C,F,ELK,ELG,RH0P1,E**1P9E10.3)
2349 FORMAT(* ITERATION FAILURE,M=12,* P**1PE10.3,* DV**1PE10.3, 
1 * DELVA=1PE10.3,* DELVB=1PE10.3)
END
SUBROUTINE POREQST

SUBROUTINE POREQST(NCALL, IN, M, C, D, DOLD, E, EOLD, F, P, CZJ, CWJ, H, DPDE, POREQST)

1. EOSTCM = EOSTDM = EOSTSM = EOSTSM = MOM = RHOSM = YADMM = NDSM = NPM = J

C ROUTINE READS INPUT DATA FOR POROUS MATERIAL AND COMPUTES PRESSURE

C READ INPUT (NCALL=0), CALL IS FROM GENRA

C INPUT - NCALL, IN, M, MATERIAL PROPERTY CAHDS

C OUTPUT - ORGANIZES DATA AND FILLS AK, MUP, PORA, RHOP, YADDP

C PREPARE - D = RHOP(M,1), CZJ = CZQ(M) = COSQ(M,6)

C YADMM = Y0(M), CWJ = CWQ(M) = CI(M,6)

C C = EXMAT + M3 = SOUND SPEED

C COMPUTE PRESSURE (NCALL=1)

C INPUT - NCALL, IN, M, C, D, DOLD, E, EOLD, F, P = POLD, H, EOSTCM, EOSTSM, RHOSM

C NDSM, NPM

C OUTPUT - C, P, H, CZJ, CWJ, DPDE, MUM, YADDD

C NOTE CHANGE IN INPUT SO THAT FIRST VALUE OF P2 IS YIELD AND

C PERTAINS TO D = RHOP(M,2)

C REAL MUM, MUP

C INTEGER M

C COMMON /PORA, AK(6), MUP(6), NREG(M), PORA(6,5), PORE (6,5), PORG(6,5)

C RHOP (6,6), YADDP(6,5)

C DIMENSION COSQ(6,6), CI(6,6), TEMP(8)

C DATA NAT, NBT, NCT, NDT, NET, NF, 10H - POREQST - 10H, G/CM3

C 1 10H, G/CM3, 10H, DYN/CM2, 10H, DYN/CM/G, 10H =

C IF (NCALL .EQ. 1) GO TO 200

C **** READ INPUT DATA FOR POROUS MATERIAL *****

C READ (IN, 1192) A1, AK(M), A2, MUP(M), A3, YZERO

C PRINT 1130 A1, AK(M), A2, MUP(M), A3, YZERO, IN, NAT, NDT, NF

C READ (IN, 1100) A1, NREG(M)

C CI = 1

C WRITE (6, 1100) A1, NREG(M)

C WRITE (6, 11110) IN, NAT, NBT

C READ (IN, 1120) A1, (RHOP(M, I), I = 1, 6)

C PRINT 1131, A1, (RHOP(M, I), I = 1, 6), IN, NAT, NCT

C DO 50 I = 1, 6

C COSQ(M, I) = 4.0

C 50 CI(M, I) = 0.15

C READ (IN, 11005) (TEMP(I), I = 1, 8)

C DECODE (31125, TEMP) A1-A2

C IF (A1 .EQ. 1HC AND. (A2 .EQ. 1HC .AND. A2 .EQ. 1HC)) GO TO 60

C IF (A1 .EQ. 1HC AND. A2 .EQ. 1HC .OR. A2 .EQ. 1HC)) GO TO 62

C GO TO 65

C DECODE (80, 1120, TEMP) A1, (COSQ(M, I), I = 1, 6)

C PRINT 1131, A1, (COSQ(M, I), I = 1, 6), IN, NAT

C GO TO 55

C 62 DECODE (80, 1120, TEMP) A1, (C1(M, I), I = 1, 6)

C PRINT 1131, A1, (C1(M, I), I = 1, 6), IN, NAT

C GO TO 55

C 65 CZQ = COSQ(M, 6), S0 CWJ = C1(M, 6)

C NP = NREG(M) $ P1 = 0

C DECODE (80, 1192, TEMP) A1P1

C PRINT 1132, A1P1, IN, NAT, NDT

C PORA(M, I) = P1, $ PORP(M, I) = PORC(M, I) = 0

C YADDP(M, I) = 0

C DO 110 N = 1, NP

C READ (IN, 1192) A1, P2, A2, Delp, A3, YADDP(M, N)

C PRINT 1130 A1, P2, A2, Delp, A3, YADDP, M, N, IN, NAT, NDT, NET

C DRHOP(M, N+1) = RHOP(M, N)

C PORP(M, N+1) = P1 * RHOP(M, N+1)/DRH0 (P2-P1-4 * Delp * RHOP(M, N)/DRH0)

C PORB(M, N+1) = - RHOP(M, N+1) * RHOP(M, N)/URHO (P2-P1-4 * Delp * RHOP(M, N+1)/DRH0)

C PORC(M, N+1) = -4 * Delp * RHOP(M, N+1) * RHOP(M, N)/DRH0)*2

C 110 P1 = P2
SUBROUTINE POREQST (Concluded)

170 YADDM = YZERU
   C = SQRT( (AK(M) + 1.333 * MUP(M))/D)
   RETURN

190 C
   **** CALCULATION OF PRESSURE IN A POROUS MATERIAL ****
   TF=1.*EQSTGM/RHOSM/EQSTCM $ DREF=D*TF $ NC=5
   C FIND APPROPRIATE DENSITY REGION OF POROUS RELATIONS
   IF (DREF .GT. RHOP(M,5)) GO TO 280
   P2 = 0.
   IF (DREF .GT. RHOP(M,5) OR H .EQ. 5R OR Q) GO TO 222
   NC=0
   205 NC=NC+1
   IF (DREF .GT. RHOP(M,NC)) GO TO 205
   P2=F*(PORA(M,NC)+PORB(M,NC)/DREF+PORC(M,NC)/DREF*2)
   IF (DREF .LT. RHOSM) GO TO 230
   C CHECK FOR CONSOLIDATION IN LAST POROUS REGION
   222 CALL EQST(E,D,P,MC,J,A1,A2)
   IF (H .EQ. 5R OR Q) GO TO 225
   IF (P < P2) GO TO 230
   Ps*Ps > H*5R*Q $ NC=5 $ RETURN

230 RHOM=RHOP(M,1)/TF $ RHOPV=F*(RHOP(M,5)-RHOSM)/RHOSM/TF
   RATIO=AMIN1((1.-(RHOM-D)/(RHOPV-RHOPV)*RHOPV/(1.-(RHOPV-D))/
   1 (RHOPV+RHOM)*RHOM/D))
   BULK=F*(AK(M)*(EQSTCM+AK(M)))*RATIO
   MUM=F*MUP(M)+MUM*MUP(M)*F)*RATIO
   PBULK=P*BULK*((D-DOLD)/(5.0*(D-DOLD)))+EQSTGM*RHOSM/EQSTCM*(E-EOLD)
   C CHECK WHETHER STATE POINT IS ON INTERMEDIATE OR YIELD SURFACE
   IF (NC .LE. NREAL(M)) YADD = YADDP(M,NC)
   RETURN

250 CSQ = (BULK + 1.333 * MUM)/D
   IF (CSQ .LT. 0.) GO TO 270
   C=CSQ+C**2 $ C=CSQ+C/C + 0.25*C/C/C $ DPDE=0.
   C COMPUTE ARTIFICIAL VISCOSITY COEFFICIENTS

270 RATIO=0.
   DELR=RHOP(M,NC+1)-RHOP(M,NC)
   IF (DELNR .NE. 0.) RATIO=(DREF-RHOP(M,NC))/DELNR
   CJ=(COSQ(M,NC)+COSQ(M,NC+1)-COSQ(M,NC))*RATIO
   CWJ=C1(M,NC)+C1(M,NC+1)-C1(M,NC)) *RATIO
   IF (NC .LE. NREAL(M)) YADDM=YADD(M,NC)
   RETURN

C COMPUTE PRESSURE IN CONSOLIDATED MATERIAL

280 H=5R $ CALL EQST(E,D,P,MC,OPDE,DPDE)

1100 FORMAT(A10,I10,A10,IPE10,3)                POREQ70
1005 FORMAT (8A10)                              POREQ71
1110 FORMAT (1H+79X,7H IND= ,5H, IN=,I2,4A10)   POREQ72
1120 FORMAT (A10,7E10,3)                        POREQ73
1125 FORMAT (1X,2A1)                            POREQ74
1130 FORMAT (3(A10,IPE10,3),20X,7H IND= ,5H, IN=,I2,3A10) POREQ75
1131 FORMAT (A10,1PE10,3,I10,7H IND= ,5H, IN=,I2,3A10) POREQ76
1132 FORMAT (A10,1PE10,3,60X,12H IND= ,IN=,I2,3A10) POREQ77
1192 FORMAT (A10,E10,3))                       POREQ78
   RETURN

END
SUBROUTINE PORHOLT

C BASIC EQUATIONS OF THIS MODEL ARE BY AL HOLT OF LLL.
C
REAL MUM,MUP
DIMENSION RHO(6),RHOC(6),ALFO(6),AK(6),MUP(6),YADDP(6),PY(6)
C
IF (NPART .EQ. 1) (30 TO 200
READ AND INITIALIZE

C READ IN SPECIAL PROPERTIES FOR POROUS MATERIAL
WRITE (6,1010)
WRITE (6,1011)
READ (IN=1001) A1,RHO(M),A2,RHOC(M),A3,TPH(M),A4,PY(M)
WRITE (6,1011) A1,RHO(M),A2,RHOC(M),A3,TPH(M),A4,PY(M)
READ (IN=1001) A1,AK(M),A2,MUP(M),A3,Y0(M),A4,YADDP(M)
WRITE (6,1011) A1,AK(M),A2,MUP(M),A3,Y0(M),A4,YADDP(M)
READ (IN=1001) A1,DPDRHO
WRITE (6,1011) A1,DPDRHO
C INITIALIZY YIELD AND DENSITY FOR GENHAT
YADD = Y0(M)
DH=RHO(M); C=SQRT((AK(M)+1.333*MUP(M))/DH)
C INITIALIZE VARIABLES FOR PORHOLT
ALFO(M)=RHO/S/RHO(M)
RHOC(M)=RHO/(PY(M)/AK(M)+1.)
RHOS(M)=RHO*(PY(M)/EQSTCM+1.)
ALFE=RHOC(M)/RHO(M)
A(M)=ALFE*(ALFE+RHOC(M)/EQSTCM*DPDRHO-(RHOES(M)-RHO)/RHO)
B(M)=(RHOES(M)-RHO)/(RHOES(M)-RHO/M)*1.)

C PHINT 102 A(M),B(M),ROHES(M),RHOE(M)
WRITE (6,1010) CJ=1.
DO 150 I=1,300
150 ALF(J)=0.
RETURN
C

C COMPUUTE PRESSURE

C CONTINUE
IF (ALF(J) .EQ. 0.) ALF(J)=ALFO(M)
C COMPUTE ELASTIC VALUE OF ALF ON UNLOAD OR RELOAD CURVES
C AAKC=ALFO(M)+1.)/(AK(M)+1.)*(EUSTCM+ALFO(M)*)
ALF=ALF(J)*1-.ALF(J)/ALF(J)
C COMPUTE STATIC VALUE OF ALF ON THE FLOW CURVE
C ALFS=(RHOC(M)+1.)*(A(M)+B(M)/(DH=RHOE(M)))'/DH
C CHECK WHETHER FLOW STRESS HAS BEEN REACHED DURING LOADING
C IF (ALF(J) .LT. ALFE) GO TO 250
C COMPUTE DYNAMIC VALUE OF ALF
C ALF(J)=MAX1(1.,(ALFS+TPH(M)*ALF/DT)/(1.+TPH(M)/DT))
C GO TO 255
250 ALF(J)=ALFE
C COMPUTE DENSITY IN THE SOLID
C DS=ALF(J)*DH
POLD=P
C COMPUTE PRESSURE IN THE SOLID MATERIAL
C CALL EQST(EH+DS,PS,M,CJ+DPDEJ)
C COMPUTE GROSS PRESSURE
P=PS/ALF(J)

SUBROUTINE PORHOLT(NPART,IN,M,C,DH,DOLD,EH,EOLD,F,P,1H,J,DPDEJ,
1 EQSTCM,MUM,YADD,RHOS,DT)
SUBROUTINE PORHOLT3
SUBROUTINE PORHOLT4
SUBROUTINE PORHOLT5
SUBROUTINE PORHOLT6
SUBROUTINE PORHOLT7
SUBROUTINE PORHOLT8
SUBROUTINE PORHOLT9
SUBROUTINE PORHOLT10
SUBROUTINE PORHOLT11
SUBROUTINE PORHOLT12
SUBROUTINE PORHOLT13
SUBROUTINE PORHOLT14
SUBROUTINE PORHOLT15
SUBROUTINE PORHOLT16
SUBROUTINE PORHOLT17
SUBROUTINE PORHOLT18
SUBROUTINE PORHOLT19
SUBROUTINE PORHOLT20
SUBROUTINE PORHOLT21
SUBROUTINE PORHOLT22
SUBROUTINE PORHOLT23
SUBROUTINE PORHOLT24
SUBROUTINE PORHOLT25
SUBROUTINE PORHOLT26
SUBROUTINE PORHOLT27
SUBROUTINE PORHOLT28
SUBROUTINE PORHOLT29
SUBROUTINE PORHOLT30
SUBROUTINE PORHOLT31
SUBROUTINE PORHOLT32
SUBROUTINE PORHOLT33
SUBROUTINE PORHOLT34
SUBROUTINE PORHOLT35
SUBROUTINE PORHOLT36
SUBROUTINE PORHOLT37
SUBROUTINE PORHOLT38
SUBROUTINE PORHOLT39
SUBROUTINE PORHOLT40
SUBROUTINE PORHOLT41
SUBROUTINE PORHOLT42
SUBROUTINE PORHOLT43
SUBROUTINE PORHOLT44
SUBROUTINE PORHOLT45
SUBROUTINE PORHOLT46
SUBROUTINE PORHOLT47
SUBROUTINE PORHOLT48
SUBROUTINE PORHOLT49
SUBROUTINE PORHOLT50
SUBROUTINE PORHOLT51
SUBROUTINE PORHOLT52
SUBROUTINE PORHOLT53
SUBROUTINE PORHOLT54
SUBROUTINE PORHOLT55
SUBROUTINE PORHOLT56
SUBROUTINE PORHOLT57
SUBROUTINE PORHOLT58
SUBROUTINE PORHOLT59
SUBROUTINE PORHOLT60
SUBROUTINE PORHOLT61
SUBROUTINE PORHOLT62
SUBROUTINE PORHOLT63
SUBROUTINE PORHOLT64
SUBROUTINE PORHOLT65
SUBROUTINE PORHOLT66
SUBROUTINE PORHOLT67
SUBROUTINE PORHOLT68
SUBROUTINE PORHOLT69
C
C 285
SUBROUTINE PORHOLT (Concluded)

MUM = ((ALFO(M) - ALF(M)) * MUP(M)) / (ALFO(M) - 1.)
YADDP = YADDP(M)
CSQ = (AAKC * EQSTM + 1.333 * MUM) / DH
IF (CSQ .LT. 1.E6) GO TO 270
CSQ = SQRT(CSQ)
RETURN

C **** PRESSURE IN CONSOLIDATED MATERIAL ****
C

300 IH = 5R S
CALL EQST(EH, DH, PM, C, DPDEJ)
RETURN

1001 FORMAT (4(A10, 1PE10.3))
1002 FORMAT (* A • B • RHOES • RHOE = #1P4E13.4)
1010 FORMAT (/)
1011 FORMAT (* READ IN PORHOLT *)
END
SUBROUTINE PRESCR

INTEGER H, POROUS, RINTER, SOLIU, SPALL
REAL MATL, NEM, NET, NEMH, NETH

C

COMMON AZERO(1), CEF (10), DELNM, DELTIM, DISCPT(10), DOLD, DTHO, DTMAX,
1 DMINT, DTMX, DUO, DXOLD, EOLD, F, FAC, FIRST, J, JYCS, JINIT,
2 JFIN, JREZON(15), JSMAX, JSTAR, JTSUB(30), M, MAXPR(30), N, NNCYS,
3 NEDIT, NPERN, NH, NREZON, NSCRB(6), NSEPHT, NSPALL, NTEUT,
4 NTEX, NTR(15), POLD, P6(20), R(30), RLAST, SLAST, SMAH, TEDIT(50),
5 TF, TIME, TJ, TREZON, TS(20), ULAST, UOLD, UZERO, XLAST, XNOW, XOLD
1 XJEDIT(20)
C

HALFSTEP VALUES
COMMON DH, DHOLD, DXUH, EH, ENH, ENUH, NEMH, NETH

CONDITION INDICATORS
COMMON INF, LINTER, MIRROR, NH, Normal, POROUS, RINTER,

CELL LAYOUT
COMMON DXX(30), JBNU(30), JMAT(30), NAUTO, MATL(6), NLA,
1 THK(30)

EQUIVALENCE (LA, A)
DATA (NAME(I), 1 = 1, 33) / 3HX, 3HX0, 3HC, 3HD, 4HPD, 4UPD, 3HE, 3HM, 3HH1,
1 3HM2, 3HH3, 3HNEM, 3HNET, 3HP, 3HH, 3HSDT, 3HS1, 3HT, 3HU, 3HS3,
2 3HMP, 3HV, 3HSD, 3S03, 3SHD, 3S02, 3S03,
3 3RIMP, 3RY, 3RS01, 3RS02, 3RSD3/

JED FIRST CARD OF A(100* ) TO BE READ FROM
JF FIRST WORD OF A( ) TO BE READ INTO
JE COUNTER FOR JEDITs

K CHARACTER COUNTER
JK MAX NUMBER OF CHARACTERS USED AT A TIME
JC COUNTER FOR CHARACTERS ON A CARD
JLOC PERIOD INDICATOR

JNUMC() TITLE ARRAY FOR HEADINGS IN SCRIBE
JEDIT() J(CELL OR COORDINATE) NUMBER

KJ = 1
NLAY1 = NAYER + 1
DO 50 I = 1, NLAY1
       JEDIT(I) = I
   50 CONTINUE

JFIN = NNUMC()

50 NLL = 0
1 IF (H(I+2) .EQ. SPALL) GO TO 70
   NLL = NLL + 1
   JEDIT(NLAYER) = 0
   70 CONTINUE

NLL = NLL + 1
SUBROUTINE PRESCR (Continued)

75  JE=NLAYER*NLL
JFIRST=NLAYER*NLL
JTLAST=0
AR=1H
NCARD=0
K=1
KB=1
JKMAX=70
JF=1
IW=1

C
C SELECT A GROUP OF CHARACTERS
80  DECODE (80,1024,A(4000*JK))(A(L)*L=JF*JKMAX)
JF1=JF-1
JL=4000+JK-1
KP=0
NCARD=NCARD+1
100 IF (K .GT. JKMAX) GO TO 300
IF ((LA(K) .GE. 1HA .AND. LA(K) .LE. 1HZ) .OR.
1 (LA(K) .GE. 1HO .AND. LA(K) .LE. 1H9)) GO TO 140
IF (LA(K) .NE. 1H) GO TO 150
KP=K
140 K=K+1
IF (K .GT. JKMAX) GO TO 160
GO TO 100
150 IF (KB .LT. K) GO TO 160
KB=KB+1
K = K+1
IF (K .GT. JKMAX) GO TO 160
GO TO 100
160 KE=K-1
C
C EXAMINE A GROUP OF CHARACTERS FOR TYPE
NK=KE-KB+1
NNL=KP-KB
JFT=KE
IF (KP .NE. 0) GO TO 220
IF (A(KB) .GE. 1HA .AND. A(KB) .LE. 1HZ) GO TO 180
C
INTEGER DATA
KN =10-NK
ENCODE (10,1021,A1)(AB+L=1+KN)(A(L)*L=KB+KE)
DECOD (10,1020,A1) JEDIT(JE)
IF (JE .NE. NLLayer .OR. IW .NE. 1) GO TO 175
NN(IW)=2HE51
IW=IW+1
175 JTLAST=1
JE=JE+1
GO TO 260
C
C ALPHABETIC DATA
180 IF (JTLAST .EQ. 0) GO TO 210
C
SET TYPE INDICATORS FOR ALL JEDIT OF A SET AFTER THE NEXT
C
ALPHA GROUP HAS OCCURRED
JE=JE-1
NDJ=0
IW=IW-1
NDJ=-(JE-JFIRST+1)
DO 205 I=1,IW
C
CHECK LEGITIMACY OF ALPHA DATA
IF ((NN(I) .AND. 777777000000000000008) .EQ. 3LCOM) GO TO 190
DO 185 IK=1,33
IF (NN(I) .EQ. NAME(IK)) GO TO 190
185 CONTINUE
GO TO 205
190 NDJ=JE-JFIRST+1+NDJ
DO 200 J=JFIRST,JE
JTYPE(J+NDJ)=NN(I)
IF (I .EQ. 1) GO TO 200
JEDIT(J+NDJ)=JEDIT(J)
200 CONTINUE
GO TO 205

SUBROUTINE PRESCR (Continued)

CONTINUE
JET=JFIRST=JE+NDJ+1
JTLAST=0
Iw=1
C
DECODE THE ALPHA GROUP OF A SET
ENCODE (NK,1021,NM(IW))(A(L),L=KB,KE)
Iw=IW+1
GO TO 250
C
JEDIT LISTED AS LAYER AND FRACTION
NNL=KP-KB
JFR=KE-KP+1
KP1=KP+1
KN=10-NNL
ENCODE (10,1021,A1)(AB,L=1,KN),(A(L),L=KB,KP1)
ENCODE (10,1020,A1)NL
DECODE (10,1021,A1) (A(L),L = KB,KE)
DECODE (10,1025,A1) FR
JEND=JBNO(NL)
JBEGL=1
IF (NL .GT. 1) JBEGL=JBEGL(NL-1)+1
OIST=X(JBEGL)*FR*(X(JEN0)-X(JBEGL))
J=JBEGL
IF (X(J) .LT. OIST) GO TO 240
JEDIT(JE)=J=JE+1
JTLAST=1
CONTINUE
K=KB=K+1
KP=0
GO TO 100
C
PREPARE FOR NEXT CARD OF DATA
IF (NCARD .GE. NDJ) GO TO 400
JF=1
JK=1+8*NCARD
JKMAX=70
KDIF=K-KB
IF (KB .EQ. K) GO TO 330
KB1=KB-1
KDIF1=KDIF+1
DO 320 KK=K,KDIF1
320 A(KK)=A(KB1*KK)
JF=1+KDIF1
JKMAX=70+KDIF1
KB=1
K=KB*KDIF1
GO TO 80
330 KB=K=1
GO TO 80
CONTINUE
JE=JE-1
NDJ=0
Iw1=Iw-1
DO 420 I=1,IW1
NDJ=(JE-JFIRST+1)*(I-1)
420 J=JFIRST+JE
JTYPE(J+NDJ)=NN(I)
IF (I .EQ. 1) GO TO 420
JEDIT(J+NDJ)=JEDIT(J)
CONTINUE
NJEDIT=JE+NDJ
C
PRINT AND PUNCH JEDIT VALUES
JFIRST=1
JENO=0
JTYPE=JTYPE(1)
DO 450 J=2,NJEDIT
C
PRESCRIBE VALUES
PRESCR117
PRESCR114
PRESCR119
PRESCR120
PRESCR121
PRESCR122
PRESCR123
PRESCR124
PRESCR125
PRESCR126
PRESCR127
PRESCR128
PRESCR129
PRESCR130
PRESCR131
PRESCR132
PRESCR133
PRESCR134
PRESCR135
PRESCR136
PRESCR137
PRESCR138
PRESCR139
PRESCR140
PRESCR141
PRESCR142
PRESCR143
PRESCR144
PRESCR145
PRESCR146
PRESCR147
PRESCR148
PRESCR149
PRESCR150
PRESCR151
PRESCR152
PRESCR153
PRESCR154
PRESCR155
PRESCR156
PRESCR157
PRESCR158
PRESCR159
PRESCR160
PRESCR161
PRESCR162
PRESCR163
PRESCR164
PRESCR165
PRESCR166
PRESCR167
PRESCR168
PRESCR169
PRESCR170
PRESCR171
PRESCR172
PRESCR173
PRESCR174
PRESCR175
PRESCR176
PRESCR177
PRESCR178
PRESCR179
PRESCR180
PRESCR181
PRESCR182
PRESCR183
PRESCR184
SUBROUTINE PRESCR (Concluded)

IF (JTYPE(J) .EQ. JTYPE) GO TO 445

JEND = J - 1

JNUMB = 10000
JA = 0
JB = (JTYPE .AND. 77777700000000000000B)
JC = (JTYPE .AND. 00000000770000000000B)
IF ((JTYPE .AND. 77777700000000000000B) .NE. 3LCOM) GO TO 429
IF ((JTYPE .AND. 00000000770000000000B) .EQ. 55000000000000B) GO TO 41

427 DECODE (10, 1052, JTYPE) JA

GO TO 428

427 DECODE (10, 1051, JTYPE) JA

428 JNUMB = 3999 * JA

II = 20

GO TO 435

429 DO 430 1 = 1, 33

IF (JTYPE .NE. NAME(I)) GO TO 430

JNUMB = 200 + (I - 1)

II = 1

GO TO 435

430 CONTINUE

435 CONTINUE

DO 440 1 = JFIRST, JEND

IF (II .EQ. 20) GO TO 439

IF (II .EQ. 25) GO TO 437

ENCODE (10, 1430, JTYPE(I)) NAME2(II), JEDIT(I)

GO TO 440

437 II = 1 + 1

ENCODE (10, 1431, JTYPE(I)) NAME2(25), II

GO TO 440

439 ENCODE (10, 1432, JTYPE(I)), JEDIT(I)

440 JNUM(I) = JNUMB

JFIRST = J

JTYPE = JTYPE(J)

445 CONTINUE

IF (J .LT. NJEDIT .OR. IEND .EQ. 1) GO TO 450

JEND = NJEDIT

IEND = 1

JTYPE = JTYPE(J)

GO TO 425

450 CONTINUE

PRINT 1450, (JTYPE(I), I = 1, NJEDIT)
RETURN

1020 FORMAT (I10)

1021 FORMAT (80A1)

1024 FORMAT (10X, 70A1)

1025 FORMAT (F10, 6)

1051 FORMAT (3X, I1, 6X)

1052 FORMAT (3X, I2, 5X)

1430 FORMAT (R5, 1H(.13, 1H))

1431 FORMAT (R5, 1H(.13, 1H))

1432 FORMAT (3HCOM, 12, 1H(.13, 1H))

1450 FORMAT (/* OUTPUT FROM PRESCR*/ (4X, 12, A10))

END
SUBROUTINE REBAR

15 SUBROUTINE REBAR(LL, IN, JC, IC, M, N, IH, DH, DOLD, SX, SY, SZ, TXY, E, P, 
12  ZEVP, ZEV, Y, ROLD, IPRINT)

C SR1 AND SR3 ARE OLD AND NEW STRESSES ON STEEL.
C SR2 AND SR4 ARE OLD AND NEW STRESSES ON CONCRETE.
C ALL STRESSES ARE DEVIATORS EXCEPT SRS ARRAY
C STRESSES ARE POSITIVE IN TENSION, PRESSURE IS POSITIVE IN COMP.
C STRAINS ARE POSITIVE IN TENSION
C PLANE OF REBARS IS INITIALLY NORMAL TO THE X DIRECTION
C THETA IS OLD VALUE OF ROTATION ANGLE, POSITIVE TOWARDS Y
C DTHETA IS INCREMENT OF THETA ON CURRENT CYCLE

1  SR4(4), THET(6), IMC(6), IMS(6), FSTEELC6), ESC(6,20)

IF (LL .GE. 0) GO TO 15
READ 1004, A1, FSTEEL(M), A2, THET(M), A3, IMC(M), A4, IMS(M)
PRINT 1004, A1, FSTEEL(M), A2, THET(M), A3, IMS(M)
1004 FORMAT(A10,E10,3,A10,E10.3,10,A10,I10,A10,I10)
LS = 0
MC = IMC(M)
MS = IMS(M)
SX = SQRT((FSTEEL(M)*(ESC(MS,2)+1.33*ESC(MS,5))+(1.-FSTEEL(M))*ESC(MC,2) + 1.33*ESC(MC,5)))/(FSTEEL(M)*(ESC(MC,1)) + (1.-FSTEEL(M))*ESC(MC,1))
DH = FSTEEL(M)*ESC(MS,1) + (1.-FSTEEL(M))*ESC(MC,1)
Y = ESC(MS,10)
RETURN
IF (ROLD .NE. 0.) GO TO 18
MC = IMC(M)
MS = IMS(M)
FS = FSTEEL(M)
THETA = THET(M)
DSTL = ESC(MS,1)
ROLD = ESC(MC,1)
CONTINUE
MC = IMC(M)
MS = IMS(M)
NTRY = 1
RHOS = ESC(MC,7)
EQSTC = ESC(MC,2)
GRUN = ESC(MC,9)
AMU = ESC(MC,5)
CRIT = 1.E7
TEVPSV = TEVP
ZEVPSV = ZEVP
YSV = Y
IHSV = IH
IPRINT = 0
FS1 = FS = (DOLD - ROLD)/(DSTL - ROLD)
COS2TH = COS(2.*THETA)
SIN2TH = SIN(2.*THETA)
SIN2TH1 = SIN2TH + DTHETA*COS2TH
COS2TH1 = COS2TH - SIN2TH*DTHETA
DE(1) = (DEX + DEY + (DEX-DEY)*COS2TH1)/2. + DEXY*SIN2TH1
DE(2) = (DEX + DEY - (DEX-DEY)*COS2TH1)/2. - DEXY*SIN2TH1
DE(3) = DEZ
DE(4) = -(DEX - DEY)*SIN2TH1/2. + DEXY*COS2TH1
C ROTATE STRESSES TO AXIS OF REBARS
SR(1) = (SX + SY + (SX-SY)*COS2TH)/2. + TXY*SIN2TH
SR(2) = (SX + SY - (SX-SY)*COS2TH)/2. - TXY*SIN2TH
SR(3) = SZ
SR(4) = -(SX-SY)*SIN2TH/2. + TXY*COS2TH
RL = 0. RR = 1.
IF (IPRINT .EQ. 1) PRINT 1120, (SR(I), I=1,4), SX, SY, SZ, TXY, COS2TH,
1  SIN2TH
120 PS = FS1 = -(SR(1) + SR(2) + SR(3))/3.
FS = FS1
PC = PI1 = (P - PS1*FS)/(1. - FS)
DO 170 I = 1, 4
SR1(I) = SR(1)*PS1
SR2(I) = SR(1) - SR(1)*FS/(1. - FS)
DEC(1) = DEC(1) + DE(1)*RR
DEC(1) = DEC(1)/ESC(MC,2)/ESC(MS,2)
DEC(1) = DEC(1)*RR - DEC(1)*FS/(1. - FS)
DEC(1) = DEC(1)*RR - DEC(1)*FS/(1. - FS)
SUBROUTINE REBAR (Continued)

180 NC=0
C **********
C BEGINNING OF ITERATION LOOP
C **********
200 NC=NC+1
DO 210 I=1,4
SR3(I)=SR1(I)
SR4(I)=SR2(I)
TEVP=TEVPSV
ZEV=ZEVPSV
Y=YSV
IH=IHSV
PS=PS1 $ PC=PC1
RX=SR4(1)-PC $ RY=SR4(2)-PC $ RZ=SR4(3)-PC $ RXY=SR4(4)
DEST=(DEC(1)+DEC(2)+DEC(3))/3.
RH=R0LD*(2.-DEST)/(2.+DEST)
IF (IPRINT .EQ. 1) PRINT 1002,RH,ROLD,RX,RY,RZ,RXY,ZEVP,TEVP
CALL CAP1 (LS, IN,MCN, IH,RH,ROLD,E,DEC(1),DEC(2), DEC(3),DEC(4),
 1 RX,RY,RZ,RYEVP,IC,TEVP)
210 CONTINUE

C **********
C END OF ITERATION LOOP, RESET FOR NEXT STRAIN INCREMENT
C **********
290 DO 295 I=1,4
SR1(I)=SR3(I)
IHSV=IH
YSV=Y
TEVPSV=TEVP
ZEVPSV=ZEVP
FS=FS*(1.+DES(1))/(FS*(1.+DES(1))+(1.-FS)*(1.+DEC(1)))
295 CONTINUE

7/31/79
SUBROUTINE REBAR (Concluded)

DSTL=D
ROL=RH
PS1=PS $ PC1=PC
RL=RL+RR
IF (RL .LT. .999) GO TO 180
C ENDING ROUTINE

320 CONTINUE

DO 330 I=1,4
SR(1)=SR4(I)*FS+SR3(I)*FS
SRS(I)=SR3(I)-PS
SRS(4)=SR3(4)

THETA2=(THETA+DTHETA)*2.
SIN2TH1=SIN(THETA2) $ COS2TH1=COS(THETA2)
SX=(SR(1)+SR(2)+(SR(1)-SR(2))*COS2TH1)/2.-SR(4)*SIN2TH1
SY=(SR(1)+SR(2)-(SR(1)-SR(2))*COS2TH1)/2.+SR(4)*SIN2TH1
SZ=SR(3)
TXY=(SR(1)-SR(2))/2.*SIN2TH1+SR(4)*COS2TH1
IF (IPRINT .EQ. 1) PRINT 1120, (SR(I),I=1,4), SX, SY, SZ, TXY, COS2TH1,
1 SIN2TH1
P=PC*(1.-FS)+PS*FS
RETURN

NTRY=NTRY+1
IF (NTRY .EQ. 5) IPRINT=1
RR=RR/3.
GO TO 120

1001 FORMAT(I1X,* NC=I5,* DES(1),DEC(1)=*, 1P2E10.3,* PC=*,E10.3,* PS=*
1,E10.3,/,1X,* SR1=*,4E10.3,* SR2=*,4E10.3,/,1X,* SR3=*,4E10.3,
2* SR4=*,4E10.3,/,1X,* (CONCRETE STRESS) SR4(1)-PC=*,E12.5,* (STEEL
3STRESS) SR3(1)-PS=*,E12.5)
1002 FORMAT(* BEFORE CAP, RH,ROL=I1P2E10.3,
1 * RX,RY,RZ,RXY=4E10.3,* ZEVP,TEVP=2E10.3)
1003 FORMAT(* AFTER CAP, RH,ROL=I1P2E10.3,
1 * RX,RY,RZ,RXY=4E10.3,* ZEVP,TEVP=2E10.3)
1120 FORMAT(* SR1,SR2,SR3,SR4=4E10.3/* SX,SY,SZ,TXY=4E10.3/* COS2TH,
1 SIN2TH=2E10.3)

END
SUBROUTINE REDR

C TO PREPARE TAPE4 FOR READING, COPY FROM INPUT TO TAPE4 WITH THE REDR
C COMMAND -- COPYCR(INPUT,TAPE4)
C THE INPUT CARDS SHOULD BE IN NORMAL FORM FOR MATERIALS INPUT;
C WITH BLANKS IN FIRST COLUMN. THE INPUT CARDS ARE BETWEEN 789 REDR
C CARDS BUT THERE ARE NO SEPARATORS BETWEEN MATERIALS.

REWIND IN
NN=0
IDD=1H
READ (IN,100) IND,NAA,NBH
NN=NN+1
IF (EOF(IN)) 15,20
PRINT 110,IDD,NAA,NBH,IND,NAA,NBB
STOP 2254

IF (NA.EQ.NAA .OR. NB.EQ.NBB .AND. NO.EQ.2)) GO TO 10
REWIND IN
IF (NN .EQ. 0) RETURN
DO 40 N=1,NN
READ (IN,100) IND
CONTINUE
RETURN

FORMAT (A1,A9,A10)
FORMAT (15H SEARCHING FOR A1,A9,A10, RH, FOUND A1,A9,A10)
END
SUBROUTINE RELAX (ICON, SD, Y2, DRO, COEF, N, J, M, ANM, ANT, DT, TSR, YD, Y1, INSR)

CALLED BY HSTRESS TO COMPUTE DEVIATOR STRESS ACCORDING TO STANDARD ANELASTIC AND TWO-PARAMETER YIELD MODELS, NDS=1 AND 4.

INPUT - ALL FORMAL PARAMETERS
OUTPUT - SD, ICON, YOLD=YNEW.

DIMENSION TSR(6,30)
YOLD = 0.6667*Y2
YAD = 0.6667*YD
YNOT = 0.6667*Y1

TRLX=TSR(M,15) $ TY=TSR(M,16) $ SDO=SD $ ICOR=ICON
YNEW=YOLD $ L=0
IN=MAX(2,INSR)/2
IF (ICON .EQ. 2) 18,2

INITIAL CONDITION OUTSIDE OF ELASTIC ZONE
L=1 $ GO TO (4,3)IN

XPY=EXP(-TY)
YNEW=YNOT*XPY*ABS(COEF)*TY/DT*(1.-XPY)*.5*(1.*SIGN(1.,SI6N(I.,SD)),SIGN(YAD,SD)) $ GO TO 5

CALCULATED YIELD STRENGTH TO ACCOUNT FOR STRAIN HARDENING
YNEW = AMIN1(AMAX1(ABS(SD)*YOLD),YOLD+YAD*ABS(DRO))

IF (YNEW.EQ.0) 32,35

CONTINUE
Y2 = 1.5* YNEW
RETURN
END
SUBROUTINE REZONE

INCREASES CELL SIZES TO GIVE MORE UNIFORM DISTRIBUTION
* STARTS REZONING AT JREZON AND WORKS TOWARD JINIT
* DOES NOT DISTURB LOCATION OF INTERFACES, JEDITs, OR SPALLS

INPUT - NREZON, SSTOPM.
OUTPUT - ARRAY VARIABLES X, C, D, DHL, EHL, H, NEM, NET, P, P
R, S, SHL, U, UHL, YHL, ZHL, AND JEDIT, JBND.

INTEGER H, POROUS, PRESS, RINTER, SOLID, SPALL
REAL MATL, NEM, NET, NEMH, NETH

COMMON AZERO(1), CEF, CKS, DAVG, DELTIM, DISCPT(10), DOLD, DRHO, DTMAX,
1 DMIN, DTN, DTNH, DU, DX, EOLD, F, FAC, FIRST, J, JCYCS, JINIT,
2 JFIN, JREZON(15), JSAX, JSSTAR, JTS, LSUB(30), M, MAXPR(30), N, NCYCS,
3 NEDIT, NPERN, NR, PREDZON, NSCRB(6), NSPALL, NTEX,
4 NEDT, NTR(15), POLD, P6(20), P8(20), Q(30), R(30), RELST, SLAST, SMAX, TEDIT(50),
5 TF, TIME, TJ, TREDZON, TR(20), ULAST, UZERO, XLAST, XNEW, XOLD
1 1 , XJEDIT(20), MS

COMMON DH, DHLAST, DUH, EH, PORT, RH, RHLAST, SHLAST, UH, UHLAST, XH, XHLAST
1 , NEMH, NETH

COMMON INF, LINTER, MIRROR, NORMAL, POROUS, PRESS, RINTER, SOLID, SPALL

COMMON DXX(30), JBND(30), JMAT(30), NAUTO, MATLC(6, 2), NLAYER, NMTRLS,
1 THK(30)

COMMON /COORD/X(200), XO(200), CHL(200), DHL(200), DPDD(200), DPDE(200),
1 EHL(200), H(200, 3), NEM(200), NET(200), PHL(200), PH(200, 3), SDDT(200),
2 SHL(200), T(200), U(200), YH(200), ZHL(200)

COMMON /JED/JEDIT(100), JNUM(100), JTYP(100), NAME2(40), JEDSIZ,
1 MODUS, NERR, NJEDIT, NTAPE

REAL MU, MUM

COMMON /EOS/ EQSTA(6), EQSTC(6), EQSTD(6), EQST(6), EQSTG(6),
1 EQSTH(6), EQST(6), EQSTG(6), EQSTH(6), EQSTC(6), EQSTD(6), EQST(6),
2 EQSTG(6), EQSTH(6), EQSTC(6), EQSTD(6), EQST(6)

COMMON /MELT/ EMELT(6, 8), GSMELT(6, 8), SPH(6), THERM(6, 8)

COMMON /RH/ RH(6), RHOS(6)

COMMON /TSR/ TSR(6, 30), EXMAT(6, 20), TENS(6, 3)

COMMON /Y/ YO(6), YADD(6), MU(6), MUM, YADD

COMMON /IND/ IQEOS(6), INDK(20), NALPHA, NCM(6), NFR(6), NNP(6),
1 NDS(6), NPP(6), NCON(6), NVAR(6)

COMMON /RAD/ SSTOP(9), START(9), SDLRM, SSTOPM, NSPEC, SSJ, ISPLOT(4),
1 XMAX(4), XMIN(4), YMAX(4), YMIN(4), IA(7), IITILE(24), NARZ, TARZ

COMMON/SS/SS(500)

COMMON/PEO/COM(200), LVAR(200), LVMAX

DIMENSION CC(20), EC(20), HC(20, 3), MASS(21), MOM(20, 2), DC(20),
1 PC(20), RC(20), SC(20), XG(20), YC(20), ANEM(20), ANET(20)

DIMENSION NEWJED(100)

DIMENSION ASC(20), PSC(20), RSC(20), RVSC(20), ENSC(20)

DIMENSION SSS(5), SSC(20, 5)

INTEGER HC, HJOLD2

REAL MASS, MOM, MASLAST, MOLAST, MASNEXT

CALL SECOND(XNOW)

SECTION 1 - LOCATE JREZON WITH RESPECT TO MATERIAL AND JEDITs

JREZ=JREZON(NR)

DIT=0.

IF (NREZON .GT. 0) GO TO 7
IF (JTS .GE. JFIN-2) RETURN

CALL EDIT

DTS=(X(JTS+1)-X(JTS))/CHL(JTS)

DTD=AMIN1(2.*DTS,1.4*DTMAX)

JREZ=JTS

JBND=JBD(1)

DO 5 L=1,NLAYER

JBD=L-1

REZONE
SUBROUTINE REZONE (Continued)

DO 4 JS=JBEG,JBNDM
IF((X(JS+1)-X(JS))/CHL(JS) .GT. DTD) GO TO 4
JREZ=JS+1
CONTINUE
JBEG=JBNDM+2
CONTINUE
JREZ=JLAST=MNO(JREZ,JFIN-1)
IF (JREZ.LE.JINIT+1) RETURN
L=0
L=L+1
IF (JREZ .GT. JBND(L)+1) GO TO 8
IF (JREZ .EQ. JBND(L)+1) JLAST=JLAST-1
MASLAST=ZHL(JLAST)
M0MLAST=0.5*MASLAST*U(JLAST)
TLAST=T(JLAST)

C     * SET JOLD, THE OLD COORDINATE VALUE, AND JNEW, THE NEW VALUE
C     * REZONING OCCURS FOR CELLS BETWEEN JOLD AND JLAST. MIDCELL
C     * QUANTITIES ARE SET FOR JLAST-1 WHILE COORDINATE
C     * QUANTITIES ARE REZONED FOR JLAST.
C     * SET DX (CELL DIMENSION) AND XN (COORD TO LEFT OF NEW CELL) FOR
C     * FIRST GROUP OF CELLS TO BE REZONED

XN=X(JLAST-1) $ DX=X(JLAST)-XN
L=L1=L-1
DXX(L)=DX=AMAX1(DX .LT. DTD*CHL(JLAST-1) )
DT=DX/CHL(JLAST-1)
JOLD=JNEW=JLAST-1 $ NCEL=NPART=0
IF (L .NE. 1) GO TO 13

C *** TERMINATION OF REZONABLE SET OF CELLS AT AN INTERFACE (PART 1)
50 IF (L-1) 790,155,52
52 IF (JOLD-JBND(L-1)-1) 790,60,155
60 NPART=1 $ HJOLD2=H(JOLD,2) $ GO TO 500
100 JLAST=JOLD-1
C *** RETURN WITH JNEW SET TO LEFT COORDINATE OF BOUNDARY, JOLD ON
125 HJNEW+1,2)=HJOLD2 $ X(JNEW)=X(JOLD-1)
130 L=L1 $ JBND(L)+1 $ JNEW+1,2=$ T(JOLD-1)+1 $ JNEW=JNEW-1
135 JOLD=JOLD-2 $ XN=X(JOLD)
138 LO=125
WRITE (6,5011) JREZ,DT,DTNH
WRITE (6,5000) LOC,JOLD,JNEW,JLAST,L,NJ,NCEL,NPART
WRITE (6,5015) (I,DXXCI),I=1,L)
C *** TERMINATION AT INITIAL BOUNDARY (PART 2)
150 IF (JOLD-JINIT) 790,155,52
160 NPART=2 $ HJOLD2=H(JOLD,2) $ GO TO 500
200 HJNEW+1,2)=HJOLD2 $ NEWJED(INJ-1)) $ GO TO 205
250 NEWJED(INJ)=JNEW+1
GO TO 50

C *** TERMINATION WHEN NUMBER OF REZONABLE OLD CELLS IS 20 (PART
300 IF ((X(JLAST)-X(JOLD))/DXX(L)-18.) 420,360,360
350 NPART=4 $ GO TO 500
400 JOLD=JOLD-1
LOC=400
C *** RETURN WITH JOLD AT PREVIOUS LOCATION, JNEW SET AT COORDINATE
C *** LEFT. MIDCELL QUANTITIES HAVE BEEN RESET UP TO JNEW+1, COORDI
C *** QUANTITIES UP TO JNEW+2
450 WRITE (6,5000) LOC,JOLD,JNEW,JLAST,L,NJ,NCEL,NPART
GO TO 50

297
SECTION 3 - COMPUTE NEW CELL COORDINATES AND PROPERTIES

NQ=0
LQC=500
WRITE (6,5000) LOC,JOLD,JNEW,JLAST,L,NJ,NCEL,NPART
NCEL=MINO(JLAST-JOLD),MAX1((X(JLAST)-X(JOLD))/DXX(I-)+.65,1.))
IF ((NCEL-1)*(NQ-1) .EQ. 0) GO TO 610

C CHECK WHETHER REGION OF LARGE CELLS LIES TO LEFT
DXMIN=DXX(L)  $  JLASTP=JLAST-1
LOC=601
WRITE (6,5000) LOC,JOLD,JNEW,JLAST,L,NJ,NCEL,NPART
DO 603 JX=JOLD,JLASTP
DELX=X(JX+1)-X(JX)
IF (DELX-DELX) 602,603,603
DXMIN=DELX  $  JXMIN=JX
CONTINUE
IF (DXMIN-0.8*DXX(L)) 604,750,750
JX=JXMIN+1
DO 605 I=JOLD,JXMIN
JX=JX-1  $ DELX=X(JX+1)-X(JX)
IF (DELX-DXX(L)) 605,605,608
CONTINUE  $  GO TO 610
J0LD=JX+1  $  NPART=4
L0C=608
WRITE (6,5000) LOC,JOLD,JNEW,JLAST,L,NJ,NCEL,NPART
NQ=1  $  GO TO 510

BEGIN COMPUTATIONS FOR NEW COORDINATES
NCEL=MIN(JLAST-JOLD,NCEL)
JOLDR=JLAST
DX=(X(JLAST)-X(JOLD))/NCEL
XSTART=X(JLAST)  $  XN=XSTART-DX
**  XN IS NEW COORDINATE LOCATION
**  DX IS NEW CELL DIMENSION
MOMd , 1 )=MOMLAST
MASt(1)=MASLAST
L0C=610
WRITE (6,5002)
WRITE (6,5610)
M=JMAT(L)
IF (NALPHA .GT. 1) XSTART=XSTART**NALPHA
XNACLX=XSTART 10/8/79
DO 650 I=1,NCEL
MASS(I+1)=AMAVQ=AMSLP=ENGY=CS=RS=PS=SX=YS=0.  $  ASUM=PSUM=RSUM=
1 RVSUM=ENSUM=0.
XNAVALF=XN+DX/2.
DXALF=DX
IF (NALPHA .LE. 1) GO TO 611
XNA=XN+**NALPHA
XNAVALF=0.5*(XNA+XNAACL)
DXALF=XNAVACL-XNA
XNAACL=XNA
CONTINUE
DO 612 INS = 1,NSPEC
HC(1,1)=SOLID
HC(1,2)=NORMAL
ANEs=ANETS=0.
HC(1,3)=2
IF (JLAST .LT. 1) GO TO 625
XEN=AXAX=AXAX(1)(X(JLAST),XN)
XJLALF=X(JLAST+1)
XJLALF=X(JLAST)
IF (NALPHA .LE. 1) GO TO 616
XEN=AXAX**NALPHA
XJLALF=XJLALF**NALPHA
XJLALF=XJLALF**NALPHA
CONTINUE
IF (XSTART-XEND) 621,621,619
DMASS=ZHL(JLAST)*(XSTART-XEND)/(XJLALF-XJLALF)
M=JMAT(L)
DS=ZSL(JLAST)
DJ(U(JLAST))
DUDL=U(JLAST+1)-UJ

10/8/79
SUBROUTINE REZONE (Continued)

DXOLD=XJLALF1-XJLALF
XS1=0.5*(XSTART+XEND)-XNAVALF
U1=UJ+DU*D*(XEND-XJLALF)/DXOLD
U2=UJ+DU*D*(XSTART-XJLALF)/DXOLD
AMAVG=0.25*DMASS*(U1+U2)+AMAVG
AMSLP=DMASS/DXALF*(1.5*(U2-U1)*XS1+0.25*(U2-U1)*XS2)+AMSLP
ENGY=ENGY+DMASS*EHL(JLAST)
IF (TIME GT SSTOPM) GO TO 620
DO 6201 INS=1,NSPEC
IF (JF=JFIN*(INS-1)+JLAST) GO TO 6201
6201 SSS(INS)=SSS(INS)+DMASS*SS(JF)
620 CONTINUE

RS=RS+DMASS*RHL(JLAST)
PS=PS+DMASS*PHL(JLAST)
SX= SX+DMASS*SHL(JLAST)
YS=YS+DMASS*YHL(JLAST)
CS=CS+DMASS*CHL(JLAST)
ANEMS=ANEMS+DMASS*NEM(JLAST)
ANETS=ANETS+DMASS*NET(JLAST)
LL=LVAR(JLAST)
IF (LL.EQ.0) GO TO 6205
ASUM=ASUM+DMASS*COM(LL+2)
RSUM=RSUM+DMASS*COM(LL)
RVSUM=RVSUM+DMASS*COM(LL+3)
ENSUM=ENSUM+DMASS*COM(LL+4)
6205 CONTINUE
HC(J,2)=MIN0(HC(J,2),H(JLAST,2))
XSTART=XEND
IF ((H(JLAST,1).EQ.5R OR H(JLAST,1).EQ.5R) .OR. H(JLAST,1).EQ.5R)
1 (JLAST,1).EQ.5R T .AND. HC(JLAST,1).EQ.5R) GO TO 621
HC(JLAST,1).EQ.5R R .AND. HC(JLAST,1).NE.5R ZHC(JLAST,1)=5R Z
IF (HC(JLAST,1).EQ.5R) HC(JLAST,1)=5R Z
HC(JLAST,1)=MAX0(H(JLAST,1),HC(JLAST,3))
621 IF (XEND1.LE.XN) GO TO 625
JLAST=JLAST-1 $ GO TO 615
625 XC(I)=XN*S DC(I)=MASS(I+1)/DXALF $ EC(I)=ENGY/MASS(I+1)
YC(I)=YS/MASS(I+1) S SC(I)=SX/MASS(I+1) $ PC(I)=PS/MASS(I+1)
CC(I)=CS/MASS(I+1) S RC(I)=RS/MASS(I+1)
ASC(I)=ASUM/MASS(I+1) $ PSC(I)=PSUM/MASS(I+1)
RSC(I)=RSUM/MASS(I+1) $ RVSC(I)=RVSUM/MASS(I+1)
ENSC(I)=ENSUM/MASS(I+1)
IF (TIME .GT. SSTOPM) GO TO 620
DO 628 INS=1,NSPEC
628 SSSC(I,INS)=SSS(INS)/MASS(I+1)
630 CONTINUE
MOM(I,2)=AMAVG+AMSLP
MOM(I+1,1)=AMAVG+AMSLP
ANEM(I)=ANEMS/MASS(I+1) $ ANET(I)=ANETS/MASS(I+1)
643 K=JNEW+1-1
LOC=643
WRITE (6,5003) LOC,K,XC(I),DC(I),MOM(I,2),MOM(I+1,1),EC(I),RC(I),
PC(I),SC(I),Y(I),MASS(I+1),HC(I,1)
650 XN=AMAX1(XN-DX,X(JOLD))
T(JNEW+1)=TLAST
DO 6550 NJD=1,NJEDIT
IF (JEDIT(NJD).GT. JOLDR .OR. JEDIT(NJD).LT. JOLDR) GO TO 6550
JEDIT=NJD
NEWJED=NJD+1-NCEL
XJED=0.5*(X(JED)+X(JED+1))
DO 6545 I=2,NCEL
IF (XJED .LT. 0.5*(X(C(I))+X(C(I-1)))) GO TO 6545
6545 NEWJED=NJED+2-1
GO TO 6550
6545 CONTINUE
6550 CONTINUE
DO 670 J=1,NCEL
J=JNEW+1-1 $ CHL(J)=CC(I) $ DHL(J)=DC(I)
EHL(J)=EC(I) $ PHL(J)=PC(I)
YHL(J)=YC(I) $ ZHL(J)=SC(I)
NET(J)=ANET(I) $ NEM(J)=ANEM(I) $ RHL(J)=RC(I)
JL=JLDR+1-1
LVAR(J)=LVAR(JL)
6550 CONTINUE
SUBROUTINE REZONE (Continued)

LL=LVAR(J)
IF (LVAR(J) .EQ. 0) GO TO 6555
COM(LL)=RSC(I)
COM(LL+1)=PSC(I)
COM(LL+2)=ASC(I)
COM(LL+3)=RVSC(I)
COM(LL+4)=ENSC(I)

6555 CONTINUE
IF (TIME .GT. SSTOMP) GO TO 660
DO 655 INS=1,NSPEC
JF=FIN(INS)+J
C0M(LL)=RSC(I)
C0M(LL+l)=PSC(I)
C0M(LL+2)=ASC(I)
C0M(LL+3)=RVSC(I)
C0M(LL+4)=ENSC(I)
CONTINUE
IF (TIME .GT. SSTOPM) GO TO 660
DO 655 INS=1,NSPEC
JF=FIN(INS)+J
SS(JF)=SSC(I,INS)
CONTINUE

U(J+1)=2.«(MOM(I,1)+MOM(I,2))/(MASS(I)+MASS(1+1 ))
TUJsTENSCM, 1 )     
X(J)=XC(I)
H( J, 2) =HC( I , 2)
H(J,3)=HC(I,3)
CONTINUE

M0MLAST = M0M(NCEL+1 , 1)
MASLAST = MASS(NCEL+1 )
TLAST=T(J_OLD)
GO TO (680,680,700,700,685) NPART
CONTINUE
T(J)=TLAST
U(J)=2.«M0MLAST/MASLAST
M0MLAST=MASLAST=RSLAST=0.
CONTINUE

L0C=700
WRITE (6,5000) LOC,J_OLD,J_NEW,J_LAST,L,N,J,NOEL,N_PART
SET J_NEW AND J_LAST IN PREPARATION FOR THE NEXT ZONE CALCULATIONS
J_NEW=J-1  $  J_LAST=J_OLD
RETURN TO APPROPRIATE PART OF REZONE FOR
FINAL RESETTING
GO TO (100,200,300,400) N_PART

RENUMBER CELLS WITHOUT REZONING
750 T(J_NEW+1)=TLAST
LOC=750
TLAST=T(J_OLD)
WRITE ( 6,5750)

752 JLAST=JLAST-1  $  DHL(J_NEW)=DHL(JLAST)
EHL(J_NEW)=EHL(JLAST) $  PHL(J_NEW)=PHL(JLAST)
IF (TIME .GT. SSTOMP) GO TO 754
DO 753 INS=1,NSPEC
JF=FIN(INS)+J
SS(JF)=SSC(I,INS)
CONTINUE

U(J_NEW+1)=2.*M0MLAST/MASLAST
X(J)=XC(I)
H( J, 2) =HC( I , 2)
H(J,3)=HC(I,3)
CONTINUE

M0MLAST = M0M(NCEL+1 , 1)
MASLAST = MASS(NCEL+1 )
TLAST=T(J_OLD)
GO TO (680,680,700,700,685) N_PART
CONTINUE
T(J)=TLAST
U(J)=2.«M0MLAST/MASLAST
M0MLAST=MASLAST=RSLAST=0.
CONTINUE

L0C=760
WRITE (6,5000) LOC,J_OLD,J_NEW,J_LAST,L,N,J,NOEL,N_PART
GO TO (760,760,300,400,765) N_PART
CONTINUE
U(J_NEW+1)=2.*M0MLAST/MASLAST
T(J_NEW+1)=T(J_OLD)
M0MLAST=MASLAST=RSLAST=0.
LOC=760
WRITE (6,5000) LOC,J_OLD,J_NEW,J_LAST,L,N,J,NOEL,N_PART
SUBROUTINE REZONE (Concluded)
GO TO (100,200,300,400) NPART
C  ERROR MESSAGE
C 790 WRITE (6,1000) NPART, JOLD, JNEW, JLAST, NJ, JEDIT(NJ), L, JBND(L)
    CALL EDIT $ LSUB(M)=1 $ CALL STORR $ CALL SCRIBE $ STOP
C  ENDING ROUTINE - INTERFACE AND BOUNDARY ADJUSTMENTS
800 JINIT=JNEW+1
    IF (H(JINIT,2) .EQ. SPRALL) R(1)=0.
    DO 820 L=1,NLAYER
        JB=JBND(L)  $  H(JB,2)=LINTER
        IF (H(JB+1,2) .EQ. SPRALL) GO TO 820
        U(JB+1) = U(JB) = (U(JB)*ZHL(JB-1) + U(JB+1)*ZHL(JB+1))/(ZHL(JB-1) + 
                      1 ZHL(JB+1))
    CONTINUE
820 CONTINUE
    WRITE (6,5825) JINIT, DUR
    WRITE (6,5826) (JEDIT(NJ),NJ=1,NJEDIT)
    WRITE (6,5827) (NEWJED(NJ),NJ=1,NJEDIT)
    DO 825 I=1,NJEDIT
        JEDIT(I) = NEWJED(I)
        CALL EDIT
    CONTINUE
    CALL SECOND(TWIX)  $  DUR=TWIX-XNOW
    WRITE (6,5010) JINIT, DUR
    RETURN
FORMAT (24H ERROR IN REZONE, NPART=I3,6H J0LD=I3,6H JNEW=I3,7H JLAST=I3, NJ-I3,11H JEDIT(NJ) = I 3,3H L=I3,9H JBND(L) = I3)
FORMAT (13H REZONE, LOC=I3,7H, J0LD=I3,7H, JNEW=I3,8H, JLAST=I3, NJ=I3,7H, NCEL=I3,8H, NPART=I3)
FORMAT (13H REZONE, LOC=I3,7H, NCEL=I3,15H, XC START, DX, XN=1P3E10.3,9H, RSLAST=1PE10.3,10H, MASLAST=1PE10.3,1 OH, MOMLAST=1PE10.3)
FORMAT(2I5, 1 PI0E10.3,3X, R1 )
FORMAT(19H0END REZONE, JINIT=I 3,17H, TIME IN REZONE=1PE10.3 SEC
       (+ TRANSFORMATION OF JEDIT VALUES*)
       (* OLD JEDIT = *18I5)
       (* NEW JEDIT = *18I5)
END
SUBROUTINE SCATTO

SUBROUTINE SCATTO (XS, ES, ECAL, NPDINT, NS, L, ESUM) 

CALLED BY OEDS TO DISTRIBUTE ABSORBED ENERGY INTO PUFF CELLS. THE ENERGY HAS BEEN PREVIOUSLY COMPUTED BY A FLUORESCENCE AND SCATTERING CODE (SUCH AS FSCATT OF S.S.S.) OR DETERMINED EXPERIMENTALLY. DEPOSITION COORDINATES MAY BE SPACED ARBITRARILY. INTERPOLATION FUNCTION IS A PARABOLA IN LOG(E) VS X FITTED THROUGH 3 DEPOSITION COORDINATES.

INPUT - FORMAL PARAMETERS WHICH CONTAIN COORDINATES (XS) AT WHICH DOSE (ES) IS KNOWN. NPOINT (NUMBER OF COORDINATE POINTS), ECAL, AND NS (SPECTRUM NUMBER). ES SHOULD BE IN CAL/CM², XS IN CM.

OUTPUT - FILLS SS ARRAY.

INTEGER H, PDRDUUS, PRESS, RINTER, SOLIO, SPALL
REAL MATL, NEM, NET, NEMH, NETH

MISCELLANEOUS

COMMON AZERO(1), CEF, CKS, DAVG, DELTIM, OISCPT(10), DDLD, ORHO, DMAX,
1 OTIM, OTN, DTHN, OUX, DX, EOLD, PF, FAC, FIRST, J, JCYCS, INIT
2 JFIN, JREZDN(15), JSMAX, JSTAR, JTS, LSUB(30), MMAXPR(30), NNCYCS,
3 NEDIT, NERR, RHEZON, NSCRB(6), NSEPRAT, NSPALL, NTEUT

OUTPUT - FILLS SS ARRAY.

INTEGER H, PDRDUUS, PRESS, RINTER, SOLIO, SPALL
REAL MATL, NEM, NET, NEMH, NETH

MISCELLANEOUS

COMMON AZERO(1), CEF, CKS, DAVG, DELTIM, OISCPT(10), DDLD, ORHO, DMAX,
1 OTIM, OTN, DTHN, OUX, DX, EOLD, PF, FAC, FIRST, J, JCYCS, INIT
2 JFIN, JREZDN(15), JSMAX, JSTAR, JTS, LSUB(30), MMAXPR(30), NNCYCS,
3 NEDIT, NERR, RHEZON, NSCRB(6), NSEPRAT, NSPALL, NTEUT

BEGIN LOOP FOR EACH MATERIAL

IF (L .GT. 1) J=JBND(L-1)*1
XEND=X(J)
JBNDM=JBND(L)
XTH=X(JBNDM)

JS2=JS1*2

ESL=ALOG(AMAX1(ES(JS1),1.E-10))

502
SUBROUTINE SCATTO (Concluded)

ES2 = ALOG (AMAX1 (ES (JS2) + 1 * E-10))
ES3 = ALOG (AMAX1 (ES (JS3) + 1 * E-10))
X1 = XS (JS1) $ X2 = XS (JS2) $ DX12 = X2 - X1 $ X3 = XS (JS3)
DX13 = X3 - X1 $ DX23 = X3 - X2

494 R12 = (XEND - X3) / DX12 $ R23 = (XEND - X1) / DX23 $ R31 = -(XEND - X2) / DX13
ESS1 = EXP (-ES1 * R12 * R31 - ES2 * R23 * R12 - ES3 * R31 * R23)

495 XBEGIN = XEND
XEND = AMIN1 (XSTOP, X (J+1))
R12 = (XEND - X3) / DX12 $ R23 = (XEND - X1) / DX23 $ R31 = -(XEND - X2) / DX13
XBEGIN = (XEND - XBEGIN) / 6. * (ESS1 + 4 * ESS2 + ESS3) + ESS
ESS1 = ESS3

IF (ABS (XEND - X (J+1)) < LT. 1 * E-10) GO TO 496

C PREPARE FOR NEXT PUFF CELL
IF (JS3 .LT. NPOINT) JS1 = JS1 + 1
GO TO 492

496 DX = X (J+1) - X (J)
SS (JFINNS + J) = ESS * FACTOR / DX
ESSUM = ESSUM + ESS

C PREPARE FOR NEW SET OF THREE XS COORDINATE POINTS
IF (XEND .GT. XS (NPOINT) AND. ESS (NPOINT) .LE. .01) GOTO 500
ESS = 0.
J = J + 1
IF (XEND .LT. XTH + 1 * E-10) GO TO 495

500 M = JMAT (L)
RETURN
END
SUBROUTINE SCRIBE

INTEGER HPOROUS, PRESS, RINTER, SOLIO, SPALL
REAL MATL, NEM, NET, NEMH, NETH

C
C            MISCELLANEOUS
C
COMMON AZERO(1), CEF, CKS, OAVG, OELTIM, OISCPT(10), DOI0, URHO, DTMAX,
1 OTIN, DTN, OTHN, DU, DX, EULO, F, FAC, FIRST, J, JCYS, JINIT,
2 JFIN, JREZON, IS MAX, JSTAR, JTS, LSUB(30), M, MAXPR(30), N, NCYCS,
3 NEDT, NPER, NR, NREZON, NSCRB, (6), NSEPRA, NSPALL, NTEDT,
4 NTEX, NTR(15), PLO, P6, (20), R, (30), RLAST, SLAST, SMAX, TEOIT(50),
5 TF, TIME, TI, TREZON, TS, T6(20), ULAST, UOLO, UZERO, XLAST, XNOW, XOLO
1 JJOI(20)

C              HALFSTEP VALUES
C
COMMON DH, DHLAST, DUH, EH, PH, RH, RHLAST, SH, SHLAST, UH, UHLAST, XH, XHLAST

C
C            CONDITION
C
COMMON INF, LINTER, MIRROR, NORMAL, POROUS, PRESS, RINTER, SOLIO, SPALL

C
C            CELL LAYOUT
C
COMMON DXX(30), JBNO(30), JMAT(30), NAUTO, MATL(6, 2), NTHK(30)

C
C            COORDINATE ARRAYS
C
COMMON/COORD/X(200), X0(200), CHL(200), OML(200), OPO0(200), OPOE(200), COORDC02
1 EHL(200), H(200), 3, NEM(200), NET(200), PHL(200), RHH(200), SOT(200), COORDC03
2 SHL(200), T(200), U(200), YHL(200), ZHL(200)

COMMON/NSC/A(5000)
COMMON/NGU/JOED/JOEIT(100), JNUM(100), JTYP(100), NAMEZ(40), JEDSZ, JEDCM
1 MODULUS, NERR, NJEOIT, NTape

DIMENSION JV(13)
NTAPE = 3
REWIND 7
WRITE (7) N
CALL SECO(XSTART)
IF (NERR .GT. 0) PRINT 1083, NERR
NSC = (NJEDIT + 2) / 10 + 1
NBUF = (N - 1) / MODULUS
NPERP = 50 / MODULUS

C
C            BEGIN DO LOOP OVER EACH SCRIBE LISTING
C
DO 900 NS = 1, NSC
LENGTH = MODULUS
IPAG = 1
C
IF (UNIT(NTAPE)) 650, 990, 640
PRINT 1082
REWIND NTAPE
C
BUFF IN FIRST RECORD OF TAPE
IF (UNIT(NTAPE)) 655, 990, 652
PRINT 1082
C
BUFF IN (NTAPE+1) (A1), A1
IF (UNIT(NTAPE)) 655, 990, 652
PRINT 1082
C
BUFF IN (NTAPE+1) (A1), A(JEDSZ*MODULUS)
IF (NS .GT. 1) GO TO 680
JENO = MINO(12, NJEOIT+5)
JBEG = 3
JV(1) = 10H, OPF6.N, 10
JV(2) = 10H, F10.3
JV(3) = 10H, F10.3
JV(4) = 10H, F10.3
JV(5) = 10H, F10.3
JV(13) = 10H
00 670 I = 1, 7
JJ = I + 5
JV(JJ) = 10H, 1PE11.3
C
IF (JNUM(I) .GE. 1400 .AND. JNUM(I) .LT. 2000) JV(JJ) = 9H, 10X, A1
IF (1 .GT. NJEOIT) JV(JJ) = 1H
C
670 CONTINUE
GO TO 695
C
680 JBEG = JENO + 1
SUBROUTINE SCRIBE (Concluded)

JEND=MIN(JBE6+9,NJEDIT+5)
JD=JBE6+9
DO 690 1=JBE6,JD
JJ=I-JBE6+3
JE=1-5
JJ=I0H,IPE11.3
IF (JNUM(JE)=GE.1400 .AND. JNUM(JE)=LT.2000) 
1 1V(JJ)=9H,10X,A1
IF (I·GT. NJEDIT+5) JV(JJ)=1H
CONTINUE
690 695
DO 850 NB=1,NBUF
700 IF (UNIT(NTAPE)=710*990) 700
PRINT 1082
710 IF (NB=NBUF) GO TO 750
IF (MOD(N,MODLUS)=EQ.0) GO TO 770
LENGTH=MOD(N,MODLUS)
GO TO 770
C BUFFER IN RECORDS
750 JBUF=MOD(NB+2)+2500
BUFFER IN (NTAPE.N)(A(JBUF+1),A(JBUF+2500))
770 CONTINUE
J1=MOD(NB-1,2)+2500
J2=LENGTH+J1
J3=JBE6+5
JD=JEND-5
IF (IPAG =EQ.1 .AND. NS .EQ. 1) PRINT 1200+DISCPT,NS,(JTYPER(I),
1 I=JB.JD)
IF (IPAG =EQ.1 .AND. NS .EQ. 1) PRINT 1100+DISCPT,NS,(JTYPER(I),
1 I=1.JD)
PRINT JV,((A(I+J),I=1,2),(A(I+J),I=JBE6,JEND),J=J1,J2,JEDSIZ)
IPAG=MOD(IPAG,NPERP)*1
CONTINUE
850 CONTINUE
REWIND NTAPE
900 CONTINUE
CALL SECOND(XEND)
DUR=XEND-XSTART
DUR2=XEND-FIRST
PRINT 1900+DUR+DUR2
RETURN
990 RETURN
RETURN
1082 FORMAT (32H PARITY ERROR ON NTAPE IN SCRIBE)
1083 FORMAT (* EOFs AND PARITY ERRORS ON TAPE 3: NERR =*I3)
1084 FORMAT (9H EOF FOUND ON NTAPE IN SCRIBE)
1100 FORMAT (1H1+10A10/* SCRIBE NO. */I2, */ USUAL UNITS ARE DYN, CM, SEC)
1 GRAM, EXCEPT TIME IN MICROSEC, DTNH IN NANOSEC*/
2 5X,*N=6X,*TIME=5X,*DTNH=4X,*DELTIM=3X,*JTS*,7(1X,A0)
1200 FORMAT (1H1+10A10/* SCRIBE NO. */I2, */ USUAL UNITS ARE DYN, CM, SEC)
2 GRAM/*
1900 FORMAT (17H0TIME IN SCRIBE = F10.3/17H COMPUTING TIME = F10.3)
END

305
SUBROUTINE SHEAR2(INCALL, IM, K, IH3, SX, SY, SZ, SX, SY, P, TAUR, OH, UOLD, IDTO, EH, EOLD, EN, EMELT, EP, EX, EY, EZ, EXY, FYL, PLEN, ROT, OHOT, ESC, CN)

C ROUTINE FOR COMPUTATION OF STRESSES WITH RATE-DEPENDENT YIELD MODEL FOR DEVIATORS AND MIE-GRUNEISEN FOR PRESSURE.
C IF THRESHOLD PLASTIC STRAIN IS REACHED, SHEAR BANDS ARE NUCLEATED AND GROWN IN 6 ORIENTATIONS.
C SX, SY, SZ ARE DEVIATORS IN EXTERNAL SIGN CONVENTION, P IS POSITIVE IN COMPRESSION.  INTERNAL SIGN CONVENTION IS POSITIVE IN COMPRESSION FOR ALL STRESS AND STRAIN QUANTITIES.
C ST IS TOTAL STRESS AT PREVIOUS TIME.  SE IS NEW DEVIATOR.
C EX, EY, EZ, EXY ARE STRAIN INCREMENTS IN EXTERNAL SIGN.
C SS, SSE CHANGE EXTERNAL SIGN CONVENTION TO INTERNAL FOR STRESS AND STRAIN, RESPECTIVELY.
C
DIMENSION BFR(6,35), NSIZE(30,9), FNUC(9), TAUZ(6), EFR(3), VFR(6), CN(100), DEP(4), CLA(IOO), ICN(IOO)

C STRESS IS NEG IN TENSION
DATA SS, SSE/-1., -1./
NC1=NCALL*1
60 TO (10,10,I00.1U0,900)NC1
10   READ(1N,1002) Al, A2, (BFR(M,I),1=22,35)
PRINT 1002. A1, A2, (BFR(M,I) .1=22,35)
1002 FORMAT (2A5.7E10.3/10X.7E10.3)
READ (IM.1003) Al, A2, (NSIZE(M,I),1=1,9)
PRINT 1003»A1,A2»(NSIZE(M,I),1=1,9)
1003 FORMAT(2A5.1415)
VMAX(M)=0.
NSIZT(M)=NSIZE(M,1)
DO 14 I=2,9
14 NSIZT(M)=NSIZT(M)+NSIZE(M,I)
VFR(M)=1.
IF (NCALL .EQ. 1) GO TO 65
NANG=BFR(M,32)
KLAST=0
DO 16 1=1,3
FNUC(I)=.111111
16   FNUC(I*3)=.222222
IF(NANG < 6) 20.40»30
20   FNUC(I)=.333333
IF (NANG .GE. 4) FNUC(I) =.25
DO 25 I=1,4
FNUC(I)=FNUC(I)
25   FNUC(I+3)=.666667
IF (NANG .EQ. 2) FNUC(2) =.666667
IF (NANG .NE. 5) GO TO 30
FNUC(4)=.125
FNUC(7)=.125
FNUC(5)=0.
FNUC(6)=0.
GO TO 40
30 DO 35 I=7,NANG
FNUC(I)=.111111
35 FNUC(I+3)=.111111
40 CONTINUE
45 RETURN
C************
C COMPUTE STRESS AND DAMAGE
C************
100 IF (IH3 .GE. 25) GO TO 600
IF (VMAX(M) .EQ. 0.) VMAX(M)=SQRT(ESC(M,5)/ESC(M,1))
C COMPUTE STRESS REDUCTION FACTORS TAUZ(I)
TAU=0.
JN=0
DO 110 NG=1,NANG
TAUZ(NG) = 0.
IF (NSIZE(M*NG) .EQ. 0 .OR. CN(JN+1) .EQ. 0.) GO TO 110
NSIZE = NSIZE(M*NG)
DO 120 I = 1, NSIZE
JN = JN + 2
120 TAUZ(NG) = TAUZ(NG) + CN(JNN) * CN(JNN+1) * 3
TAU = TAU * TAUZ(NG)
JN = JN + 2
END
C INITIAL TRANSFORMATION
C ADJUST SIGNS, ROTATE STRESS, TRANSFORM TO STRESS IN SOLID(ST)
RT = ROT * ROT + DROT
EMU = DOLD / ESC(M,1) - 1.
PH = EMU * (ESC(M,2) + EMU * (ESC(M,3) + EMU * ESC(M,4)))
PS = PH * (1. - ESC(M,9)) * DOLD * ESC(M,9) * EOLD
IF (PS .GT. 0.) P = P - PS * TAU
SA = (SX + SY) / 2. * COR + SX * SY * SOR * SS
SOL = SQRT(1.5 * (SE(1) * SE(2) / 2) + SE(4) * SE(4) / 2.2)
DO 160 I = 1, 6
SE(I) = ST(I) + G2 * (ES(I) - EBAR) / AMAX1(0.02, 1 + BFR(M,33) / SN)
SE(4) = ST(4) * G2 * (1.5 * TAUZ(4)) / AMAX1(0.02, 1 + BFR(M,33) / SN)
SN = SQRT(1.5 * (SE(1) * SE(2) / 2) + SE(4) * SE(4) / 2.2)
DHN = DOLD / ESC(M,1) - 1.
EMU = DHN / ESC(M,1) - 1.
PH = EMU * (ESC(M,2) + EMU * (ESC(M,3) + EMU * ESC(M,4)))
PE = PH * (1. - ESC(M,9)) * EMU / 2.
Y1 = AMAX1(0.02, 1 + BFR(M,33) / SN)
IF (SN .LT. Y1) GO TO 500
C YIELD AND PLASTIC STRAIN CALCULATIONS
C EXPT = EXP(-DT0 / BFR(M,30) / NSTEP)
YEG = (Y1 + BFR(M,31) * SN / 2 / G2) / (1. + BFR(M,31) / 2 / G2) / SN
DO 180 I = 1, 6
SE(I) = ST(I) * EXP + (YEG * SE(I) + BFR(M,30) * SE(I) - ST(I) + P) / (1. + EXP)
SE(4) = ST(4) * EXP + (YEG * SE(4) + BFR(M,30) * SE(4) - ST(4)) / UTO * NSTEP)
1 (1. + EXP)
DO 200 I = 1, 6
SUBROUTINE SHEAR2 (Continued)

200  DEP(I) = ES(I) - EBAH/NSTP - (SE(I) - ST(I)^P) / G2
    DEP(4) = ES(4) - (SF(4) - ST(4)) / G2
    DGamma = SQRT(1.5*(DEP(1)^2 + DEP(2)^2 + DEP(3)^2) + 0.75*DEP(4)
    1^2)
    YH = YHL + BFR(M, 31)*DGamma
    DPLENR = (SE(1) - ST(1)^P)*DEP(I) + (SE(2) - ST(2)^P)*DEP(2) + (SE(3) + S
    IT(3) - P)*DEP(3) + (SE(4) - ST(4)^P)*DEP(4) / DGamma^*AMAX1(0, 1 - TAU)
    DPLENR = ABS(DPLENR)
    YHL = YHL*BFR(M, 31)*DGamma
    TML = ThL*b|-K(M, t)"UljAMMA
    DPLENR = ((SE(I)*ST(I) - P)*DEP(I)*ISE<2) * ST(2) - P)*DEP(2)*I
    (3) - P)*DEP(3)*(SE(4)*ST(4)) * DEP(4) / 2./0HN"1
    DPLENR = ABS(DPLENR)

C COMPUTE PLASTIC STRAIN IN EACH ORIENTATION

STR1 = ABS( SE(4) )
STR2 = ABS( SE(4) )
STR3 = 0.
STR4 = ABS( SE(1) - SE(2) ) / 2.
STR5 = SQRT( (SE(1) - SE(3))^2 + SE(4)^2 ) / 2.
STR6 = SQRT( (SE(2) - SE(3))^2 + SE(4)^2 ) / 2.
SN = SQRT( 1.5*(SE(1)^2 + SE(2)^2 + SE(3)^2 + SE(4)^2) )

TEP(1) = DGamma / SN * STR1
TEP(2) = DGamma / SN * STR2
TEP(3) = DGamma / SN * STR3
TEP(4) = DGamma / SN * STR4
TEP(5) = DGamma / SN * STR5
TEP(6) = DGamma / SN * STR6

C GROWTH PROCESS

NTOT = 2*NSTP(M)
DO 250 I = 1, NTOT
CNA(I) = CN(I)
IF (EN .EQ. 0) GO TO 360
JN = 0
DC = VMAX(M) * DTO / NSTP
DO 350 NG = 1, NANG
DGAMM = 0.
DO 300 I = 1, NSTFM
JN2 = JN * 2 * (NSIZE(M, NG) + 1)
CLA(JN2) = AMIN1(CN(JN2) * EXP(CN(JN2) * DC)
            CLA(JN2) = AMIN1(CN(JN2) * EXP(CN(JN2) * DC)
    300  DGamma = DGamma + CN(JN2 - 1) * EXP(M, 27) * (CLA(JN2)^3 * CN(JN2)^3)
    IF (DGamma .LE. TEP(NG)) GO TO 345
    RR = TEP(NG) / DGamma
    DCR = DC * RR
    EXP = EXP * RR
    NSIZE = NSIZE(M, NG)
    DO 340 I = 1, NSTFM
    JN2 = JN + 2 * (NSIZE(M, NG) - 1)
    340 CLA(JN2) = AMIN1(CN(JN2) * EXP(CN(JN2) + DGamma)
    350 CN(JN) = JN + NSIZE(M, NG)^2
    360 CONTINUE
    DO 365 NG = 1, NANG
    CN(TOT + NG) = CN(TOT + NG) + TEP(NG)

C NUCLEATION PROCESS

TEPM = 0
DO 370 NG = 1, NANG
    TEP = AMAX1(CN(TOT + NG))
    IF (TEPM .LT. BFR(M, 26)) GO TO 500
    JN = 0
    DO 450 NG = 1, NANG
    IF (NSIZE(M, NG) .EQ. 0) GO TO 450

C
SUBROUTINE SHEAR2 (Continued)

IF (CN(NTOT*NG) .LT. flFR(M,2b) .OH. Tt.P(NG) .LT. l.E-5)           SHRtAR^Ob
1 GO TO 50
SriEAK^07
DNO=TEP(NG)»BFH(M,25)»FNUC(NG)•(DPL£NR/BTO*NSTEP/BFR(M,35))««2    SHtAPdO^7
CNH = 0.
SHEAh(?09
NSIZEM=NSIZE(MiMG)
SH£AW?10
DO 4*0
I=1»NSIZEM
SHEAR?11
II=NSI2E(M,NG)*1-I
SHEAR?12
JNI=JN*2»II-1
SMfcAK^13
IF(CLA(JNI) .NE. 0.) GO TO 420
SrltAH^U
CLACJNl>aBFR(M,28)*(l»-BFR(M,29l»«llJ/(1.-BFR(M,29) <n*NSIZE(M,NO))
SNEAH^I^7
CN(JNI)=CLA(JNI)
5hEAK?16
420
CMl=DNO»EXP(-(CLA(JNI) ♦
CN(JNI))/2./HFR(M,24))
SHEAR^17
JNN = JN
II+2
II-1
SHEAR218
CNA (JNN)=CNL-CNP*CN(JNN)
SHtiAK?!^7
440 CNR=CNL
SHEAR220
EN=EN*CNL
SMtAR221
450 JN»NS1ZE<MIN6)«2*JN
SHt'AR^22
470   CONTINUE
SriLAH223
Co*********
5Ht;AR?24
C         COMPUTE TAU AND REFILL MAIN
ARRAYS
SHEAR^26
C*»*««*««*««
TAU = 0.
SHEAR^27
JN=0
SHEAH^2n
IF(EN .EQ. 0.)  GO TO 500
SHEAH?2^
DO 490
NG=1.NANG
SHEAH
c
'3n
TAUZ(NG)=0.
SHEAR^Sl
IF (NSIZE(M.NG) .E(J. Q) GO TO 490
SHEA6C
C
'32
IF ICNA(JN*l) .F.Q. 0.) GO TO 490
SHtAH^33
NSI7EM
= NSIZE (MtNG)
SHEAR^5*
DO 480
I=1.NSIZEM
SHEA^.^b
JNN = JN*2
II-1
SHEA^.^b
CN(JNN*l)=CLA(JNN*l)
SHEARr37
CN(JNN)3CNA(JNN)
SHEA«?3H
480  TAUZ(NG)=TAUZ(NG) ♦
CLA(JNN*1)»»3
SHEAR^39
TAU = TAU*TAUZMG)
SnEAR240
490 JN=JN*NSIZE(M,NG)»2
SHEAfi241
IF(TAU»VFR(M) .GE. 1.) GO TO 800
SHEAR<
3
42
500  CONTINUE
SHEAR^44
P=(ST{1)
♦
ST{2)*ST(3))/3.
SHEAR245
SA=(ST(1)*ST(2))/2.
SHEAR.-47
bOO CONTINUE
SHEAR248
C**********
SHEAR249
C     TRANSFORMATION TO GLOBAL ORIENTATION
SHEAR250
C**««»*»«*«
IF(J .EQ. 17 .AND. TAU .GT.  .05) PRINT 1601.ST,P
1601 FORMATUH ST= lP4E10.3.4H PS= E10.3)
ST(4)=ST(4)*AMAX1(0..(1.-3.*TAUZ(1)*TAUZ(2))»3.*TAUZ(4) »VFR(M)SHEAR254
1 ))
ST(1)=ST(1)*AMAX1(0..(1.-3.*TAUZ(1)*1.5*(TAUZ(4)*TAUZ(5)))*VFR(M)SHEAR255
1 ))
ST(2)=ST(2)*AMAX1(0..(1.-3.*TAUZ(2)*1.5*(TAUZ(4)*TAUZ(5)*TAUZ(6)))»VFR(M)SHEAR256
1 ))
ST(3)=ST(3)*AMAX1(0..(1.-3.*TAUZ(3)*1.5*(TAUZ(5)*TAUZ(6))))»VFR(M)SHEAR257
1 ))
P=(ST(1)*ST(2)+ST(3))/3.
SHEAR258
SA=(ST(1)*ST(2))/2.
SHEAR259
SB=(ST(1)+ST(2))/2.*COR=ST(4)*SOR
SXY=(ST(1)+ST(2))/2.*SOR=ST(4)*COR#SS
SX=(SA+SB+P)*SS
SY=(SA-SB-P)*SS
SZ=(ST(3)-P)*SS
IF (PE .GT. 0.) P=P*PE*TAU
IHA3=20.*TAU*2.9
IF (J .EQ. 17 .AND. TAU .GT. .05) PRINT 1400+SX+SY+SZ+P*ST.EX.EY.DMSHEAR271
IF (NCALL .GE. 3) GO TO 900
RETURN

309
SUBROUTINE SIGMAT

FUNCTION SIGMAT(LS,T)
DIMENSION PS(10), TS(10)
DATA PS/54.4E6, 34.0E6, 40.8E6, 24.5E6, 60.0E6/
DATA TS/0.6E-4, 8.4E-4, 3.2E-3, 3.6E0, 0.0/
DATA NM/4/
N=1
20 N=N+1
IF ( T .GT. TS(N) .AND. N .LT. NM) GO TO 20
SIGMAT=PS(N-1)*(PS(N)-PS(N-1))/(TS(N)-TS(N-1))*(T-TS(N-1))
RETURN
END
SUBROUTINE SSCALH

FUNCTION SSCALH(JS)

COMPUTES RADIANT ENERGY FOR DEPOSITION IN EACH CELL AT HALFSTEP POINT AND Initializes ENERGY IN NEW ZONES

INPUT - J(JS), NSPEC, SDFRM, TIME, DTNH, SSTOPM, DTN.

OUTPUT - SSCALH.

INTEGER H, POROUS, PRESS, RINTER, SOLID, SPALL
REAL MATL, NEM, NET, NEMH, NETH

COMMON AZERO(1), CEF, CKS, DAVG, DELTIM, DISCTP(10), OOLD, ORHO, OMAX,
1 DTMIN, DTN, DTNH, DU, DX, EDLO, F, FAC, FIRST, J, JCYCS, JINIT,
2 JFIN, JREZON(15), JSTOP(15), JTSUB(30), M, MAXP(30), MNCYCS,
3 NEDIT, NPERN, NR, NREZON, NSCRB(6), NSPERM, NSPALL, NTEDT,
4 NTEX, NTR(15), POLD, P6(20), R(30), RLAST, SLAST, SMAX, TEDIT(50),
5 TF, TIME, TJ, Trezon, TS(16), ULAST, ULOL, UZERD, XLAST, XNOW, XLOL
1 XJDIIT(20)

COMMON DH, DHLAST, DUH, EH, PH, RH, RHLAST, SH, SHLAST, UH, UHLAST, XH, XHLAST

COMMON Azero(l), CEF, CKS, DAVG, DELTIM, DISCTP(10), OOLD, ORHO, OMAX,
1 DTMIN, DTN, DTNH, DU, DX, EDLO, F, FAC, FIRST, J, JCYCS, JINIT,
2 JFIN, JREZON(15), JSTOP(15), JTSUB(30), M, MAXP(30), MNCYCS,
3 NEDIT, NPERN, NR, NREZON, NSCRB(6), NSPERM, NSPALL, NTEDT,
4 NTEX, NTR(15), POLD, P6(20), R(30), RLAST, SLAST, SMAX, TEDIT(50),
5 TF, TIME, TJ, Trezon, TS(16), ULAST, ULOL, UZERD, XLAST, XNOW, XLOL
1 XJDIIT(20)

IF (NSPEC .LE. 0) RETURN

IF (JS .GT. JSS) GO TO 50

IF (SDFRM .EQ. 1.) RETURN

IF (TIME-0TNH-.5«DTN .GT. SSTOPM) RETURN

DO 48 I=1,NSPEC

JFINNS=JFIN«(I-l)

IF ((TIME-.5»DTNH-.5»DTN-SSTOP(I)) .LT. (START(I))) 46,48,48

SSCALH=SSCALH+SS(JFINNS*JS)«AMAX1(SSTOP(I),TIME-.5»DTNH)-
1 AMIN1(START(I),TIME-DTNH-.5»DTN)

48 CONTINUE

RETURN

C ENERGY ADDITION FOR NEW ZONES

DO 60 I=1,NSPEC

JFINNS=JFIN«(I-1)

IF ((TIME-.5»DTNH-.LT. START(I)) GO TO 60

SSCALH=SSCALH+SS(JFINNS*JS)«AMAX1(SSTOP(I),TIME-.5»DTNH)-
1 START(I))

60 CONTINUE

RETURN

END
SUBROUTINE STORR

INTEGER H, POROUS, PRESS, RINTER, SOLID, SPALL
REAL MATL, NEM, NET, NEMH, NETH

COMMON AZEROM(1), CEF, CKS, DAVG, DELTIM, DISCPT(10), DOLD, DRHO, DTMAX,
1  DTMN, DTN, DTH, DU, DX, EOLD, F, FAC, FIRST, J, JCYCS, JINIT,
2  JFIN, JN, TREPZON(15), JSMAX, JSTAR, JTS, LSUB(30), M, MAXPR(30), N, NCYCS,
3  NEDIT, NPERN, NREZON, NSCRB(6), NSEPRAT, NSEL, NSPALL, NTEDT,
4  NTEX, NTR(15), POLD, P6(20), R(30), RLAST, SLAST, SMAX, TEDIT(50),
5  TF, TIME, TREPZON, TS, T6(20), ULAST, UOLD, ZERO, XLAST, XNOW, XORL
1  , XJDUCT(20), M, HALFSTEP VALUES
2  COMMON DH, DHLAST, DUH, E, RH, RHLAST, SH, SHLAST, UH, UHLAST, XH, XHLAST
3  , NEMH, NETH
4  CONDITION INDICATORS
5  COMMON INF, LINTER, MIRROR, NORMAL, POROUS, PRESS, RINTER, SOLID, SPALL
6  CELL LAYOUT
7  COMMON DXX(30), JBND(30), JMAT(30), NAUTO, MATL(6,2), NLAYER, NMTRLS,
8  THK(30)
9  COORDINATE ARRAYS
10  COMMON /COORD/X(200), XO(200), CHL(200), DHL(200), DPDD(200), DPDE(200),
11  EHL(200), H(200,3), NEM(200), NET(200), PHL(200), RHL(200), SDT(200),
12  COMMON /NSC/A(5000)
13  COMMON /IND/I(200), INDC(200), LALPHA, NCMF(6), NFR(6), NPGF(6),
14  NSSCOM...NSCOM
15  COMMON /JED/JEDIT(100), JNUM(100), JTYP(100), NAME2(40), JEDSZ,
16  COMMON /PE/ LMAX, LVTOT, LVAR(200), COM(4000)
17  DIMENSION RIMP(20), JINT(20)
18  DIMENSION KB(300)
19  EQUIVALENCE (A(2501), KB)
20  IF (N.GT.1) GO TO 100

INITIALIZATION

1  NTAPE=3 $ NERR=IBUF=0 $ MODLUS=50
2  IF (NJEDIT.GT.45) MODLUS=25
3  IF (NJEDIT.GT.95) MODLUS=10
4  IF (NJEDIT.GT.245) MODLUS=5
5  JEDSZ=2500/MODLUS
6  DO 30 I=1,10
7  30 A(2500+I)=DISCPT(I)
8  KB(1)=MODLUS
9  KB(2)=JEDSZ
10  KB(3)=JEDSZ
11  KB(4)=NJEDIT
12  DO 40 I=1,100
13  40 KB(14+I)=JTYP(I)
14  DO 50 I=1,20
15  50 JINT(I)=JEDIT(I)
16  JINT(1)=JEDIT(1)
17  JINT(1)=0.

BEGIN STORAGE

1  IF (LSUB(7).NE.0) GO TO 600
2  IB=JEDSZ+1BUF
3  A(IB+1)=N
4  A(IB+2)=TIME+1.6E
5  A(IB+3)=DTH+1.6E
6  A(IB+4)=DELTIM
7  A(IB+5)=JTS
8  IC=IB+5
9  IR=0
10  DO 50 JE=1,NJEDIT
11  JD=JEDIT(JE)
12  JNUMB=JNUM(JE)
13  IF (JNUMB.GE.4000) GO TO 200
14  STORAGE FOR ALL ARRAY VARIABLES
15  A(1C+JE)=X(JNUMB+JD)
16  GO TO 500
17  STORAGE FOR COM VARIABLES
18  IF (JNUMB.GE.5000) GO TO 300
19  JN=JNUMB-4000
20  L=LVAR(JD)
21  IF (IB+1+LU.F.2) GO TO 600
22  BEGIN STORAGE
23  IF (LSUB(7).NE.0) GO TO 600
24  IB=JEDSZ+1BUF
25  A(IB+1)=N
26  A(IB+2)=TIME+1.6E
27  A(IB+3)=DTH+1.6E
28  A(IB+4)=DELTIM
29  A(IB+5)=JTS
30  IC=IB+5
31  IR=0
32  DO 50 JE=1,NJEDIT
33  JD=JEDIT(JE)
34  JNUMB=JNUM(JE)
35  IF (JNUMB.GE.4000) GO TO 200
36  STORAGE FOR ALL ARRAY VARIABLES
37  A(1C+JE)=X(JNUMB+JD)
38  GO TO 500
39  STORAGE FOR COM VARIABLES
40  IF (JNUMB.GE.5000) GO TO 300
41  JN=JNUMB-4000
42  L=LVAR(JD)
43  IF (IB+1+LU.F.2) GO TO 600
A(1C+JE)=COM(L+JN)
GO TO 500
300 JB=JNUMB/200-24
    IF (JB .GT. 8) GO TO 500
    GO TO (310,320,330,340,350,360,370,380) JB
C    INTERFACE STRESS
310 JD=JINT(JE)
    A(1C+JE)=R(JD+1)
    GO TO 500
C    SECOND PRINCIPAL STRESS
320  IF (NALPHA .EQ. 2) GO TO 325
    A(1C+JE)=-0.5*SHL(JD)+1.5*PHL(JD)
    GO TO 500
325  A(1C+JE)=PHL(JD)+SDT(JD)
    GO TO 500
C    THIRD PRINCIPAL STRESS
330  IF (NALPHA .EQ. 2) GO TO 335
    A(1C+JE)=-0.5*SHL(JD)+1.5*PHL(JD)
    GO TO 500
335  A(1C+JE)=-SHL(JD)+2.*PHL(JD)-SDT(JD)
    GO TO 500
C    IMPULSE
340  IR=IR+1
    RIMP(IR)=RIMP(IR)+RHL(JD)*DTNH
    A(1C+JE)=RIMP(IR)
    GO TO 500
C    SPECIFIC VOLUME
350  IF (DHL(JD) .GT. 0.) A(1C+JE)=1./DHL(JD)
    GO TO 500
C    DEVIATOR STRESS - FIRST DIRECTION
360  A(1C+JE)=SHL(JD)-PHL(JD)
    GO TO 500
C    DEVIATOR STRESS - SECOND DIRECTION
370  IF (NALPHA .EQ. 2) GO TO 375
    A(1C+JE)=0.5*(PHL(JD)-SHL(JD)
    GO TO 500
375  A(1C+JE)=SDT(JD)
    GO TO 500
C    DEVIATOR STRESS - THIRD DIRECTION
380  IF (NALPHA .EQ. 2) GO TO 385
    A(1C+JE)=0.5*(PHL(JD)-SHL(JD))
    GO TO 500
385  A(1C+JE)=PHL(JD)-SHL(JD)-SDT(JD)
    GO TO 500
500 CONTINUE
    IBUF=IBUF+1
C    BUFFER OUT ARRAY ONTO NTAPE
    IF (IBUF .NE. MODLUS) GO TO 550
    IF (UNIT(NTAPE)) 520,510,510
510 NERR=NERR+1
520 BUFFER OUT(NTAPE,1)(A(1),A(JEDSZ*MODLUS))
    IF (LSUB(7) .NE. 0) GO TO 615
    RETURN
550 IF (IBUF .NE. 2*MODLUS) RETURN
555 IF (UNIT(NTAPE)) 570,560,560
560 NERR=NERR+1
570 BUFFER OUT(NTAPE,1)(A(JEDSZ*MODLUS+1),A(5000))
    IF (LSUB(7) .NE. 0) GO TO 615
    RETURN
600 IF (IBUF .EQ. 0) GO TO 615
IF (IBUF-MODLUS) 505,615,555
615 IF (UNIT(NTAPE)) 625,620,620
620 NERR=NERR+1
625 REWIND NTAPE
RETURN
END
SUBROUTINE STRES2

SUBROUTINE STRES2(LS, IND, I3, M, J, N, D, DOLD, RHOS, SDH, MUM, F, DTNP1, NEMSTRES)

1   NET TSR

REAL MUM,NEM,NET

COMMON /S2/ ALF, CO, EEN, EENP1, EPN, KS, TAUEL, TAUJ, TAU, TAUJ, VELS, VMU, ALCOM
1 ZAM, ZAMUSV, ZEP, ZEPDSV, ZEPMAX, ZEPSAVE, ZTAU, ZTAUJM, ALCOM

DIMENSION TAUY(300), TAUJ(300), EPMAX(300), EPSAVE(300), EP(300), TAU(300)

DIMENSION TSR(6, 30)

VALUE OF IND = 0 COMPLETE CALCULATION
1 COMPLETE CALC., EXCEPT FOR RESETING ARRAYS
2 ONLY RESET ARRAYS

IF (LS .GT. 0) GO TO 5
KS = 0
ZAM = MUM
CO = TSR(M, 15)
TAUJ = TSR(M, 16)
TAUJ = TSR(M, 17)
ALF = TSR(M, 18)
GO TO 4
I = 1
TAUJ(I) = ZTAUJM(I) = EPMAX(I) = EPMAX(I) = EPSAVE(I) = EP(I)
1 TAU(I) = 0
AMUSV(I) = ZAM
4 CONTINUE
LS = 1

IF (IND .EQ. 2) GO TO 100
ZTAUJ = ZTAUJ(J) $ ZTAUJM = ZTAUJM(J) $ ZEPMAX = EPMAX(J)
ZEP = EPSAVE(J) $ ZAM = ZAMUSV $ TAUJ = TAU(J)

*** TESTS FOR MATERIAL EXCEEDING MELT ENERGY ***

IF (ZEPMAX .LT. 0.) GO TO 90
IF (F .GT. 0.) GO TO 10
ZEPMAX = 1.
SDH = 0.
GO TO 90

*** TEST FOR INITIALIZING SHEAR STRESS CALCULATIONS ***

IF (ZEPMAX .GT. 0.) GO TO 20
ENU = D/RHOS
IF (ABS(ENU - 1.) .GE. 1.E-6) GO TO 20
TAUEL = 0.
GO TO 70

*** UPDATE STRAIN AND ELASTIC SHEAR STRESS AT TIME(N+1) ***

20 KS = 0
VELS = ALOG(D/DOLD)
EEN = ALOG(D/DOLD/RHOS)
EENP1 = EEN + VELS
EPN = ZEP
ZAMSUSV = AMIN1(ZAMSUSV, AMAX1(MUM - ALF * ABS(EENP1 + 1.)))
C] = AMIN1(ZAMSUSV, AMAX1(ZAM - ALF * ABS(EENP1 + 1.)))
TAUEL = C1 * (EENP1 - 1.5 * EPN)

IF (ABS(EPN) .LT. 1.E-6 .AND. ABS(TAUEL) .LT. TAUO) EPN = 0.

*** TEST 1 - TEST FOR EXCEEDING ELASTIC LIMIT AT TIME(N) ***

IF (ABS(ZEPMAX) .GT. 0. .OR. ABS(EPSAVE) .GT. 0.) GO TO 30

*** TEST 2 - TEST FOR EXCEEDING ELASTIC LIMIT AT TIME(N+1) ***

IF (ABS(TAUEL) .LT. TAUO) GO TO 70

*** INITIAL CROSSING OF ELASTIC LIMIT ***
SUBROUTINE STRES2 (Concluded)

C

KS=1
GO TO 60

C *** TEST 3 - TEST FOR ELASTIC OR PLASTIC CALCULATION AT
C     TIME(N+1) ***
C
30 IF (ABS(ZTAUYMX) .LT. ABS(TAUEL) .AND. TAUEL-ZTAUYMX .GE. 0.) GO TO 40

C *** TEST 4 - TEST FOR CROSSING FROM ELASTIC TO PLASTIC UNLOADING
C     PHASE ***
C
IF (TAUN+TAUEL .GT. 0.) GO TO 50
IF (ABS(TAUN) .GT. ABS(ZTAUY) .AND. ZTAUYMX = 0.) GO TO 60
GO TO 50

C *** TEST 5 - TEST FOR RELOADING OR REUNLOADING FROM AN ELASTIC
C     POINT AT TIME(N) ***
C
40 IF (ABS(ZTAUYMX) .GT. 0.) KS=3
GO TO 60

C *** TEST 6 - TEST FOR FIRST ELASTICALLY CALCULATED POINT IN
C     UNLOADING PHASE ***
C
50 IF (ABS(TAUN) .LE. ABS(ZTAUY)) GO TO 70
KS=4
GO TO 60

C *** CALL BECOM TO CALCULATE POINT AT TIME(N+1) ON A PLASTIC
C     LOADING OR UNLOADING CURVE ***
C
60 CALL BECOM(D,SDH,DTNP1,J,N)
GO TO 80

C *** POINT AT TIME(N+1) IS ON ELASTIC CURVE ***
C
70 ZTAUY=TAEUL
SDH=4.*TAUEL/3.
ZEPDSV=0.

C *** UPDATE TAUYMX AT TIME(N+1) ***
C
80 ZTAUYMX=AMAX1(ABS(ZTAUYMX),ABS(ZTAUY))*SIGN(1.,ZTAUY)
IH=KS
SDSTORE=SDH
SDH=SDH*F
IF (IND .EQ. 0) GO TO 100

90 RETURN

100 TAU(J)=ZTAUY $ TAUJM(J)=ZTAUYMX $ EPMABS(J)=ZEMAX
EPMAXC(J)=ZEMAXC $ EPOSV(J)=ZEPDSV $ EPSAVE(J)=ZEPSAVE
AMUSV(J)=ZAMUSV $ EP(J)=ZEP $ TAU(J)=.75*SDSTORE
RETURN
END
SUBROUTINE TSQE

FUNCTION TSQE(IP,PP,GRE,C,D,S,H,ES,ROS,EN,E,EQSTVM,EQSTAM,NCYC) TSQE
C
C** CALCULATES MU OR PTH FROM KNOWN PRESSURE AND EOS RELATION. **C
C
C  IP = 0, INVERSE EOS. IP = 1, INVERSE EoS FOR PTH = ALFA*PST. C
C
C  TSQE
C
C NC=O $ PO=EMU0=P11=0. $ G2=G2/2
ESUBC=1.0 $ ENN=0.5
IF (EQSTVM .GT. 0. .AND. E .GT. ES) ESUBC=1.+ALOG(E/ES)
IF (EQSTVM .NE. 0. ) ENN=ABS(EQSTVM)
ERAT=E/ES $ IF (E .GT. ES) ERAT=1.0
EN2=(EN/ERAT*EQSTAM)/ESUBC $ ES2=ES*ESUBC
IF (NC .EQ. 0) IXX=0
IF (PP .EQ. 0. .AND. E .LT. ES) GO TO 67
IND = IP+1
IF (PP .LE. GRE) IND = IND+2
EMU1 = (PP-GRE)/C
C
S4=0.
GO TO (10,15,20,25) IND
C
C PATH FOR COMPRESSION - SOLID PRESSURE KNOWN. ** C
10 WMU = 1.+EMU1
P1 = GRE+PH*(1.-G2*EMU1/WMU)
EMU2 = TSQE = EMU1+(PP-P1)*0.5/(PH/G2*WMU)**2+(C+EMU1*(2.*D+EMU1*3)/3)
1. **S)***(1.-G2*EMU1/WMU))/0.5*(EMU1-EMU0)/(P1-PH*G2/3)
GO TO 30
C
C PATH FOR COMPRESSION - POROUS PRESSURE KNOWN. ** C
15 WMU = 1.+EMU1
ETA = 1.-G2*EMU1/WMU
PH = EMU1*(C+EMU1*(D+EMU1*S))
P1 = (PH*ETA+GRE)/WMU
EMU2 = EMU1+(PP-P1)*0.5*(ETA*(C+EMU1*(2.*D+EMU1*3*S)))-P1-PH*G2/
1. WMU=x*2)/WMU)+0.5*(EMU1-EMU0)/(P1-PH)
GO TO 30
C
C PATH FOR EXPANSION - POROUS PRESSURE KNOWN. ** C
25 WMU = 1.+EMU1
S4=WMU**ENN
S2=H+(G-H)*S
IF (EN2*EMU1/WMU**2 .GT. 30.) S4=EXP(EN2*EMU1/WMU**2)
S3=E-ES2*(1.-S4)
P1=S2*S3
DPDMU=ROS*S2*S3+ROS*S3**ENN*(G-H)*S+ROS*S2=ES2*S4*ENN/(1.-EMU1)/WMU
1. EMU2=AMAX1(-1.+1.8*NC,AMIN1(EMU2,-1.8*NC))
GO TO 30
C
C PATH FOR EXPANSION - SOLID PRESSURE KNOWN. ** C
30 IF (ABS(PO-PP) .LT. ABS(PI-PP)) GO TO 80
IF (NC .GT. 7) PRINT 32,IP,PP,GRE,P1,EMU2,EMU1,EMU0,NC,IXX
CONTINUE
FORMAT(* IF (IP .LT. 13),* PP,GRE,P1*1P3E+10.5,* EMU2,EMU1,EMU0*X1P3E+5.
1. * NC,IXX) X=3)
62 IF (NC .EQ. 10) IXX=1+IXX+1 $ IF (IXX .GT. 10) STOP
63 IF (ABS(EMU2-EMU1) .GT. 1.E-4)*AMAX1(ABS(EMU1),1.E-3)) GO TO 75
70 CONTINUE
65 IF (IP .EQ. 1) TSQE=PTh=PTh*(1.+EMU2)
66 TSQE=EMU2
67 IF (PP .EQ. 0.) TSQE=2*ROS/(1.+.SQRT(1.-4./(EN/ERAT*EQSTAM)+ALOG(1.
1. -E/ES)))
70 RETURN
75 CONTINUE
FROM 7 TO 65
70 IF (ABS(PO-PP) .LT. ABS(P1-PP)) GO TO 80
PO = P1 $ EMU0 = EMU1
RETURN
SUBROUTINE TSQE (Concluded)

80 IF (PP .GT. GRE) GO TO 90
IF (P11 .EQ. 0. .OR. (P1-PP)*(PP-P11) .LE. 0.) GO TO 90
EMU1=0.5*(EMU1+EMU2)
90 EMU1=EMU2
95 GO TO 8
END
Appendix J

GLOSSARY

Nomenclature of Text

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( A_w )</td>
<td>Atomic weight, g/mole</td>
</tr>
<tr>
<td>( A_i )</td>
<td>Coefficients of the fit between ( \sigma_a ) and ( h )</td>
</tr>
<tr>
<td>( a )</td>
<td>Coefficient of Murnaghan equation for pressure, dyn/cm(^2)</td>
</tr>
<tr>
<td>( b )</td>
<td>Number of cells over which a detonation front is spread, or dimensionless coefficient in Murnaghan equation for pressure</td>
</tr>
<tr>
<td>( C )</td>
<td>Bulk modulus, dyn/cm(^2)</td>
</tr>
<tr>
<td>( C_b )</td>
<td>( 10^{-24} \text{ cm}^2 /\text{barn} ), a conversion factor</td>
</tr>
<tr>
<td>( C_c )</td>
<td>( 4.186 \times 10^7 \text{ erg/cal} ), a conversion factor</td>
</tr>
<tr>
<td>( C_e )</td>
<td>Effective sound speed, cm/sec</td>
</tr>
<tr>
<td>( C_F )</td>
<td>An effective coefficient of artificial viscosity</td>
</tr>
<tr>
<td>( C_L )</td>
<td>Constant in linear relation between shock velocity and particle velocity, cm/sec</td>
</tr>
<tr>
<td>( C_P )</td>
<td>Specific heat at constant pressure, dyn-cm/g/°C</td>
</tr>
<tr>
<td>( C_S )</td>
<td>Sound speed, cm/sec</td>
</tr>
<tr>
<td>( C_0 )</td>
<td>Coefficient of quadratic artificial viscosity</td>
</tr>
<tr>
<td>( C_1 )</td>
<td>Coefficient of linear artificial viscosity</td>
</tr>
<tr>
<td>( c )</td>
<td>Cohesion or shear strength at zero normal stress, dyn/cm(^2)</td>
</tr>
<tr>
<td>( D )</td>
<td>Density, g/cm(^3); or second coefficient in series expansion for Hugoniot pressure, dyn/cm(^2)</td>
</tr>
<tr>
<td>( D_\text{x} )</td>
<td>Detonation velocity, cm/sec</td>
</tr>
<tr>
<td>( E )</td>
<td>Internal energy, erg/g</td>
</tr>
<tr>
<td>( E_{\text{CJ}} )</td>
<td>Chapman-Jouguet energy, erg/g</td>
</tr>
<tr>
<td>( E_e )</td>
<td>Effective sublimation energy used in calculation of expanded states, erg/g</td>
</tr>
<tr>
<td>( E_m )</td>
<td>Melt energy, erg/g</td>
</tr>
<tr>
<td>( E^p )</td>
<td>Plastic energy, erg/g</td>
</tr>
</tbody>
</table>
$E_{rad}$  
Radiant energy, erg/g

$E_S$  
Sublimation energy, erg/g

$F$  
Thermal reduction factor

$F_{ai}, F_{bi}, F_{ci}$  
Coefficients in the thermal reduction series in Appendix D

$F_{\alpha}$  
Fraction of explosive detonated

$F_B$  
Thermal softening factor applied to shear modulus

$F_G$  
Thermal softening factor applied to yield strength

$G$  
Shear modulus, dyn/cm$^2$

$H$  
Grüneisen ratio for expanded states

$h$  
Planck's constant, $4.1354 \times 10^{-18}$ keV-sec

$I$  
Fluence, cal/cm$^2$

$I_j$  
Cumulative impulse from the front up to the $j^{th}$ coordinate, dyn-sec/cm$^2$

$I_0$  
Incident fluence in cal/cm$^2$

$J_1$  
$\sigma_1 + \sigma_2 + \sigma_3$, First invariant of the stress tensor, dyn/cm$^2$

$J_2$  
$1/2 \sigma_{11}^2 + \sigma_{12}^2$, Second invariant of the deviator stress tensor, dyn$^2$/cm$^4$

$j$  
Coordinate or cell numbers

$k$  
Boltzmann's constant, $8.6164 \times 10^{-5}$ eV/$^\circ$K; or shear strength constant in the Coulomb model of Drucker and Prager, dyn/cm$^2$

$M$  
Work-hardening modulus, dyn/cm$^2$

$M_1$  
Mass of cell 1

$M_{12}$  
Momentum between coordinates 1 and 2

$N_a$  
$6.02252 \times 10^{23}$, Avogadro's number, atom/mole

$N_c$  
Number of cells in a zone or number of constituents in a material

$N_{\phi}$  
$tan^2(45^\circ + \phi/2)$, a factor appearing in Coulomb strength calculations

$n$  
Time step (cycle) number

$P$  
Pressure, dyn/cm$^2$

$P_{CJ}$  
Chapman-Jouguet pressure, dyn/cm$^2$

$P_H$  
Hugoniot pressure, dyn/cm$^2$

$Q$  
Artificial viscous stress, dyn/cm$^2$
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Q_x$</td>
<td>Energy of an explosive, erg/g</td>
</tr>
<tr>
<td>$R$</td>
<td>Total mechanical stress in direction of propagation, dyn/cm²</td>
</tr>
<tr>
<td>$R_x$</td>
<td>Geometric ratio between successive cells</td>
</tr>
<tr>
<td>$r$</td>
<td>Radial distance in cylindrical or spherical coordinates, cm</td>
</tr>
<tr>
<td>$S$</td>
<td>Third coefficient in series expansion for Hugoniot pressure, dyn/cm²</td>
</tr>
<tr>
<td>$S_L$</td>
<td>Coefficient in linear relation between shock velocity and particle velocity</td>
</tr>
<tr>
<td>$S_e$</td>
<td>Entropy, erg/g°C</td>
</tr>
<tr>
<td>$T$</td>
<td>Spall strength, dyn/cm² or Kelvin temperature, °K; or time constant for stress relaxation, sec</td>
</tr>
<tr>
<td>$T_h$</td>
<td>Zone thickness, cm</td>
</tr>
<tr>
<td>$\Delta T_n$</td>
<td>Duration of the nth source</td>
</tr>
<tr>
<td>$t$</td>
<td>Problem time, sec</td>
</tr>
<tr>
<td>$t_b$</td>
<td>Time of detonation, sec</td>
</tr>
<tr>
<td>$U$</td>
<td>Particle velocity, cm/sec</td>
</tr>
<tr>
<td>$U_s$</td>
<td>Shock velocity, cm/sec</td>
</tr>
<tr>
<td>$u_{CJ}$</td>
<td>Chapman-Jouguet particle velocity, cm/sec</td>
</tr>
<tr>
<td>$V$</td>
<td>Specific volume, cm³/g</td>
</tr>
<tr>
<td>$V_{CJ}$</td>
<td>Chapman-Jouguet specific volume, cm³/g</td>
</tr>
<tr>
<td>$W$</td>
<td>ln(hv), with hv in keV</td>
</tr>
<tr>
<td>$X$</td>
<td>Coordinate location, cm</td>
</tr>
<tr>
<td>$X_D$</td>
<td>Point of initiation of a detonation, cm</td>
</tr>
<tr>
<td>$\bar{X}$</td>
<td>Midcell location, cm</td>
</tr>
<tr>
<td>$\Delta X$</td>
<td>Cell size in direction of propagation, cm</td>
</tr>
<tr>
<td>$\Delta X_f$</td>
<td>Last cell in a zone, cm</td>
</tr>
<tr>
<td>$\Delta X_1$</td>
<td>First cell in a zone, cm</td>
</tr>
<tr>
<td>$Y$</td>
<td>Yield strength, dyn/cm²</td>
</tr>
<tr>
<td>$Y_D$</td>
<td>Work-hardening coefficient, dyn/cm²/(g/cm³)</td>
</tr>
<tr>
<td>$Y_\tau$</td>
<td>Yield stress in shear, dyn/cm²</td>
</tr>
<tr>
<td>$Z$</td>
<td>Cell mass (g/cm², g/cm, or g for planar, cylindrical, or spherical flow)</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Volumetric thermal expansion coefficient, 1/°C or coefficient in the Coulomb model of Drucker and Prager</td>
</tr>
</tbody>
</table>

321
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta$</td>
<td>Coulomb coefficient for the effect of pressure on yield strength</td>
</tr>
<tr>
<td>$\Gamma$</td>
<td>Grüneisen's ratio</td>
</tr>
<tr>
<td>$\Gamma_e$</td>
<td>Effective Grüneisen ratio used in calculation of expanded states</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>Shear strain, or polytropic gas constant</td>
</tr>
<tr>
<td>$\delta$</td>
<td>Added change in thickness between successive cells in an arithmetic layout</td>
</tr>
<tr>
<td>$\delta_{ij}$</td>
<td>Kronecker delta: zero for $i \neq j$; one for $i = j$</td>
</tr>
<tr>
<td>$\varepsilon$</td>
<td>$1 - \rho_o/\rho$, Lagrangian strain</td>
</tr>
<tr>
<td>$\varepsilon_{ij}$</td>
<td>Strain tensor</td>
</tr>
<tr>
<td>$\varepsilon_{ij}^E$</td>
<td>Component of the elastic strain tensor</td>
</tr>
<tr>
<td>$\varepsilon_{ij}^P$</td>
<td>Component of the plastic strain tensor</td>
</tr>
<tr>
<td>$\varepsilon_{EL}$</td>
<td>$Y(2G)$, strain to the Hugoniot elastic limit</td>
</tr>
<tr>
<td>$\dot{\varepsilon}_{\text{P}}$</td>
<td>Equivalent plastic strain, defined in Eq. (4.33)</td>
</tr>
<tr>
<td>$\theta$</td>
<td>Temperature, °C; or angle between the radiation direction and normal incidence on the layer</td>
</tr>
<tr>
<td>$d\theta$</td>
<td>Small angle containing the typical cell in cylindrical or spherical geometry</td>
</tr>
<tr>
<td>$\lambda$</td>
<td>Proportionality factor used in plasticity calculations, cm$^2$/dyn</td>
</tr>
<tr>
<td>$\mu$</td>
<td>$\rho/\rho_o - 1$, a strain</td>
</tr>
<tr>
<td>$\mu_a$</td>
<td>Linear absorption coefficient, 1/cm</td>
</tr>
<tr>
<td>$\nu$</td>
<td>Vibration frequency, Hz</td>
</tr>
<tr>
<td>$\xi_i$</td>
<td>Dimensionless parabolic interpolation factors</td>
</tr>
<tr>
<td>$\rho$</td>
<td>Density, g/cm$^3$</td>
</tr>
<tr>
<td>$\rho_o$</td>
<td>Initial density</td>
</tr>
<tr>
<td>$\sigma_{ij}$</td>
<td>Thermodynamic stress in $i$ direction on $j$ plane, dyn/cm$^2$</td>
</tr>
<tr>
<td>$\sigma_{ij}^P$</td>
<td>Deviator stress in $i$ direction on $j$ plane, dyn/cm$^2$</td>
</tr>
<tr>
<td>$\sigma_{ij}^N$</td>
<td>Deviator stress computed on an elastic basis, dyn/cm$^2$</td>
</tr>
<tr>
<td>$\sigma_{ij}$</td>
<td>Effective stress $\sqrt{\frac{3}{2}} \sigma_{ij}^P \sigma_{ij}^N$, dyn/cm$^2$</td>
</tr>
<tr>
<td>$\sigma^N$</td>
<td>Effective stress based on elastically computed stresses, $\sqrt{\frac{3}{2}} \sigma_{ij}^N \sigma_{ij}^N$, dyn/cm$^2$</td>
</tr>
</tbody>
</table>
\[ \sigma_a \] Mass absorption coefficient, barns/atom
\[ \sigma_N \] Normal stress, dyn/cm\(^2\)
\[ \sigma_r \] Radial stress, dyn/cm\(^2\)
\[ \sigma_\theta \] Circumferential stress, dyn/cm\(^2\)
\[ \tau \] Shear stress, dyn/cm\(^2\)
\[ \tau_c \] Shear yield stress, dyn/cm\(^2\)
\[ \phi \] Angle of internal friction, or yield function. Yield occurs for \( \phi = 0 \). For negative values of \( \phi \), behavior is elastic; positive values are not permitted.
\[ \omega \] \(\hbar v/\hbar T\), a nondimensional quantity proportional to photon energy
Nomenclature of the PUFF code

AK(M) Initial bulk modulus of a porous material, dyn/cm\(^2\) (input)
ANGLE Angle between the direction of radiation and the normal to the layers, degrees (input)
BURN(M) Point of initiation of detonation, cm (input)
CFP Abbreviated symbol for the indicators NCMP(M), NFR(M), and NPOR(M)
CHL(J) Sound speed, cm/sec
CKS Maximum distance of wave front. Computation stops if wave reaches CKS (input), cm
COM Array containing additional variables for special material models; see Appendix C
COSQ Indicator used with NYAM (see NYAM)
COSQ(M) Coefficient of quadratic artificial viscosity (input)
C1(M) Coefficient of linear artificial viscosity in compression (input)
C2(M) Coefficient of linear artificial viscosity in tension (input)
DELFIN Size of the last cell in a zone, cm. Used for arithmetic cell layout (input)
DELTIM Calculational time for a cycle, sec
DELX Size of the first cell in a zone, cm. Used for arithmetic and geometric cell layout (input)
DET(M) Detonation velocity of an explosive, cm/sec
DHL(J) Cell density, g/cm\(^3\)
DIST(M) Number of cells over which a detonation front is spread, cm (input)
DPY Abbreviated symbol for the indicators NDS(M), NPR(M), and NYAM
DTN Previous time increment in the calculation, sec
DTNH Current time increment in the calculation, sec
DTMAX Time step desired after automatic rezoning, sec. If negative, |DTMAX| is the number of cells desired in the layer numbered |NREZON| (input)
ECAL Fluence, cal/cm\(^2\) (input)
ECJ(M) Chapman-jouguet energy, erg/g
EHL(J) Internal energy, erg/g
EI Energy at a specific photon energy in an arbitrary spectrum, cal/cm\(^2\) (input)
EMELT      Indicator used with NYAM (see NYAM)
EMELT(M)   Internal energy at melting, erg/g (input)
EQSTC(M)   Bulk modulus, dyn/cm². Read in as C for C,D,S Hugoniot pressure form, \( C_L \) for the linear \( U_s - U \) form, or \( a/b \) for the Murnaghan form (input)
EQSTD(M)   Second coefficient in the expansion for Hugoniot pressure, dyn/cm². Read in as D for C,D,S form, \( S_L \) for the linear \( U_s - U \) form, or \( b \) for the Murnaghan form (input)
EQSTE(M)   Sublimation energy, erg/g (input)
EQSTG(M)   Grüneisen ratio (input)
EQSTH(M)   Grüneisen ratio for expanded states (input)
EQSTS(M)   Third coefficient in the expansion for Hugoniot pressure, dyn/cm². Read in as S for C,D,S form, 1.0 for Murnaghan or 2.0 for linear \( U_s - U \) form (input)
EXMAT(M,I) Array containing additional property data
              \( I = 1 \) contains Coulomb coefficient (input)
              \( I = 3 \) contains initial sound speed of porous material or explosive
FBURN      Fraction of explosive detonated
GMELT      Indicator used with NYAM (see NYAM)
H(J,I)     Indicator arrays. \( H(J,1) \) indicates solid or porous state; \( H(J,2) \) shows coordinate type and path to be followed in HYDRO: \( H(J,3) \) indicates the material state in the cell
J          Coordinate or cell number
JBND(L)    Final J value of the Lth layer
JCYCS      Number of calculational cycles at which computation will terminate (input). If JCYCS is set to zero, only a layout is performed
JFIN       Last coordinate value, equals last cell number + 2
JMAT(L)    Material number in layer L. \( JMAT(L) = 0 \) if the Lth layer is a gap (input)
JREZON     Rightmost coordinate of a nonautomatic rezone (input)
JSTAR      The J value of the right-most active cell
JTS        J value of cell governing time step. In SCRIBE histories, JTS is listed as JTS plus 1000 times the number of spalled interfaces
LSUB( )    Indicator array, mainly used for initializing special material model subroutines. \( LSUB(7) \) is set to 1 at several places in the program to halt calculations because of an error
LVAR(J)  Array containing starting location of additional variables for cell J in the COM array; see Appendix C

M    Material number

MATFL  Indicator for problem type (input). MATFL > 0 means impact or explosion and MATFL is the last layer in the flyer plate. MATFL = 0. Radiation deposition
MATFL = -1 Mirror or symmetric impact
MATFL = -2 Pressure boundary at J = 1
MATFL = -3 Pressure boundary at J = JFIN

MATL  Array containing the material name (input)

MELT  Indicator used with NYAM (see NYAM)

MU(M)  Shear modulus, dyn/cm² (input)

MUP(M)  Initial shear modulus in porous material, dyn/cm² (input)

N    Current calculation cycle

NALPHA  Geometry indicator: 0 or 1 for planar, 2 for cylindrical, and 3 for spherical (input)

NARB  Indicator for an arbitrary deposition (depth-dose profile), (input)

0  Normal operation
1  Normalize energy to the ECAL designated
-I  Modify X-scale to fit the present density (I is arbitrary)
-1  Modify X-scale to fit the present density and normalize to the ECAL designated

NARZ  Maximum number of automatic rezones (input)

NBB  Number of black bodies in a spectrum (input)

NCELLS  Number of cells in a zone (input)

NCMP(M)  Indicator for a model for a composite material: zero for no model, 1 for REBAR (input)

NCON(M)  Indicator for number of constituents in a mixture or compound. Used for radiation absorption calculations only (input)

NDS(M)  Indicator for a deviator stress model: zero for standard model (Section 4), 1 for one-parameter stress relaxation model, 2 for Band dislocation model, 3 for Gilman dislocation model, 4 for two-parameter stress relaxation model, 5 for Bauschinger model, 6 for Read relaxation model for beryllium (input)

NEDIT  Number of cycles between calls to EDIT. EDIT calls may be controlled by either TEDITs or NEDIT, or by both

NEM(J)  Number of mobile dislocations or relative void volume

NET(J)  Total number of dislocations or number of voids/cm³

NFR(M)  Indicator for a fracture model: zero for no model, 1 for DFRACT, 2 for BFRACT, 3 for SHEAR2, and 4 for both BFRACT and SHEAR2 (input)

326
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NJEDIT</td>
<td>Number of lines of data in the request for historical listings (input)</td>
</tr>
<tr>
<td>NHNU</td>
<td>Number of energy values in an arbitrary spectrum (input)</td>
</tr>
<tr>
<td>NLAYER</td>
<td>Number of layers, counting blank layers or gaps. A hollow cylinder or sphere is represented with a gap as the first layer (input)</td>
</tr>
<tr>
<td>NMTRLS</td>
<td>Number of materials for which data are supplied in the problem input (input)</td>
</tr>
<tr>
<td>NPOINT</td>
<td>Number of points in a depth-dose profile (input)</td>
</tr>
<tr>
<td>NPOR(M)</td>
<td>Indicator for a porous material: zero for no model, 1 for POREQST, 2 for PORHOLT, 3 for PEST, and 4 for CAP (input)</td>
</tr>
<tr>
<td>NPR(M)</td>
<td>Indicator for pressure model: zero for EQST, 1 for explosive, 2 for ESA, 3 for Philco-Ford equation of state, 4 for variable modulus model (HYPO), 5 for GRAY equation of state, 6 for tabular equation of state, and 7 for a linear equation of state provided in HAFSTEP (input)</td>
</tr>
<tr>
<td>NREZON</td>
<td>Rezone control parameter. For positive values, NREZON means the number of rezones requested. A negative NREZON indicates automatic rezoning. See Section 5.2 for further information on rezone controls (input)</td>
</tr>
<tr>
<td>NSCRB</td>
<td>An array of indicators for controlling radiation deposition plots from DEPOS (input)</td>
</tr>
<tr>
<td>NSPEC</td>
<td>Number of spectra (input)</td>
</tr>
<tr>
<td>NTEDT</td>
<td>Number of time edits requested at specified times (input)</td>
</tr>
<tr>
<td>NTR</td>
<td>Number of the TEDIT for which a rezone is requested (input)</td>
</tr>
<tr>
<td>NVAR(M)</td>
<td>Number of extra variables required for each cell for the material model being used. Current models and extra variables required are: BFRACT2 (11), BFRACT3 (20), HYPO (3), PEST (5), REBAR (7), and SHEAR2 (variable) (input)</td>
</tr>
<tr>
<td>NYAM</td>
<td>Indicator for the number of lines containing spall strength, viscosity, thermal strength reduction, and yield data. The first word on each of these lines contain letters showing the data type (input)</td>
</tr>
<tr>
<td></td>
<td>C (COSQ) or V (VISC): artificial viscosity</td>
</tr>
<tr>
<td></td>
<td>E (EMELT) or M (MELT): thermal strength reduction factor</td>
</tr>
<tr>
<td></td>
<td>GM (GMELT): thermal reduction for shear modulus</td>
</tr>
<tr>
<td></td>
<td>T (TENS): spall strength</td>
</tr>
<tr>
<td></td>
<td>Y (YIELD): yield strength, shear modulus, work-hardening modulus</td>
</tr>
<tr>
<td>NZONES</td>
<td>Number of zones in a layer (input)</td>
</tr>
<tr>
<td>PCJ(M)</td>
<td>Chapman-Jouguet pressure dyn/cm²</td>
</tr>
<tr>
<td>PHL(J)</td>
<td>Pressure, dyn/cm² (positive in compression)</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>U(J)</td>
<td>Particle velocity, cm/sec</td>
</tr>
<tr>
<td>UZERO</td>
<td>Flyer velocity, cm/sec (input)</td>
</tr>
<tr>
<td>VCJ(M)</td>
<td>Chapman-Jouguet specific volume, cm$^3$/g</td>
</tr>
<tr>
<td>VISC</td>
<td>Indicator used with NYAM (see NYAM)</td>
</tr>
<tr>
<td>X(J)</td>
<td>Eulerian location of coordinate J, cm</td>
</tr>
<tr>
<td>YADD(M)</td>
<td>Work-hardening coefficient, dyn/cm$^2$ during input and dyn/cm$^2$/(g/cm$^3$) after resetting in GENRAT (input)</td>
</tr>
<tr>
<td>YHL(J)</td>
<td>Yield strength</td>
</tr>
<tr>
<td>YIELD</td>
<td>Indicator used with NYAM (see NYAM)</td>
</tr>
<tr>
<td>YOS</td>
<td>Initial yield strength, dyn/cm$^2$ (input)</td>
</tr>
<tr>
<td>ZHL(J)</td>
<td>Cell mass; g/cm$^2$, g/cm, or g for planar, cylindrical, or spherical geometry</td>
</tr>
</tbody>
</table>
P6
Pressure coefficient in prescribed exponential pressure boundary. P6(1) is for left boundary, P6(2) for right (input), dyn/cm²

QEXPL(M)
Energy of an explosive, erg/g (input)

RATIO
Geometric ratio used for geometric cell layout (input)

RHL(J)
Mechanical stress in the direction of propagation, dyn/cm² (positive in compression)

RHO(M)
Initial cell density, g/cm³ (input)

RHOS(M)
Initial solid density, g/cm³ (input)

SDH
Deviator stress in the direction of propagation, dyn/cm² (positive in compression)

SDT(J)
Deviator stress in the transverse (circumferential) direction in cylindrical geometry, dyn/cm²

SDURM
An indicator for radiation calculations. Set to the longest duration of an active radiation source during radiation; reset to 1.0 after radiation is complete, sec

SHL(J)
Stress in direction of propagation, dyn/cm² (positive in compression)

SS
Spectral energy that is gradually deposited into the cells during radiation

SSTOPM
Maximum stop time for radiation deposition, sec

SSTOP(N)
Stop time of Nth radiation source, sec (input)

START(N)
Start time of Nth radiation source, sec (input)

T(J)
Spall strength, dyn/cm² (negative)

TARZ
Problem time when automatic rezoning is terminated, sec (input)

TBL
hv, photon energy, keV (input)

TEDIT
Specified time at which an edit is requested (input), sec

TEMP
Black body temperature, keV (input)

TENS
Indicator used with NYAM (see NYAM)

TENS(M,I)
Spall strength, dyn/cm² (input),
I = 1 for solid
I = 2 for porous
I = 3 for interface with following material

TH
Zone thickness, cm (input)

TIME
Current time in the problem, sec

TREZON
Time interval between automatic rezones, sec (input)

TS
Stop time for the problem, sec (input)

TSR
Array used for deviator stress and fracture properties (input)

T6
Time factor in prescribed exponential pressure boundary. T6(1) is for left boundary, T6(2) for right (input), sec
REFERENCES


## DISTRIBUTION LIST

<table>
<thead>
<tr>
<th>No. of Copies</th>
<th>Organization</th>
<th>No. of Copies</th>
<th>Organization</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>Commander&lt;br&gt;Defense Technical Info Center&lt;br&gt;ATTN: DDC-DDA&lt;br&gt;Cameron Station&lt;br&gt;Alexandria, VA 22314</td>
<td>5</td>
<td>Commander&lt;br&gt;US Army Watervliet Arsenal&lt;br&gt;ATTN: Dr. T. Davidson&lt;br&gt;Dr. M. A. Hussain&lt;br&gt;Dr. S. L. Pu&lt;br&gt;Mr. D. P. Kendall&lt;br&gt;Dr. John Underwood&lt;br&gt;Watervliet, NY 12189</td>
</tr>
<tr>
<td>4</td>
<td>Director&lt;br&gt;Defense Advanced Research Projects Agency&lt;br&gt;ATTN: Tech Info&lt;br&gt;Dr. Ernest F. Blase&lt;br&gt;Dr. Bement&lt;br&gt;Dr. Ray Gogolewski&lt;br&gt;1400 Wilson Boulevard&lt;br&gt;Arlington, VA 22209</td>
<td>1</td>
<td>Commander&lt;br&gt;US Army Aviation Research and Development Command&lt;br&gt;ATTN: DRSAV-E&lt;br&gt;P. O. Box 209&lt;br&gt;St. Louis, MO 63166</td>
</tr>
<tr>
<td>1</td>
<td>Deputy Assistant Secretary of the Army (R&amp;D)&lt;br&gt;Department of the Army&lt;br&gt;Washington, DC 20310</td>
<td>1</td>
<td>Director&lt;br&gt;US Army Air Mobility Research and Development Laboratory&lt;br&gt;Ames Research Center&lt;br&gt;Moffett Field, CA 94035</td>
</tr>
<tr>
<td>1</td>
<td>Commander&lt;br&gt;US Army Materiel Development and Readiness Command&lt;br&gt;ATTN: DRCDMD-ST&lt;br&gt;5001 Eisenhower Avenue&lt;br&gt;Alexandria, VA 22333</td>
<td>1</td>
<td>Commander&lt;br&gt;US Army Communications Research and Development Command&lt;br&gt;ATTN: DRDCO-PPA-SA&lt;br&gt;Fort Monmouth, NJ 07703</td>
</tr>
<tr>
<td>6</td>
<td>Commander&lt;br&gt;US Army Armament Research and Development Command&lt;br&gt;ATTN: DRDAR-TSS (2 Cys)&lt;br&gt;J. D. Corrie&lt;br&gt;R. J. Weimer&lt;br&gt;J. Beetle&lt;br&gt;E. Bloore&lt;br&gt;Dover, NJ 07801</td>
<td>1</td>
<td>Commander&lt;br&gt;US Army Electronics Research and Development Command&lt;br&gt;Technical Support Activity&lt;br&gt;ATTN: DELSD-L&lt;br&gt;Fort Monmouth, NJ 07703</td>
</tr>
<tr>
<td>1</td>
<td>Commander&lt;br&gt;US Army Armament Materiel Readiness Command&lt;br&gt;ATTN: DRSAR-LEP-L, Tech Lib&lt;br&gt;Rock Island, IL 61299</td>
<td>1</td>
<td>Commander&lt;br&gt;US Army Harry Diamond Labs&lt;br&gt;ATTN: DRXDO-TI&lt;br&gt;2800 Powder Mill Road&lt;br&gt;Adelphi, MD 20783</td>
</tr>
<tr>
<td>No. of Copies</td>
<td>Organization</td>
<td>No. of Copies</td>
<td>Organization</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>3</td>
<td>Commander</td>
<td>4</td>
<td>Commander</td>
</tr>
<tr>
<td></td>
<td>US Army Missile Research and Development Command</td>
<td></td>
<td>US Army Research Office</td>
</tr>
<tr>
<td></td>
<td>ATTN: DRDMI-R</td>
<td></td>
<td>ATTN: Dr. Hermann Robl</td>
</tr>
<tr>
<td></td>
<td>DRDMI-YDL</td>
<td></td>
<td>Dr. E. Saibel</td>
</tr>
<tr>
<td></td>
<td>Dr. Raymond Conrad</td>
<td></td>
<td>Dr. George Mayer</td>
</tr>
<tr>
<td></td>
<td>Redstone Arsenal, AL 35809</td>
<td></td>
<td>Dr. James Murray</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>P. O. Box 12211</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Research Triangle Park</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NC 27709</td>
</tr>
<tr>
<td>3</td>
<td>Commander</td>
<td>1</td>
<td>Commander</td>
</tr>
<tr>
<td></td>
<td>US Army Mobility Equipment Research &amp; Development Cmd</td>
<td></td>
<td>US Army Research and Standardization Group (Europe)</td>
</tr>
<tr>
<td></td>
<td>ATTN: DRDME-WC</td>
<td></td>
<td>Box 65</td>
</tr>
<tr>
<td></td>
<td>DRSME-RZT</td>
<td></td>
<td>ATTN: Dr. B. Steverding</td>
</tr>
<tr>
<td></td>
<td>STSFB-MW, Dr. J. Bond</td>
<td></td>
<td>Box 65</td>
</tr>
<tr>
<td></td>
<td>Fort Belvoir, VA 22060</td>
<td></td>
<td>FBO NY 09510</td>
</tr>
<tr>
<td>1</td>
<td>Commander</td>
<td>1</td>
<td>Commander</td>
</tr>
<tr>
<td></td>
<td>US Army Tank Automotive Research &amp; Development Cmd</td>
<td></td>
<td>US Army War College</td>
</tr>
<tr>
<td></td>
<td>ATTN: DRDTA-UL</td>
<td></td>
<td>ATTN: Lib</td>
</tr>
<tr>
<td></td>
<td>Warren, MI 48090</td>
<td></td>
<td>Carlisle Barracks, PA 17013</td>
</tr>
<tr>
<td>1</td>
<td>Director</td>
<td>1</td>
<td>Commander</td>
</tr>
<tr>
<td></td>
<td>US Army TRADOC Systems Analysis Activity</td>
<td></td>
<td>US Army Command and General Staff College</td>
</tr>
<tr>
<td></td>
<td>ATTN: ATAA-SL, Tech Lib</td>
<td></td>
<td>ATTN: Archives</td>
</tr>
<tr>
<td></td>
<td>White Sands Missile Range</td>
<td></td>
<td>Fort Leavenworth, KS 66027</td>
</tr>
<tr>
<td></td>
<td>NM 88002</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Commander</td>
<td>1</td>
<td>Commander</td>
</tr>
<tr>
<td></td>
<td>US Army Materials and Mechanics Research Center</td>
<td></td>
<td>US Military Academy</td>
</tr>
<tr>
<td></td>
<td>ATTN: DRXMR-ATL</td>
<td></td>
<td>ATTN: Library</td>
</tr>
<tr>
<td></td>
<td>Dr. E. Wright</td>
<td></td>
<td>West Point, NY 10996</td>
</tr>
<tr>
<td></td>
<td>Mr. R. Shea</td>
<td></td>
<td>Office of Naval Research</td>
</tr>
<tr>
<td></td>
<td>DRXMR-H, Dr. D. Dandekar</td>
<td></td>
<td>ATTN: Code 402</td>
</tr>
<tr>
<td></td>
<td>DRXMR-T, Mr. J. Mescall</td>
<td></td>
<td>Department of the Navy</td>
</tr>
<tr>
<td></td>
<td>Watertown, MA 02172</td>
<td></td>
<td>Washington, DC 20360</td>
</tr>
<tr>
<td>1</td>
<td>Commander</td>
<td>1</td>
<td>Commander</td>
</tr>
<tr>
<td></td>
<td>Naval Research Laboratory</td>
<td></td>
<td>Naval Research Laboratory</td>
</tr>
<tr>
<td></td>
<td>ATTN: Code 2020, Tech Lib</td>
<td></td>
<td>ATTN: Code 2020, Tech Lib</td>
</tr>
<tr>
<td></td>
<td>Washington, DC 20375</td>
<td></td>
<td>Washington, DC 20375</td>
</tr>
<tr>
<td>No. of Copies</td>
<td>Organization</td>
<td>No. of Copies</td>
<td>Organization</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------------</td>
<td>--------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>7</td>
<td>Commander</td>
<td>6</td>
<td>Sandia Laboratories</td>
</tr>
<tr>
<td></td>
<td>Naval Research Laboratory</td>
<td></td>
<td>ATTN: Tech Lib</td>
</tr>
<tr>
<td></td>
<td>Engineering Materials Division</td>
<td></td>
<td>Dr. A. L. Stevens</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dr. Lee Davison</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dr. W. E. Warren</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dr. L. D. Bertholf</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dr. Marlin Kipp</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Albuquerque, NM 87115</td>
</tr>
<tr>
<td>1</td>
<td>Commander</td>
<td>7</td>
<td>SRI International</td>
</tr>
<tr>
<td></td>
<td>Naval Research Laboratory</td>
<td></td>
<td>ATTN: Dr. George R. Abrahamson</td>
</tr>
<tr>
<td></td>
<td>Metallurgy Division</td>
<td></td>
<td>Dr. Donald R. Curran</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dr. Donald A. Shockey</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dr. Lynn Seaman</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dr. R. Caligiuri</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Mr. D. Erlich</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Mr. R. Burback</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>333 Ravenswood Avenue</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Menlo Park, CA 94025</td>
</tr>
<tr>
<td>2</td>
<td>Commander</td>
<td></td>
<td>Aberdeen Proving Ground</td>
</tr>
<tr>
<td></td>
<td>Naval Surface Weapons Ctr</td>
<td></td>
<td>Dir, USAMSAA</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ATTN: DRXSY-D</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>DRXSY-MP, H. Cohen</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Cdr, USATECOM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ATTN: DRSTE-TO-F</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Dir, Wpns Sys Concepts Team, Bldg. E3516, EA</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ATTN: DRDAR-ACW</td>
</tr>
<tr>
<td>1</td>
<td>AFOSR(Dr. Alan H. Rosenstein)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bolling AFB, DC 20332</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>AFWL(Tech Lib)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Kirtland AFB, NM 87117</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Director</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Lawrence Livermore Laboratory</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ATTN: Dr. M. L. Wilkins</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>P. O. Box 808</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Livermore, CA 94550</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
USER EVALUATION OF REPORT

Please take a few minutes to answer the questions below; tear out this sheet and return it to Director, US Army Ballistic Research Laboratory, ARADCOM, ATTN: DRDAR-TSB, Aberdeen Proving Ground, Maryland 21005. Your comments will provide us with information for improving future reports.

1. BRL Report Number

2. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which report will be used.)

3. How, specifically, is the report being used? (Information source, design data or procedure, management procedure, source of ideas, etc.)

4. Has the information in this report led to any quantitative savings as far as man-hours/contract dollars saved, operating costs avoided, efficiencies achieved, etc.? If so, please elaborate.

5. General Comments (Indicate what you think should be changed to make this report and future reports of this type more responsive to your needs, more usable, improve readability, etc.)

6. If you would like to be contacted by the personnel who prepared this report to raise specific questions or discuss the topic, please fill in the following information.

   Name:

   Telephone Number:

   Organization Address: