AD-A084 995

AD-A084 995

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

US ARMY ARMAMENT RESEARCH AND DEVELOPMENT COMMAND
BALLISTIC RESEARCH LABORATORY
ABERDEEN PROVING GROUND, MARYLAND

Approved for public release; distribution unlimited.

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
CONTRACT REPORT ARBRL-CR-00420

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE *(and Subtitle)*
SRI PUFF 8 COMPUTER PROGRAM FOR ONE-DIMENSIONAL STRESS WAVE PROPAGATION

5. TYPE OF REPORT & PERIOD COVERED
Final Report, Volume II
Sep 1977 - Aug 1978

6. PERFORMING ORG. REPORT NUMBER
PYU-6802

7. AUTHOR(s)
L. Seaman
D. R. Curran, Supervisor

8. CONTRACT OR GRANT NUMBER(s)
DAAK11-77-C-0083

9. PERFORMING ORGANIZATION NAME AND ADDRESS
SRI International
333 Ravenswood Avenue
Menlo Park, CA 94025

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
US Army Armament Research & Development Command
US Army Ballistic Research Laboratory (DRDAR-BL)
Aberdeen Proving Ground, MD 21005

12. REPORT DATE
MARCH 1980

13. NUMBER OF PAGES
337

14. MONITORING AGENCY NAME & ADDRESS *(if different from Controlling Office)*

15. SECURITY CLASS. *(of this report)*
UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

Computer Code
Stress
Wave Propagation
Stress Relaxation
Plasticity
Fracture
Spall
Porous Material
Radiation Deposition
Equation of State
Finite Difference
Planar Flow
Cylindrical Flow
Spherical Flow

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

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)

DD FORM 1 JAN 73 **1473** EDITION OF 1 NOV 65 IS OBSOLETE

Block 20. (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

APPENDIXES

ILLUSTRATIONS

TABLES

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.

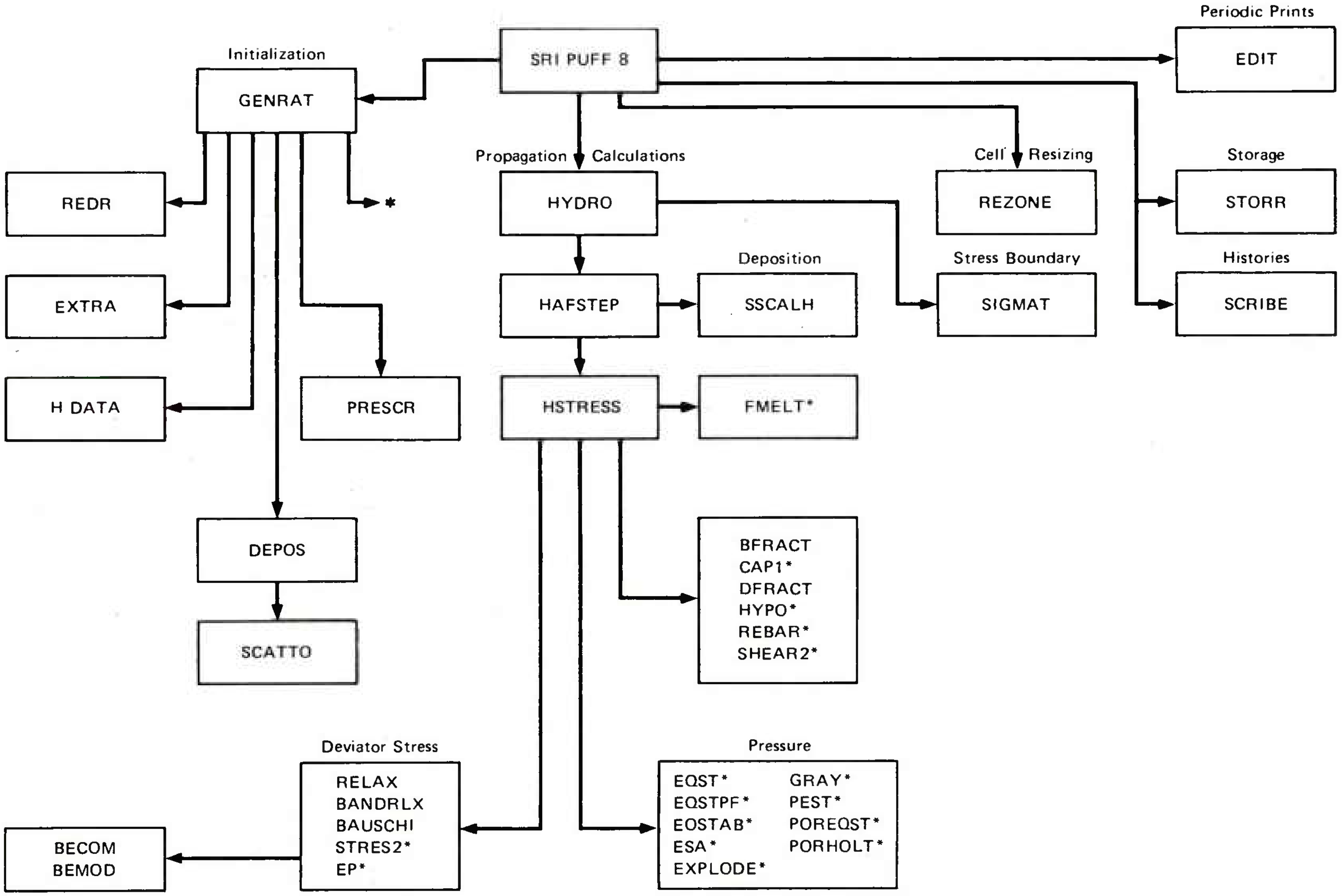
2. ORGANIZATION OF THE CODE

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).



*Starred stress-strain routines are also called by GENRAT for initialization.

MA-6802-1

FIGURE 2.1 FLOW CHART OF SRI PUFF 8

- 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 (TIME > SSTOPM), 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

$$\left(\frac{\partial U}{\partial t}\right)_H = -\frac{1}{D_o}\left(\frac{\partial R}{\partial H}\right)_t \qquad \text{(momentum)} \qquad (3.1)$$

$$\left(\frac{\partial X}{\partial t}\right)_H = U \qquad \text{(velocity)} \qquad (3.2)$$

$$\left\{\begin{aligned} &\left(\frac{\partial D}{\partial t}\right)_H = -\frac{D^2}{D_o}\left(\frac{\partial U}{\partial X}\right)_t \quad \text{or equivalently} \\ &\left(\frac{\partial X}{\partial H}\right)_t = D_o/D \end{aligned}\right. \qquad \text{(mass)} \qquad (3.3)$$

$$\left\{\begin{aligned} &\left(\frac{\partial E}{\partial t}\right) = -\frac{R}{D_o}\left(\frac{\partial U}{\partial H}\right)_t + \left(\frac{\partial E_{rad}}{\partial t}\right)_H \quad \text{or, equivalently} \\ &\left(\frac{\partial E}{\partial t}\right)_H = -R\left(\frac{\partial V}{\partial t}\right)_H + \left(\frac{\partial E_{rad}}{\partial t}\right)_H \end{aligned}\right. \qquad \text{(energy)} \qquad (3.4)$$

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_o = current and original density

R = total mechanical stress

E = internal energy

E_{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.

$$R = F(E, D, \ldots) \quad \text{(equation of state)} \tag{3.5}$$

$$= P + \sigma' + Q \tag{3.6}$$

The total mechanical stress (in the direction of propagation), R, is composed of the pressure P, the deviator stress σ' 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, $t^{n+\frac{1}{2}} = t^n + \frac{1}{2}\Delta t^{n+\frac{1}{2}}$.
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:

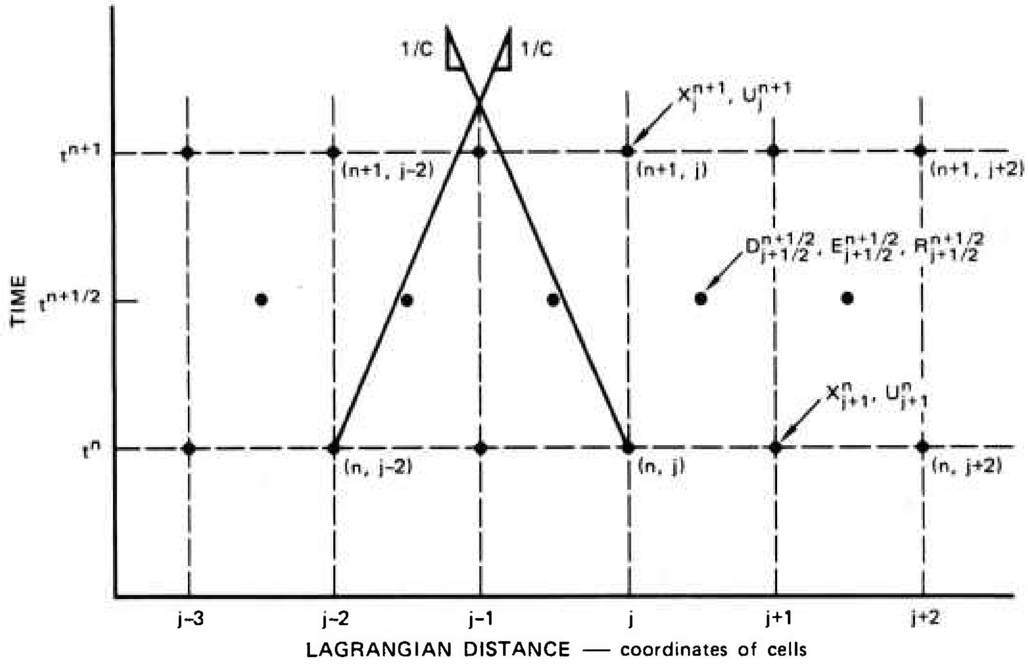


FIGURE 3.1 GRID FOR DEPICTING COORDINATES AND TIME INCREMENTS

$$D_{j+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{Z_{j+\frac{1}{2}}}{X_{j+1}^{n} - X_{j}^{n} + \frac{\Delta t^{n+\frac{1}{2}}}{2}\left(U_{j+1}^{n} - U_{j}^{n}\right)} \qquad (3.7)$$

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.

$$E_{j+\frac{1}{2}}^{n+\frac{1}{2}} = E_{j+\frac{1}{2}}^{n-\frac{1}{2}} + \left(\frac{1}{D_{j+\frac{1}{2}}^{n-\frac{1}{2}}} - \frac{1}{D_{j+\frac{1}{2}}^{n+\frac{1}{2}}}\right) R_{j+\frac{1}{2}}^{n-\frac{1}{2}} + \left(\Delta E_{j+\frac{1}{2}}^{n}\right)_{rad} \qquad (3.8)$$

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^{th} coordinate point.

$$U_{j}^{n+1} = U_{j}^{n} - \frac{R_{j+\frac{1}{2}}^{n+\frac{1}{2}} - R_{j-\frac{1}{2}}^{n+\frac{1}{2}}}{\left(Z_{j+\frac{1}{2}} + Z_{j-\frac{1}{2}}\right)/2} \Delta t^{n+\frac{1}{2}} \qquad (3.9)$$

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

$$X_{j}^{n+1} = X_{j}^{n} + \frac{1}{2}\left(U_{j}^{n+1} + U_{j}^{n}\right)\Delta t^{n+\frac{1}{2}} \qquad (3.10)$$

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), $\partial U/\partial t$ and $\partial R/\partial Z$ are both centered precisely at $(n+\frac{1}{2},j)$, and therefore, the solution scheme is of second order, although no numerical approximations to $\partial^2U/\partial t^2$ or $\partial^2R/\partial Z^2$ 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_j^{n+1} and X(J) = X_j^{n+1}, and the cell quantities are of the form RHL(J) = $R_{j+\frac{1}{2}}^{n+\frac{1}{2}}$. The time step is DTNH = $\Delta t^{n+\frac{1}{2}}$. 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.

$$P = P(E,\rho) \tag{3.11}$$

$$E = E_o + \int_o \frac{\sigma d\rho}{\rho^2} + \Delta E_r \tag{3.12}$$

where ΔE_r is radiant energy. These expressions may be solved simultaneously as in WONDY[16] 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

$$R = Q + \sigma = Q + \sigma' + P \tag{3.13}$$

where Q, σ, σ', 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:

$$R = R_1 + \frac{\partial P}{\partial \rho} \Delta\rho + \frac{\partial P}{\partial E} \Delta E \tag{3.14}$$

Thus only changes in P are considered; changes in Q and σ' are presumed to be small. The derivative $\partial P/\partial E$ 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.

$$\frac{\partial P}{\partial \rho} = \frac{R - R_1 - \partial P/\partial E \cdot \Delta E}{\Delta\rho} \tag{3.15}$$

The two derivatives have approximately the following values:

$$\frac{\partial P}{\partial \rho} = \Gamma E + \frac{C}{\rho_o} \tag{3.16}$$

$$\frac{\partial P}{\partial E} = \Gamma \rho \tag{3.17}$$

where Γ is the Grüneisen ratio, C is the bulk modulus, and ρ_o is the initial density. The estimate of internal energy E is made by evaluating Eq. (3.2) with the available densities ρ_1 and ρ_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 ΔE_r

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

(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', σ_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).

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

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 σ 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 (σ) 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

$$Q = - C_1 C_s \rho \Delta U \tag{3.21}$$

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

$$Q = C_o^2 \rho (\Delta U)^2 \tag{3.22}$$

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

$$\Delta t \leq \frac{\Delta X}{C_e} \tag{3.23}$$

where ΔX 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 and C_o^2), Herrmann et al. (Ref. 16, p. 37) derived the following reduction factor F to be applied to the time step:

$$F = \frac{1}{\sqrt{1 + (C_1 + C_2 \cdot |\Delta U|/C_s)^2} + C_1 + C_2 \cdot |\Delta U|/C_s} \tag{3.24}$$

where C_s is the material sound speed and ΔU 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

$$\sqrt{1 + C_F^2} + C_F \cong 1 + 0.5\, C_F^2 + C_F \tag{3.25}$$

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

$$X_N = U_2\, \Delta t + X_2 \tag{3.26}$$

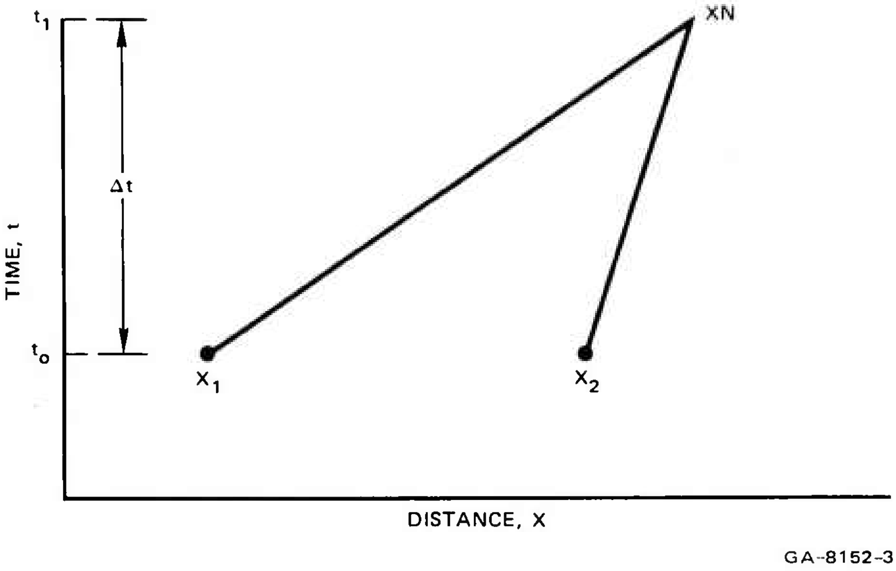


FIGURE 3.2 AN X-t PLOT FOR THE TIME STEP COMPUTATION

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

$$\Delta t = \int_{X_1}^{X_N} \frac{dX}{U + C_s} \tag{3.27}$$

It will be assumed that $U + C_s$ varies linearly from X_1 to X_2 so that

$$U + C_s = U_1 + C_{s1} + \tau(U_2 + C_{s2} - U_1 - C_{s1}) \tag{3.28}$$

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

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

$$= \frac{X_N - X_1}{C_{s2} + U_2 - C_{s1} - U_1} \ln\left(\frac{C_{s2} + U_2}{C_{s1} + U_1}\right) \tag{3.29}$$

$$= \frac{X_N - X_1}{C_{s2} + U_2 - C_{s1} - U_1} \ln\frac{1 + y}{1 - y}$$

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 $2(y + y^3 \ . \ . \ . \ )$. Only the first term is used here, giving

$$\Delta t = \frac{2(X_N - X_1)}{C_{s2} + U_2 + C_{s1} + U_1} = \frac{2(X_2 + U_2\Delta t - X_1)}{C_{s2} + U_2 + C_{s1} + U_1} \qquad (3.30)$$

When the Δt terms are collected on the left side, the result is

$$\Delta t = \frac{2(X_2 - X_1)}{C_{s2} + C_{s1} + U_1 - U_2} \qquad (3.31)$$

If the value of Δt computed from this equation is negative, the two paths do not intersect and Δt 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:

$$\Delta t = 0.8\left(\frac{2(X_2 - X_1)\ F}{C_{s2} + C_{s1} + U_1 - U_2}\right) \qquad (3.32)$$

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

$$C_e = \frac{C_{s2} + C_{s1} + U_1 - U_2}{2F} \qquad (3.33)$$

The sound speed C_s is required only to control the time step. The analytical expression for sound speed is

$$C_s^2 = \left(\frac{\partial\sigma}{\partial\rho}\right)_{\bar{S}} = \left(\frac{\partial P}{\partial\rho}\right)_{\bar{S}} + \left(\frac{\partial\sigma'}{\partial\rho}\right)_{\bar{S}} \tag{3.34}$$

where σ, σ' are the stress and deviator stress in the direction of propagation and $\bar{S}$ = 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

$$C_s^2 = \frac{C}{\rho} + \frac{4}{3}\frac{G}{\rho} \tag{3.35}$$

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/ρ 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/ρ. A procedure for numerically evaluating the partial derivative was developed for the program. The first law of thermodynamics for an isentropic ($d\bar{S} = 0$) process is

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

The usual rule for partial differentiation provides

$$dE = \left(\frac{\partial E}{\partial P}\right)_\rho dP + \left(\frac{\partial E}{\partial \rho}\right)_P d\rho \qquad (3.37)$$

From these two equations and the chain rule

$$-\left(\frac{\partial E}{\partial \rho}\right)_P = \left(\frac{\partial E}{\partial P}\right)_\rho \left(\frac{\partial P}{\partial \rho}\right)_E \qquad (3.38)$$

the required derivative is obtained:

$$\frac{dP}{d\rho} = \left(\frac{\partial P}{\partial \rho}\right)_E + \frac{P}{\rho^2}\left(\frac{\partial P}{\partial E}\right)_\rho \qquad (3.39)$$

The derivative $dP/d\rho$ was taken along an isentrope and therefore is properly written $(\partial P/\partial \rho)_{\bar{S}}$.

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

$$P = P_H\left(1 - \frac{\Gamma\mu}{2}\right) + \Gamma\rho E \qquad (3.40)$$

where

$P_H = C\mu + D\mu^2 + S\mu^3$, the pressure on the Hugoniot

C, D, S = material constants with units of bulk moduli

Γ, Γ_o = the current and initial values of Grüneisen's ratio

$\mu = \rho/\rho_o - 1$, a strain.

Then the expression for sound speed, derived from Eq. (3.39) is

$$C_s^2 = \left(\frac{\partial P}{\partial \rho}\right)_{\bar{S}} + \frac{4G}{3\rho} = \frac{C+2D\mu + 3S\mu^2}{\rho_o}\left[1 - \frac{\Gamma_o}{2}\left(1 - \frac{\rho_o}{\rho}\right)\right]$$

$$+ P_H\left(-\frac{\Gamma_o\rho_o}{2\rho^2}\right) + \frac{P}{\rho^2}\Gamma_o\rho_o + \frac{4G}{3\rho} \tag{3.41}$$

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 (JSTAR) 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 (R) 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

$$P = 1/3 \sum_i \sigma_{ii} \tag{4.1}$$

and the stress deviator elements are

$$\sigma'_{ij} = \sigma_{ij} - P\delta_{ij} \tag{4.2}$$

where σ_{ij} are stress tensor elements and δ_{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

$$P = P(E,V) \tag{4.3}$$

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

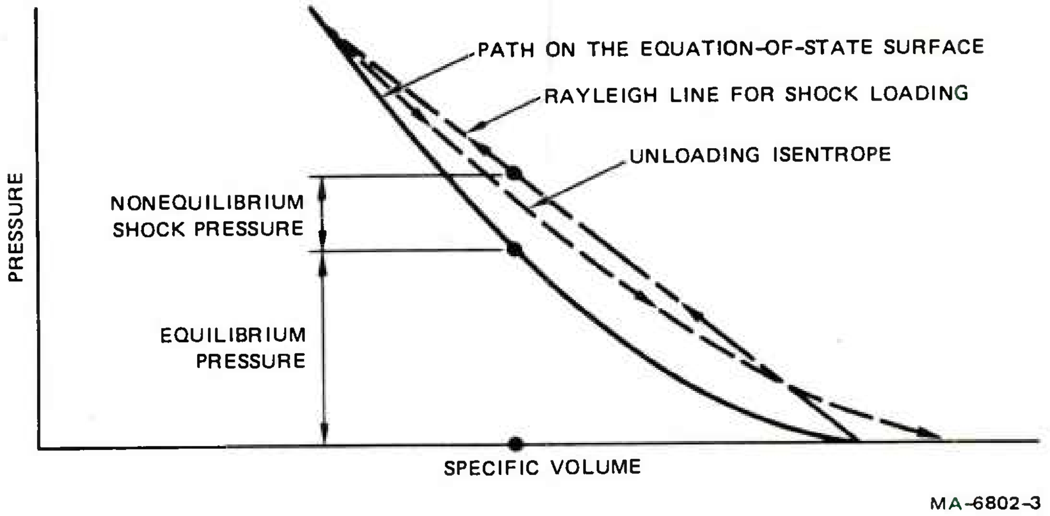


FIGURE 4.1 PRESSURE PATHS FOR SHOCK LOADING AND UNLOADING OF A MATERIAL

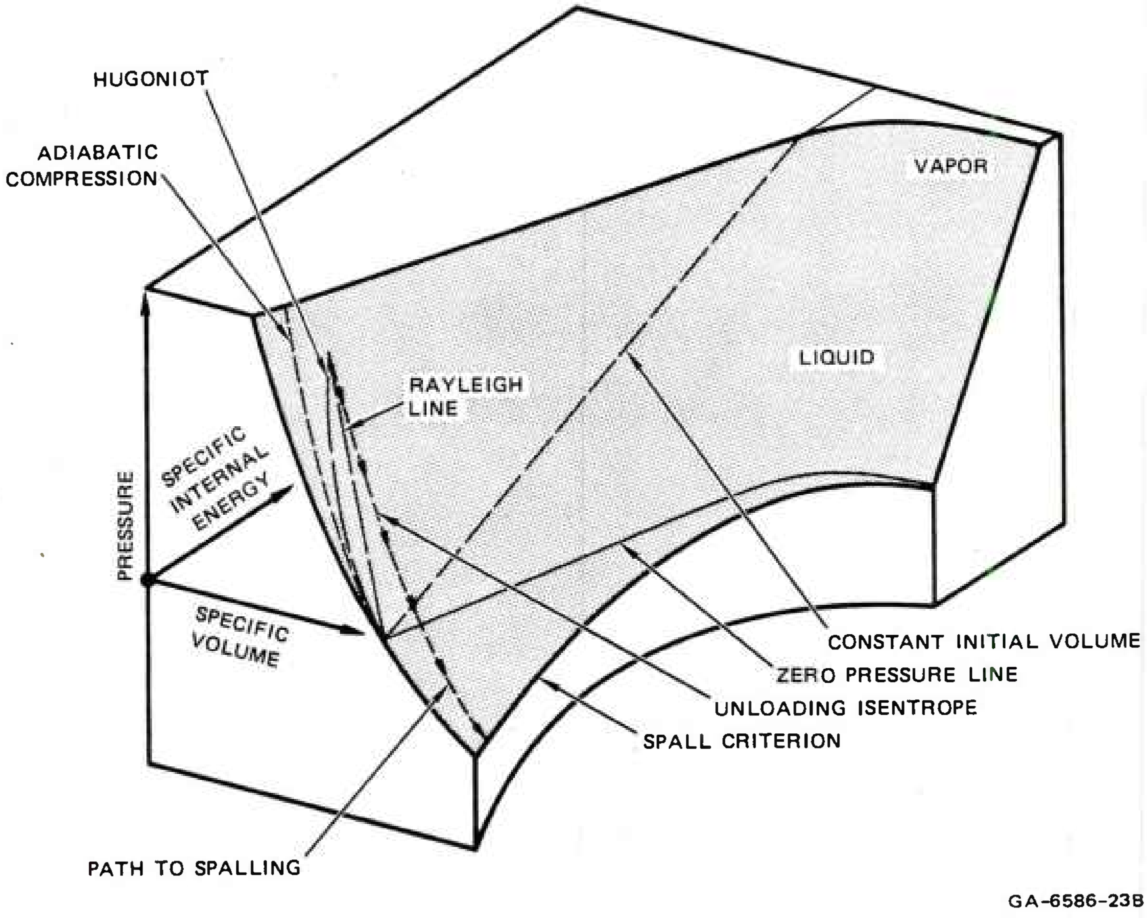


FIGURE 4.2 ENERGY-PRESSURE-VOLUME (E-P-V) SURFACE FOR A SOLID MATERIAL

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.

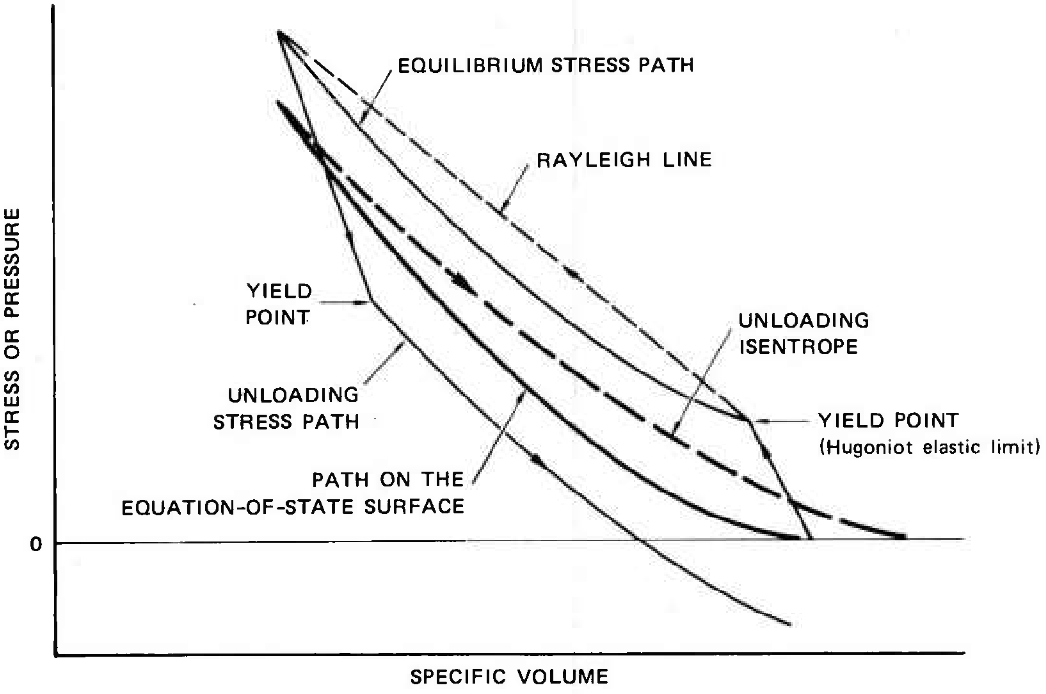
Several other lines of interest are shown in Figure 4.2. The adiabatic compression path is followed by a rapid but nonshock loading in which no heat conduction occurs. The unloading isentrope is a similar, equilibrium process without heat conduction. The zero pressure line is the locus of points obtained 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

$$V = V_o(1 + \alpha\Delta\theta) \qquad (4.4)$$

where V_o = the initial specific volume

α = the volumetric thermal expansion coefficient

$\Delta\theta$ = the change in temperature.



GA-6586-22A

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

$$\begin{cases} E = \text{vaporization energy} \\ P = 0 \end{cases}$$

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

$$P - P_{REF} = \frac{\Gamma(V)}{V} (E - E_{REF}) \qquad (4.5)$$

where

P_{REF} and E_{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 Γ 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

$$P_H = C\mu + D\mu^2 + S\mu^3 \tag{4.6}$$

where

$\mu = \frac{\rho}{\rho_o} - 1 = \frac{V_o}{V} - 1$

C = bulk modulus

D,S = coefficients with the units of moduli.

The internal energy along the Hugoniot is

$$E_H = \tfrac{1}{2}P_H(V_o - V_H) \tag{4.7}$$

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

$$P = (C\mu + D\mu^2 + S\mu^3)\left(1 - \frac{\Gamma\mu}{2}\right) + \Gamma\rho E \tag{4.8}$$

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

$$E = A(V_1) \cdot P \qquad (4.9)$$

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

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

where

ρ, ρ_{so} = current and initial density

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

$H = \gamma - 1$ for expansion at low densities and γ is the polytropic gas exponent

$E_e = E_s$ in general

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

E_s = sublimation energy for metals

= incipient vaporization energy for mixed-oxide ceramics.

n = a constant, usually 0.5 for metals, 1.67 for mixed-oxide ceramics

$$N = \frac{C}{\Gamma_o \rho_{so} E_e} \text{ for } \Gamma_1 = 0$$

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

$\Gamma = \Gamma_o + \Gamma_1\mu$, the effective Grüneisen ratio for $\rho \gtrsim \rho_{so}$

C = 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 $D = \Gamma_o C$. 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 ΔH^0_{fo} in the JANAF tables[38] for the gas state.

The Grüneisen ratio Γ may be estimated from thermal expansion data, using the relation

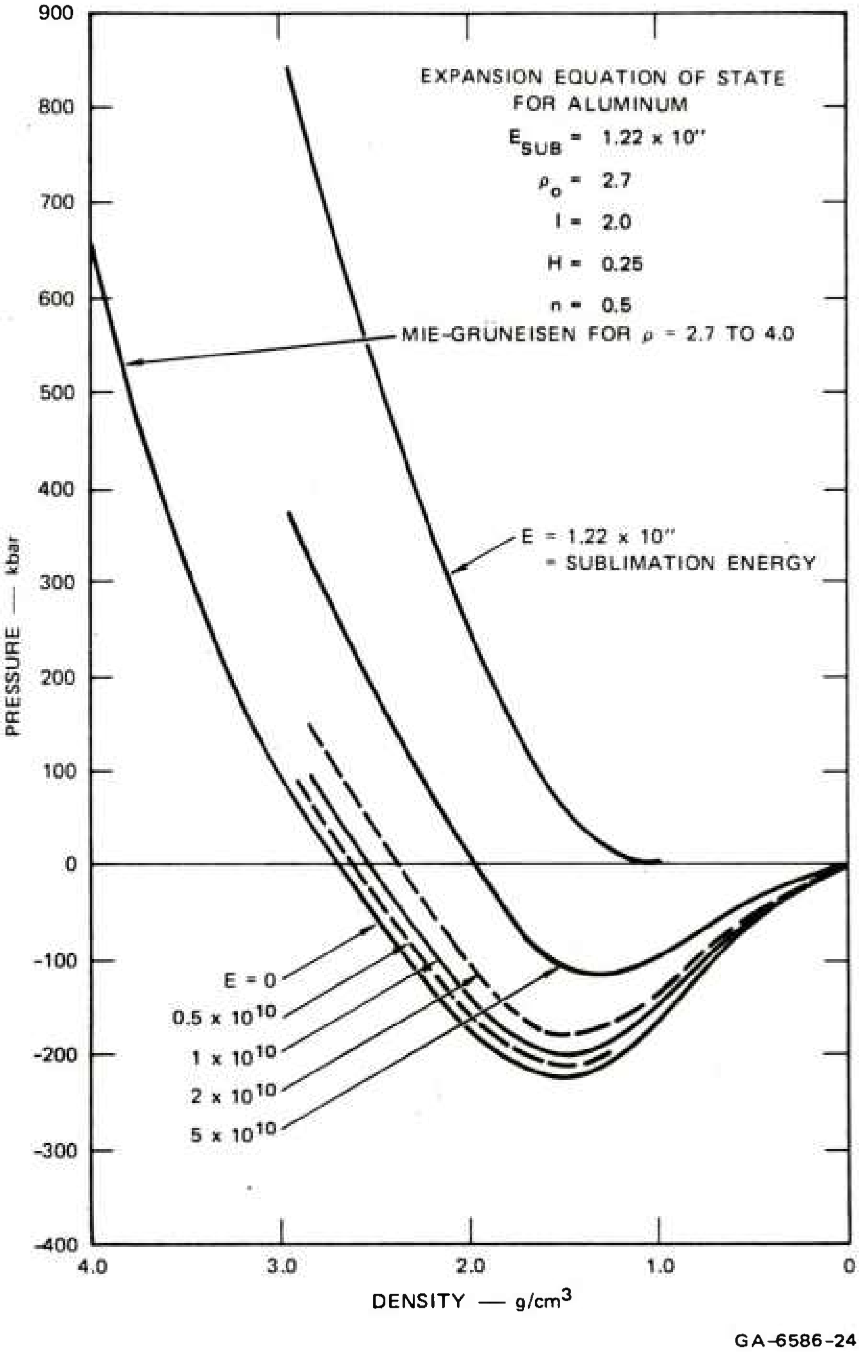


FIGURE 4.4 PRESSURE-VOLUME RELATIONS AT CONSTANT INTERNAL ENERGY FOR AN ALUMINUM

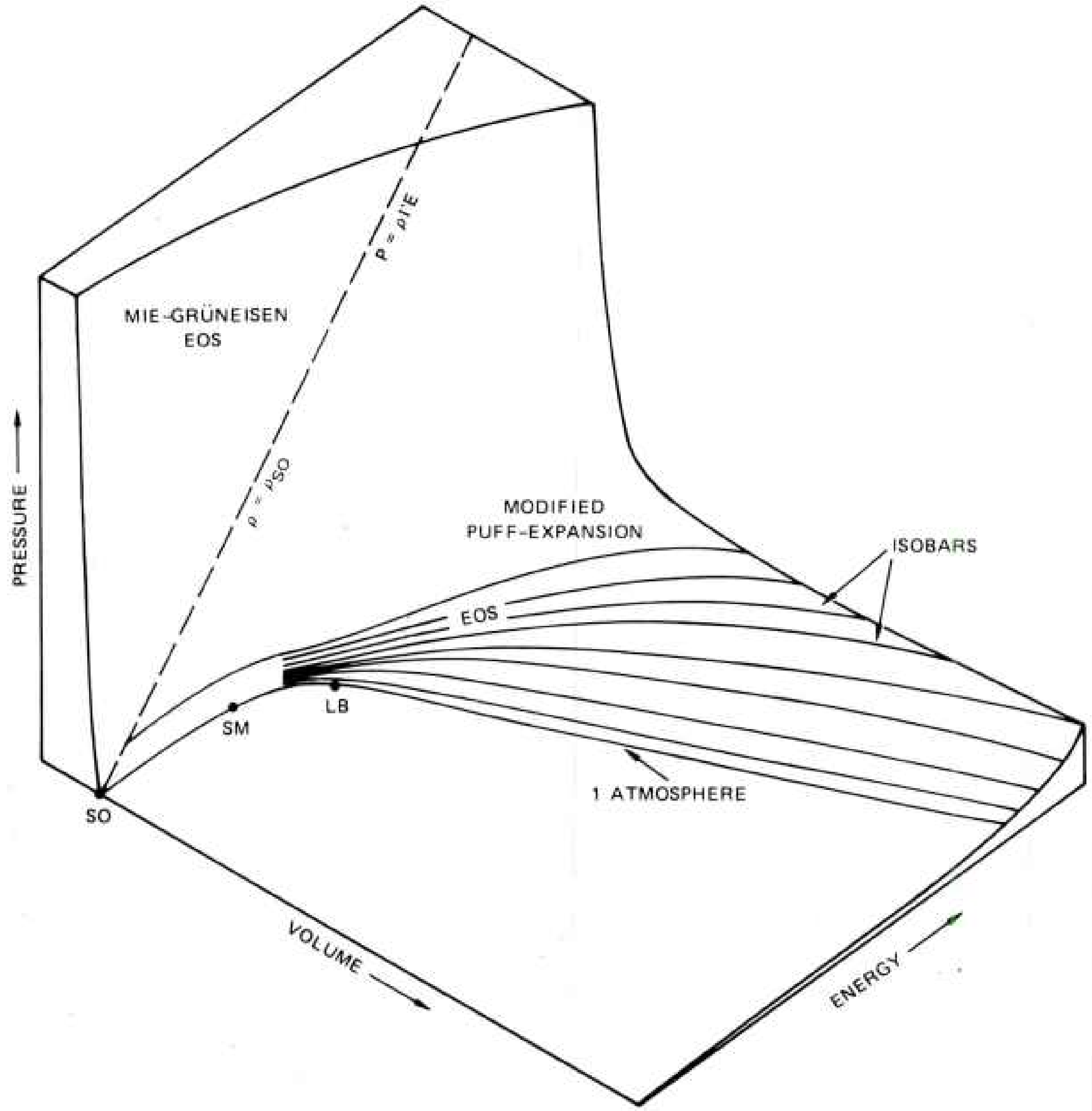


FIGURE 4.5 SKETCH OF MIE-GRÜNEISEN AND MODIFIED PUFF-EXPANSION EQUATION-OF-STATE MODEL

$$\Gamma = \frac{C\alpha}{\rho_{so} C_p} \qquad (4.11)$$

where

α = the volumetric thermal expansion coefficient

C_p = 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, Γ 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.

$$-V \left(\frac{\partial P}{\partial V}\right)_H = a + bP \qquad (4.12)$$

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

$$P_H = \frac{a}{b} \left[\left(\frac{V_o}{V}\right)^b - 1 \right] \qquad (4.13)$$

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)

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

Eq. (4.12) can be expanded to

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

Therefore

$$\begin{aligned} a &= C \\ b &= 1 + 2D/C \end{aligned} \qquad (4.16)$$

Another estimate of b is obtained from the Rice, McQueen, and Walsh[37] relation $\Gamma = D/C$.

Then

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

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

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

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

$$P_H = \rho_o U U_s = \rho_o(C_L U + S_L U^2) \qquad (4.19)$$

Next we replace the velocities by using the expression for the conservation of mass across a shock front

$$\frac{\rho}{\rho_o} = \frac{U_s}{U_s - U} \qquad (4.20)$$

and a Lagrangian strain ε

$$\varepsilon = 1 - \rho_o/\rho \qquad (4.21)$$

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

$$P_H = \frac{\rho_o C_L^2 \varepsilon}{(1 - S_L \varepsilon)^2} \qquad (4.22)$$

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

$$S_L = \tfrac{1}{2}(\frac{D}{C} + 1) \qquad (4.23)$$

$$C_L = \sqrt{\frac{C}{\rho_o}} \qquad (4.24)$$

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

$$\sigma'_{ij} = 2G\left(\varepsilon^E_{ij} - \frac{\delta_{ij}}{3}\sum_\ell \varepsilon^E_{\ell\ell}\right) \tag{4.25}$$

$$P = C\Sigma\varepsilon_{ii} \tag{4.26}$$

Here, σ'_{ij} and ε^E_{ij} are the deviatoric stress and elastic strain in the ij direction, G is the shear modulus, δ_{ij} is the Kronecker delta, P is pressure, and C is the bulk modulus. For the elastic case, $\varepsilon_{ij} = \varepsilon^E_{ij}$, 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.

$$d\varepsilon_{ij} = d\varepsilon^E_{ij} + d\varepsilon^P_{ij} \tag{4.27}$$

where $d\varepsilon_{ij}$ is the total strain increment and $d\varepsilon^P_{ij}$ 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:

$$\varepsilon'_{ij} = \varepsilon^E_{ij} - \frac{\delta_{ij}}{3}\sum_\ell \varepsilon_{\ell\ell} \tag{4.28}$$

Then Eq. (4.25) becomes

$$\sigma'_{ij} = 2G\, \varepsilon'^E_{ij} \tag{4.29}$$

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

$$\bar{\sigma} = \sqrt{\frac{3}{2}(\sigma'_{ij}\ \sigma'_{ij})} \tag{4.30}$$

where the repeated subscripts indicate summation. The yield criterion is

$$\bar{\sigma} = Y \tag{4.31}$$

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:

$$d\varepsilon_{ij}^{p} = \sigma'_{ij}\ d\lambda \tag{4.32}$$

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

$$d\bar{\varepsilon}^{p} = \sqrt{\frac{2}{3}\ d\varepsilon_{ij}^{p}\ d\varepsilon_{ij}^{p}} \tag{4.33}$$

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

$$d\bar{\varepsilon}^{p} = \frac{2}{3}\ \bar{\sigma}\ d\lambda \tag{4.34}$$

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

$$d\varepsilon_{ij}^{p} = \sigma'_{ij}\ \frac{3d\bar{\varepsilon}^{p}}{2\bar{\sigma}} \tag{4.35}$$

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,

$$\sigma'^{N}_{ij} = 2G\left(\varepsilon'^{E}_{ijo} + \Delta\varepsilon'_{ij}\right) = 2G\left(\varepsilon'^{E}_{ij} + \Delta\varepsilon_{ij}^{p}\right) \tag{4.36}$$

where

ε'^{E}_{ijo} = the elastic deviator up to the current strain step

$\Delta\varepsilon'_{ij}$ = the total deviator strain increment

ε'^{E}_{ij} = the elastic deviator strain after the current increment

$\Delta\varepsilon^{P}_{ij}$ = the plastic strain increment.

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

$$\sigma'^{N}_{ij} = \sigma'_{ij}(1 + 3Gd\bar{\varepsilon}^{P}/\bar{\sigma}) \tag{4.37}$$

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

$$\bar{\sigma}^{N} = \bar{\sigma}(1 + 3Gd\bar{\varepsilon}^{P}/\bar{\sigma}) \tag{4.38}$$

Here, $\bar{\sigma} = Y$.

Combining Eqs. (4.37) and (4.38) yields a solution for σ'_{ij}

$$\sigma'_{ij} = \sigma'^{N}_{ij}\frac{\bar{\sigma}}{\bar{\sigma}^{N}} \tag{4.39}$$

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

$$d\bar{\varepsilon}^{P} = \frac{\bar{\sigma}^{N} - \bar{\sigma}}{3G} \tag{4.40}$$

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 τ_c, which is a function of a cohesion c, normal stress σ_N, and the angle of internal friction ϕ

$$\tau_c = c + \sigma_N \tan \phi \tag{4.41}$$

Following Terzaghi,[39] this expression is transformed to

$$\sigma_1 = 2c\sqrt{N_\phi} + \sigma_3 N_\phi \tag{4.42}$$

where $N_\phi = \tan^2(45^\circ + \phi/2)$; and σ_1 and σ_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[40]

$$\sqrt{J_2'} = k + 3\alpha P \tag{4.43}$$

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

$$\bar{\sigma} = \frac{3c\sqrt{N_\phi} + \frac{3}{2}(N_\phi - 1)P}{1 + N_\phi/2} \tag{4.44}$$

The constants k and α have been replaced by c and N_ϕ 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:

$$Y = Y_o + Y_D|\Delta\rho| \tag{4.45}$$

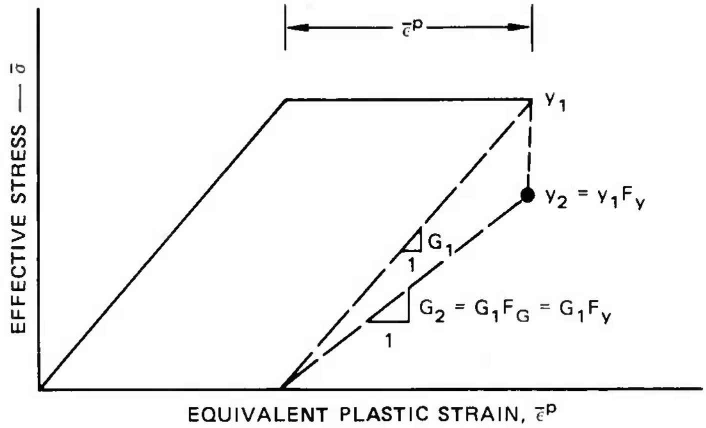
where Y_D is a work-hardening coefficient with the units of $\mathrm{dyn/cm^2/(g/cm^3)}$. This form is used mainly for historical reasons because it was present in PUFF 66. The input value of Y_D is discussed in Section 5.5. The present formulation is satisfactory for planar flow in which all strain 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 $\bar{\varepsilon}^p$, as shown. When F_G is greater than F_Y, there is an apparent increase in $\bar{\varepsilon}^p$, 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 $\bar{\varepsilon}^p$.

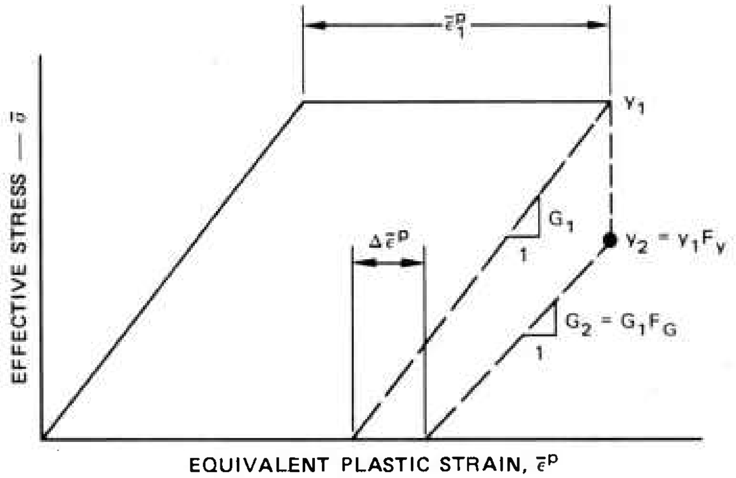
4.2 Constitutive Model Types and Switching Routine for Selecting Models

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



(a) EQUAL THERMAL-SOFTENING FUNCTIONS



(b) UNEQUAL THERMAL-SOFTENING FUNCTIONS

MA-3503-21A

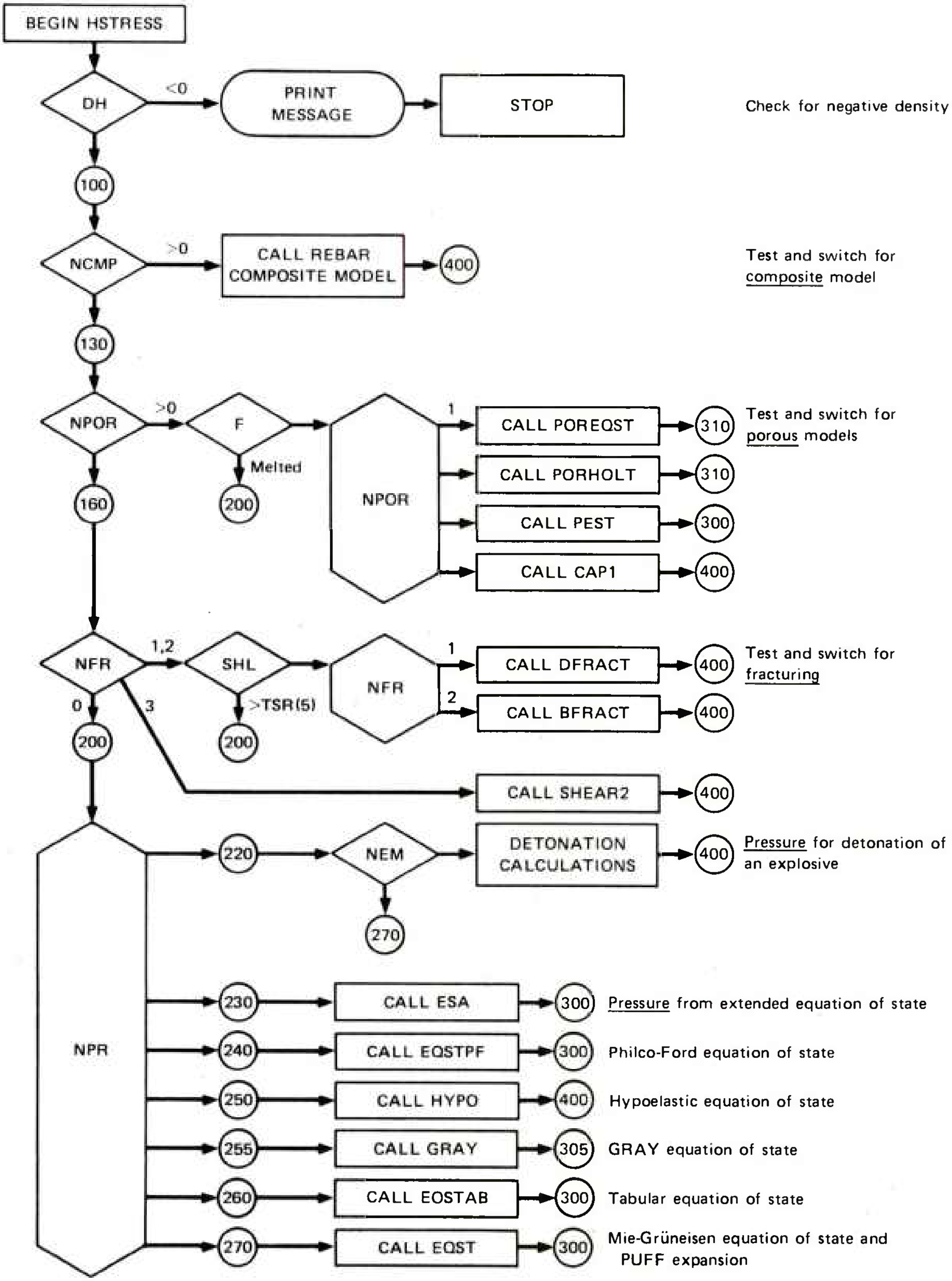
FIGURE 4.6 EFFECTS OF YIELD AND MODULUS THERMAL-SOFTENING FUNCTIONS ON PURE SHEAR STRESS-STRAIN RELATIONS possible to treat the material with a continuum model up to incipient fracture and then transfer to a fracture model. Furthermore, the material state should determine which type of fracture model 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



Check for negative density

Test and switch for composite model

Test and switch for porous models

Test and switch for fracturing

Pressure for detonation of an explosive

Pressure from extended equation of state

Philco-Ford equation of state

Hypoelastic equation of state

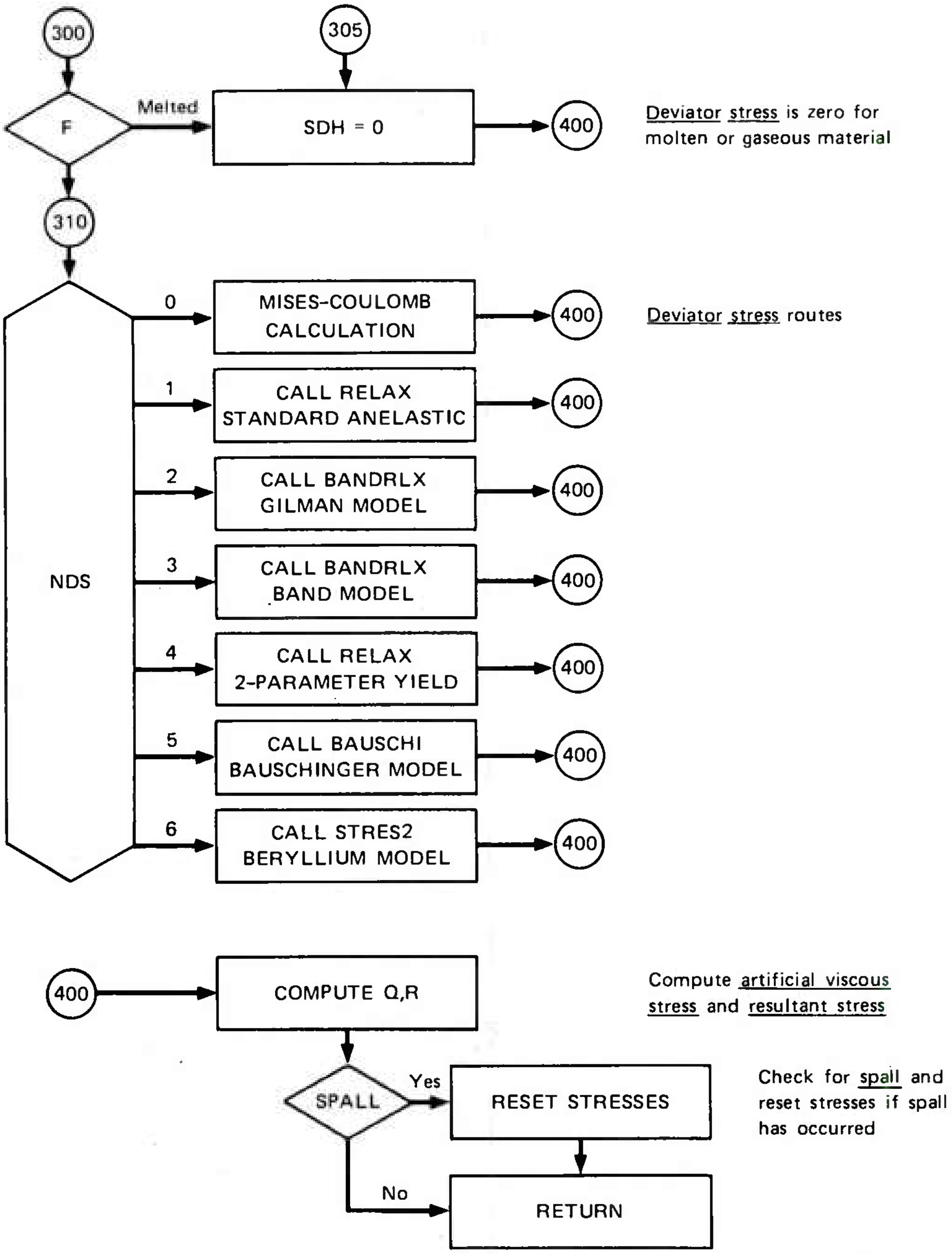
GRAY equation of state

Tabular equation of state

Mie-Grüneisen equation of state and PUFF expansion

MA-6802-8

FIGURE 4.7 FLOW CHART OF HSTRESS, STRESS-SWITCHING ROUTINE



Deviator stress is zero for molten or gaseous material

Deviator stress routes

Compute artificial viscous stress and resultant stress

Check for spall and reset stresses if spall has occurred

MA-6802-8

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_J. 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 σ_2 and σ_3 in the other two directions. The stresses in the three directions are

(1) $R = \sigma_1 + Q = \sigma_1' + P + Q = SDH + PHL + Q$

(2) $\sigma_2 = \sigma_2' + P = -\frac{1}{2}\sigma_1' + P$ for planar and spherical

$= P + \sigma_2' = PHL + SDT$ for cylindrical

(3) $\sigma_3 = \sigma_2$ for planar and spherical

$= P - \sigma_1' - \sigma_2' = PHL - SDH - SDT$ for cylindrical

The first step in the spall calculations is to compare the stresses in all three directions with the spall criterion T_J.

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 $\Delta\sigma_i^f$ equal in magnitude to the tensile stress $\Delta\sigma_i^f$ in the spall direction. Because the stress $\Delta\sigma_i^f$ 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.

$$\Delta P^f = \frac{C}{C + \frac{4}{3} G} \Delta\sigma_i^f \qquad (4.46)$$

$$\Delta\sigma_i'^f = \frac{\frac{4}{3}G}{C + \frac{4}{3}G}\,\Delta\sigma_i^f \qquad (4.47)$$

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

$$\sigma_i \rightarrow 0$$

$$\sigma_j \rightarrow \sigma_j - \Delta P^f + \tfrac{1}{2}\Delta\sigma_i'^f$$

$$\sigma_k \rightarrow \sigma_k - \Delta P^f + \tfrac{1}{2}\Delta\sigma_i'^f$$

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

$$\sigma_i' \rightarrow \sigma_i' - \Delta\sigma_i'^f$$

$$\sigma_j' \rightarrow \sigma_j' + \tfrac{1}{2}\Delta\sigma'^f$$

$$\sigma_k' \rightarrow \sigma_k' + \tfrac{1}{2}\Delta\sigma'^f$$

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 right-most 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:

	C, D, S form	Murnaghan	Linear U_s - U
EQSTC	C	a/b	C_L
EQSTD	D	b	S_L
EQSTS	S	1.0	2.0

The parameters EQSTG and EQSTH are the Grüneisen ratios Γ and 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 coefficents 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

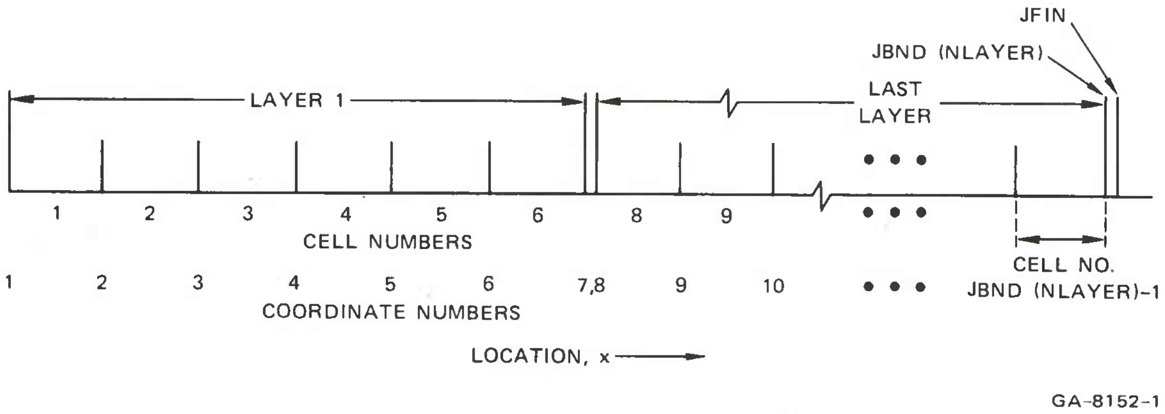
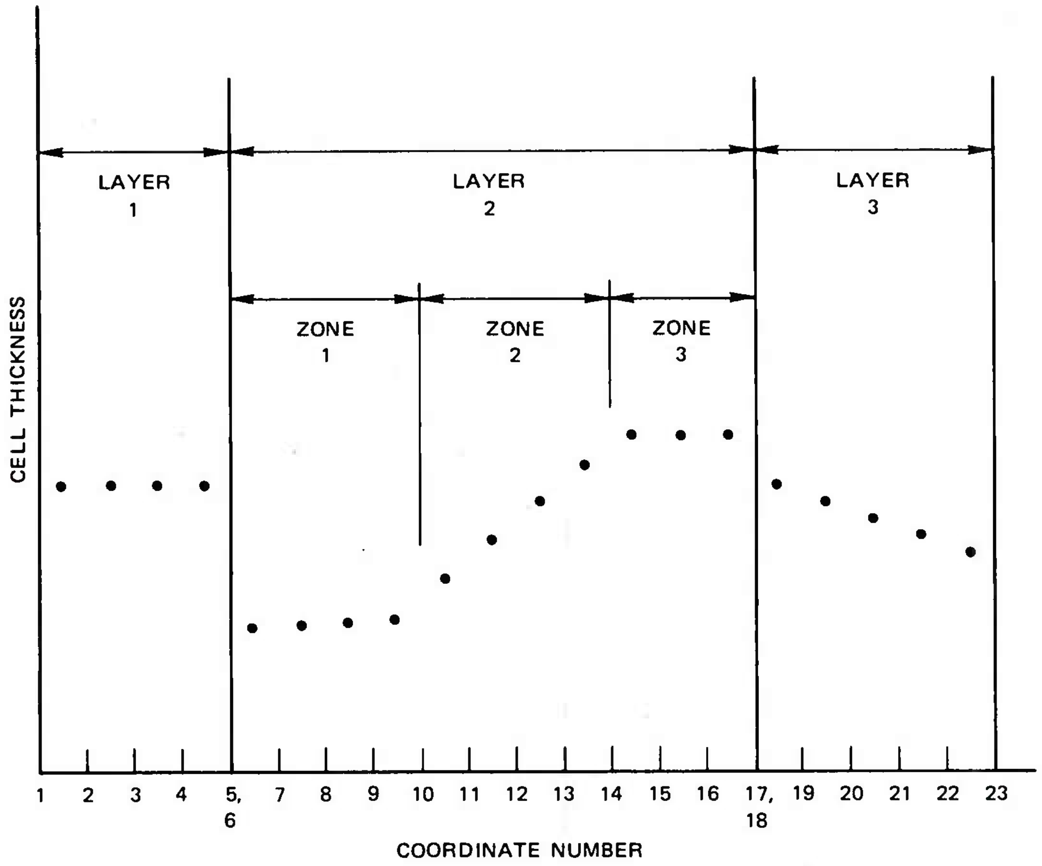


FIGURE 5.1 COORDINATE LAYOUT FOR SRI PUFF



GA-6586-4A

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 ΔX_i

$$T_h = \Sigma \Delta X_i = \Delta X_1 + (\Delta X_1 + \delta) + (\Delta X_1 + 2\delta) + [\Delta X_1 + (N_c - 1)\delta] \qquad (5.1)$$

where ΔX_1 = DELX is the thickness of the first cell, δ 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:

$$T_h = N_c \Delta X_1 + \delta(N_c - 1)N_c/2 \qquad (5.2)$$

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

$$\delta = 2\ (T_h/N_c - \Delta X_1)/(N_c - 1) \qquad (5.3)$$

According to Eq. (5.1), the expression for the thickness of the last cell, ΔX_f = DELFIN, is the following

$$\Delta X_f = \Delta X_1 + (N_c - 1)\delta \tag{5.4}$$

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

$$T_h = N_c(\Delta X_1 + \Delta X_f)/2 \tag{5.5}$$

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

$$\Delta X_f = 2T_h/N_c - \Delta X_1 \tag{5.6}$$

$$\Delta X_1 = 2T_h/N_c - \Delta X_f \tag{5.7}$$

When an arithmetic progression zoning is desired, either ΔX_1 or ΔX_f may be entered. For a uniform distribution of cell sizes, both ΔX_1 and ΔX_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 DELX = ΔX_1, and the last cell thickness is

$$\Delta X_f = \Delta X_1 \, R_x^{(N_c - 1)} \tag{5.8}$$

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

$$T_h = \Delta X_1 \frac{\left(1 - R_x^{N_c}\right)}{1 - R_x} \tag{5.9}$$

The geometric cell layout is actually overspecified by the input. Therefore the input value of ΔX_1 is disregarded and ΔX_1 is computed from Eq. (5.9) and the given values of T_h, R_x, and N_c. The nonzero value of ΔX_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 optimizing 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 interface should be matched in such a way that the interface particle velocity computed on the first cycle is approximately 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 coefficients 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. Computations 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 mismatched cases gave an initial overstress only 8% higher than the best matched case.
- For porous materials that are compacted during the computations, 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/cm^2), 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/cm^2 versus $h\nu$ (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 $h\nu$ (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 hν 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:

$$\ln \sigma_a = A_0 + A_1 w + A_2 w^2 + A_3 w^3 \qquad (5.10)$$

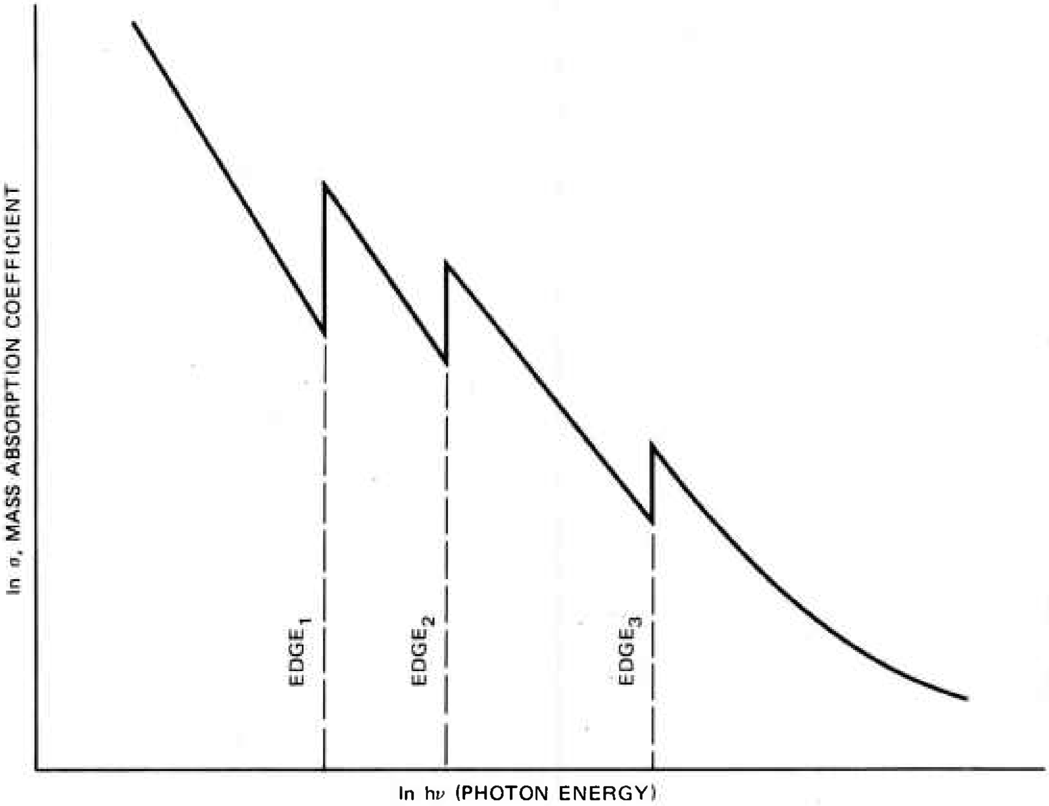
where σ_a = mass absorption coefficient (barns/atom)

w = ln(hν), with hν 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.



MA-6802-4

FIGURE 5.3 TYPICAL VARIATION OF PHOTOELECTRIC MASS ABSORPTION COEFFICIENT WITH PHOTON ENERGY

For an arbitrary spectrum, the additional data are the number of $h\nu$ values (NHNU, not more than 109), the format for reading the table of data, and NHNU pairs of $h\nu$ (TBL) and energy (EI) 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 (EI) 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

$$I_J = 1.2\sqrt{2\int_1^J Z\left[E - E_m\left(1 + \ln \frac{E}{E_m}\right)\right] dZ} \qquad (5.11)$$

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 from columns 11 to 15 and contains "NARB", "NBB", or "NHNU" for depth-dose input, black body, or arbitrary spectrum, respectively.

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/cm^2. 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:

$$Y = Y_1 + Y_D\ \Delta\rho + \beta\ P = \mathrm{YHL} + \mathrm{YADD(M)}\cdot\Delta\rho + \mathrm{EXMAT(M,1)}\cdot P \qquad (5.12)$$

where

Y_1 = the yield at the previous time (YHL)

Y_D = a work-hardening modulus (YADD)

β = 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

$$\Delta Y = \frac{\rho_2 - \rho_1}{\rho_o (0.2 - \varepsilon_{EL})} \text{ YADD} \qquad (5.13)$$

where

ρ_0, ρ_1 and ρ_2 = the initial density and the densities before and after a time increment

ε_{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

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

In GENRAT, YADD(M) is reset to Y_D.

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

$$\tau_c = c + \sigma_N \tan\phi \qquad (5.15)$$

where

τ_c = the shear stress at yield

c = the cohesion

σ_N = the normal stress on the yielding surface

ϕ = the angle of internal friction.

As shown in Section 4, this Coulomb law can be rewritten into the following form

$$Y = \frac{3c\sqrt{N_\phi}}{1 + N_\phi/2} + \frac{1.5\ (N_\phi - 1)P}{1 + N_\phi/2} \tag{5.16}$$

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:

$$\mathrm{YO(M)} = \frac{3c\sqrt{N_\phi}}{1 + N_\phi/2} \tag{5.17}$$

$$\mathrm{EXMAT(M,1)} = \frac{1.5\ (N_\phi - 1)}{1 + N_\phi/2} \tag{5.18}$$

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

$$\begin{aligned} \mathrm{CHL} &= \sqrt{\frac{\text{bulk modulus} + 4/3\ (\text{shear modulus})}{\text{density}}} && \text{for normal solids} \\ &= \text{detonation velocity for explosives} \\ &= \text{EXMAT(M,3) for porous materials.} \end{aligned} \tag{5.19}$$

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: NEM = TSR(M,21)

Gilman model: NEM = TSR(M,19)

Bauschinger model: NEM = 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) = N, normal coordinate inside a material
L, left interface of a layer
R, right interface of a layer
S, spalled interface or free surface
M, mirror or symmetric boundary
P, pressure history boundary
I, infinite boundary.

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 nonradiation 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 erg/g and cal/g, the cumulative amount of energy absorbed in 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.

```
DATE = 78/08/17. IDENT 4409-1    DUCTILE FRACTURE IN HOT 1145 AL UNDER IMPACT              SHOCKEY

 J     DX          X(J)       U(J)      YHL(J)    CHL(J)     DHL(J)      T(J)      ZHL(J)     EHL(J)        MATERIAL     COND     J
       CM          CM        CM/SEC    DYN/CM2   CM/SEC     GM/CM3     DYN/CM2    GM/CM2     ERG/GM
 1 3.130E-02 0.          1.460E+04 4.130E+09 6.716E+05 2.784E+00-1.000E+11 8.714E-02 0.             1145 AL     S S B     1
 2 3.130E-02 3.130E-02 1.460E+04 4.130E+09 6.716E+05 2.784E+00-1.000E+11 8.714E-02 0.             1145 AL     S N B     2
 3 3.130E-02 6.260E-02 1.460E+04 4.130E+09 6.716E+05 2.784E+00-1.000E+11 8.714E-02 0.             1145 AL     S N B     3
 4 3.130E-02 9.390E-02 1.460E+04 4.130E+09 6.716E+05 2.784E+00-1.000E+11 8.714E-02 0.             1145 AL     S N B     4
 5 3.130E-02 1.252E-01 1.460E+04 4.130E+09 6.716E+05 2.784E+00-1.000E+11 8.714E-02 0.             1145 AL     S N B     5
 6 3.130E-02 1.565E-01 1.460E+04 4.130E+09 6.716E+05 2.784E+00-1.000E+11 8.714E-02 0.             1145 AL     S N B     6
 7 3.130E-02 1.878E-01 1.460E+04 4.130E+09 6.716E+05 2.784E+00-1.000E+11 8.714E-02 0.             1145 AL     S N B     7
 8 3.130E-02 2.191E-01 1.460E+04 4.130E+09 6.716E+05 2.784E+00-1.000E+11 8.714E-02 0.             1145 AL     S N B     8
 9 3.130E-02 2.504E-01 1.460E+04 4.130E+09 6.716E+05 2.784E+00-1.000E+11 8.714E-02 0.             1145 AL     S N B     9
10 3.130E-02 2.817E-01 1.460E+04 4.130E+09 6.716E+05 2.784E+00-1.000E+11 8.714E-02 0.             1145 AL     S N B    10
11 0.          3.130E-01 1.460E+04 4.130E+09 6.716E+05 2.784E+00-1.000E+00 0.          0.             1145 AL     S L B    11

12 3.175E-02 3.130E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S R B    12
13 3.175E-02 3.447E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    13
14 3.175E-02 3.765E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    14
15 3.175E-02 4.082E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    15
16 3.175E-02 4.400E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    16
17 3.175E-02 4.717E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    17
18 3.175E-02 5.035E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    18
19 3.175E-02 5.352E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    19
20 3.175E-02 5.670E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    20
21 3.175E-02 5.987E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    21
22 3.175E-02 6.305E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    22
23 3.175E-02 6.622E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    23
24 3.175E-02 6.940E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    24
25 3.175E-02 7.257E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    25
26 3.175E-02 7.575E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    26
27 3.175E-02 7.892E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    27
28 3.175E-02 8.210E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    28
29 3.175E-02 8.527E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    29
30 3.175E-02 8.845E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    30
31 3.175E-02 9.162E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+11 8.587E-02 4.460E+09 1145 AL H S N B    31
32 0.          9.480E-01 0.              2.000E+09 6.814E+05 2.705E+00-1.000E+00 0.          0.          1145 AL H S L B    32
TIME TO COMPLETE  GENRAT IS       .776 SECONDS.
```

FIGURE 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

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

```
                         ****    SR1 PUFF B    ****

  DATE =  78/03/17.
IDENT = NEUTRON GEN/CONFIGURATION W/ZRCLY-2 CLADDING                            IND=  , IN= 5
NTEOT =           1     NJEDIT =      1 NREZON =          0                   1 IND=  , IN= 5
TEDIT =   1.000E-09-0.       -0.       -0.       -0.       -0.       -0.        IND=  , IN= 5  SEC
JEDIT =  S1,1.03,1.06,1.09,1.2,1.4,1.6,1.8,2.5,3.5,4.5                          IND=  , IN= 5
NEDIT =           5 JCYCS =           0 CKS =     1.000E+01 TS =      6.000E-06 IND=  , IN= 5 , , , CM,  SEC
NMTRLS =          4 MATFL =           0 UZERO =   0.                 ********** IND=  , IN= 5 , , , CM/SEC
         NSCRB INDICATORS  -  RAD. PLOTS  1-E, 2-T, 3-P, 4- ,      SCRIBE HISTORIES  1-6S, 2-I2S, 3-OS,Y, 4-R,V, 5-U,1, 6-NEM,NET

U-3PCTMO            RHOS =    1.786E+01 CFP = 000 DPY = 004        -0        -0 IND=  , IN= 5 G/CM3
EQST =    1.202E+12 1.459E+12 2.059E+10 2.030E+00 2.500E-01 5.840E+12-0.        IND=  , IN= 5 DYN/CM2,=, ERG/G, , , DYN/CM2, ERG/G
MELT  =   2.050E+09 1.271E+09 1.160E-01 5.000E-02 0.       -0.       -0.        IND=  , IN= 5 ERG/G,
YIELD =   9.100E+08 6.314E+11-0.       -0.       -0.        0.        0.        IND=  , IN= 5 DYN/CM2,  DYN/CM2,  DYN/CM2,
TENS =   -8.000E+09-1.000E+11 0.        0.        0.        0.        0.        IND=  , IN= 5 DYN/CM2,=
VISC =    4.000E+00 5.000E-02 5.000E-02 0.        0.        0.        0.        IND=  , IN= 5

NA-600C             RHOS =    8.080E-01 CFP = 000 DPY = 001        -0        -0 IND=  , IN= 5 G/CM3
EQST =    3.200E+10 0.        1.000E+11 1.000E+00 2.500E-01-0.       -0.        IND=  , IN= 5 DYN/CM2,=, ERG/G, , , DYN/CM2, ERG/G
TENS =   -1.000E+11-1.000E+11 0.        0.        0.        0.        0.        IND=  , IN= 5 DYN/CM2,=

     ZIRCALOY-2     RHOS =    6.550E+00 CFP = 000 DPY = 004        -0        -0 IND=  , IN= 5 G/CM3
EQST =    9.031E+11 0.        6.714E+10 7.700E-01 2.500E-01 0.       -0.        IND=  , IN= 5 DYN/CM2,=, ERG/G, , , DYN/CM2, ERG/G
EMELT =   6.615E+09 2.381E+09-7.200E-02 2.400E-01-6.000E-02-0.       -0.        IND=  , IN= 5 ERG/G,
VISC =    4.000E+00 5.000E-02 5.000E-02 0.        0.        0.        0.        IND=  , IN= 5
TENS =   -1.000E+11-1.000E+11 0.        0.        0.        0.        0.        IND=  , IN= 5 DYN/CM2,=
YIELD =   3.000E+09 3.472E+11-0.       -0.       -0.        0.        0.        IND=  , IN= 5 DYN/CM2,  DYN/CM2,  DYN/CM2,

NAK-100C            RHOS =    8.470E-01 CFP = 000 DPY = 001        -0        -0 IND=  , IN= 5 G/CM3
EQST =    5.300E+10 0.        1.000E+11 1.000E+00 2.500E-01-0.       -0.        IND=  , IN= 5 DYN/CM2,=, ERG/G, , , DYN/CM2, ERG/G
TENS =   -1.000E+11-1.000E+11-1.000E+11 0.        0.        0.        0.        IND=  , IN= 5 DYN/CM2,=

NLAYERS =         4 JMAT =       1    2    3    4   -0   -0   -0   -0   -0   -0 IND=  , IN= 5
NZONES= 1        29 CELLS IN  5.350E-01 CM OX =   6.000E-03 RATIO =   9.346E-01 IND=  , IN= 5 CM, CM,
NZONES= 1         3 CELLS IN  1.270E-02 CM       -0.                 -0.        IND=  , IN= 5 CM, CM.
NZONES= 1         3 CELLS IN  2.540E-02 CM       -0.                 -0.        IND=  , IN= 5 CM, CM,
NZONES= 1        12 CELLS IN  5.000E-02 CM       -0.                 -0.        IND=  , IN= 5 CM, CM,
NSPEC =           1 ANGLE     0.       -0.       -0.       -0.       -0.        IND=  , IN= X -DEPOS-   ANGLE FROM NORMAL(DEG)
PROTON    NARB =  0 ECAL =     .590E+00 START =   0.        SSTOP =    .200E-06 IND=  , IN= X -DEPOS-  , , ,CAL/CM2,SEC,SEC
IDENT = U           TH = NP =         4 (1P8E10.3)                              IND=  , IN= 5 -DEPOS-
0,        1.000E+00 2.000E-01 1.000E+00 4.000E-01 1.000E+00 5.350E-01 1.000E+00 IND=  , IN= 5 -DEPOS-  .CM,ERG/G
IDENT = NA          TH = NP           3 (1P8E10.3)                              IND=  , IN= 5 -DEPOS-
0.        4.000E-01 6.350E-03 4.000E-01 1.270E-02 4.000E-01                     IND=  , IN= 5 -DEPOS-  ,CM,ERG/G
IDENT = ZRCLY-2     TH = NP           3 (1P8E10.3)                              IND=  , IN= 5 -DEPOS-
0.        4.000E-01 1.270E-02 4.000E-01 2.540E-02 4.000E-01                     IND=  , IN= 5 -DEPOS-  ,CM,ERG/G
IDENT = NAK         TH = NP=          3 (1P8E10.3)                              IND=  , IN= 5 -DEPOS-
0.        4.000E-01 2.000E-02 4.000E-01 5.000E-02 4.000E-01                     IND=  , IN= 5 -DEPOS-  ,CM,ERG/G
H =          1    2           M H =       52    2         M
$NLIST RHOS(1)=1.845E+01,EQSTN(1)=1.55867E0,EHL(1)=30*1.065E+09$                INPUT FROM -EXTRA- ROUTINE
EOF ENCOUNTERED BY EXTRA
```

FIGURE 6.1 SAMPLE GENRAT 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 $TAU = \Sigma NL^3$ 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

```
DATE =   78/03/09. IDENT PROJECTILE INPUT ON CONCRETE AT 22.34M/SEC - TEST REBAR

  J   DX         X(J)       U(J)      YHL(J)     CHL(J)     DHL(J)     T(J)      ZHL(J)     EHL(J)      MATERIAL    COND     J
      CM         CM        CM/SEC    DYN/CM2    CM/SEC     GM/CM3    DYN/CM2    GM/CM2     ERG/GM
  1 7.257E-01 0.         2.234E+03 1.222E+10 5.843E+05 7.850E+00-1.000E+11 5.697E+00 0.           IMPACTOR   S S B     1
  2 7.257E-01 7.257E-01 2.234E+03 1.222E+10 5.843E+05 7.850E+00-1.000E+11 5.697E+00 0.           IMPACTOR   S N B     2
  3 7.257E-01 1.451E+00 2.234E+03 1.222E+10 5.843E+05 7.850E+00-1.000E+11 5.697E+00 0.           IMPACTOR   S N B     3
  4 7.257E-01 2.177E+00 2.234E+03 1.222E+10 5.843E+05 7.850E+00-1.000E+11 5.697E+00 0.           IMPACTOR   S N B     4
  5 7.257E-01 2.903E+00 2.234E+03 1.222E+10 5.843E+05 7.850E+00-1.000E+11 5.697E+00 0.           IMPACTOR   S N B     5
  6 7.257E-01 3.629E+00 2.234E+03 1.222E+10 5.843E+05 7.850E+00-1.000E+11 5.697E+00 0.           IMPACTOR   S N B     6
  7 7.257E-01 4.354E+00 2.234E+03 1.222E+10 5.843E+05 7.850E+00-1.000E+11 5.697E+00 0.           IMPACTOR   S N B     7
  8 7.257E-01 5.080E+00 2.234E+03 1.222E+10 5.843E+05 7.850E+00-1.000E+11 5.697E+00 0.           IMPACTOR   S N B     8
  9 7.257E-01 5.806E+00 2.234E+03 1.222E+10 5.843E+05 7.850E+00-1.000E+11 5.697E+00 0.           IMPACTOR   S N B     9
 10 7.257E-01 6.531E+00 2.234E+03 1.222E+10 5.843E+05 7.850E+00-1.000E+11 5.697E+00 0.           IMPACTOR   S N B    10
 11 7.257E-01 7.257E+00 2.234E+03 1.222E+10 5.843E+05 7.850E+00-1.000E+11 5.697E+00 0.           IMPACTOR   S N B    11
 12 7.257E-01 7.983E+00 2.234E+03 1.222E+10 5.843E+05 7.850E+00-1.000E+11 5.697E+00 0.           IMPACTOR   S N B    12
 13 7.257E-01 8.709E+00 2.234E+03 1.222E+10 5.843E+05 7.850E+00-1.000E+11 5.697E+00 0.           IMPACTOR   S N B    13
 14 7.257E-01 9.434E+00 2.234E+03 1.222E+10 5.843E+05 7.850E+00-1.000E+11 5.697E+00 0.           IMPACTOR   S N B    14
 15 0.         1.016E+01 2.234E+03 1.222E+10 5.843E+05 7.850E+00-1.000E+00 0.         0.           IMPACTOR   S L B    15

 16 5.080E-01 1.016E+01 0.         0.         4.995E+05 2.220E+00-1.000E+11 1.128E+00 0.           CONCRETE   P R B    16
 17 0.         1.067E+01 0.         0.         4.995E+05 2.220E+00-1.000E+00 0.         0.           CONCRETE   P L B    17

 18 5.080E-01 1.067E+01 0.         0.         5.458E+05 2.501E+00-1.000E+11 1.271E+00 0.           REBAR      S R B    18
 19 0.         1.118E+01 0.         0.         5.458E+05 2.501E+00-1.000E+00 0.         0.           REBAR      S L B    19

 20 5.080E-01 1.118E+01 0.         0.         4.995E+05 2.220E+00-1.000E+11 1.128E+00 0.           CONCRETE   P R B    20
 21 5.080E-01 1.168E+01 0.         0.         4.995E+05 2.220E+00-1.000E+11 1.128E+00 0.           CONCRETE   P N B    21
 22 5.080E-01 1.219E+01 0.         0.         4.995E+05 2.220E+00-1.000E+11 1.128E+00 0.           CONCRETE   P N B    22
 23 5.080E-01 1.270E+01 0.         0.         4.995E+05 2.220E+00-1.000E+11 1.128E+00 0.           CONCRETE   P N B    23
 24 5.080E-01 1.321E+01 0.         0.         4.995E+05 2.220E+00-1.000E+11 1.128E+00 0.           CONCRETE   P N B    24
 25 5.080E-01 1.372E+01 0.         0.         4.995E+05 2.220E+00-1.000E+11 1.128E+00 0.           CONCRETE   P N B    25
 26 0.         1.422E+01 0.         0.         4.995E+05 2.220E+00-1.000E+00 0.         0.           CONCRETE   P L B    26

 27 5.080E-01 1.422E+01 0.         0.         5.458E+05 2.501E+00-1.000E+11 1.271E+00 0.           REBAR      S R B    27
 28 0.         1.473E+01 0.         0.         5.458E+05 2.501E+00-1.000E+00 0.         0.           REBAR      S L B    28

 29 5.080E-01 1.473E+01 0.         0.         4.995E+05 2.220E+00-1.000E+11 1.128E+00 0.           CONCRETE   P R B    29
 30 0.         1.524E+01 0.         0.         4.995E+05 2.220E+00-1.000E+00 0.         0.           CONCRETE   P L B    30
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

```
DATE =  78/03/17. IDENT = NEUTRON GEN/CONFIGURATION W/ZRCLY-2 CLADDING

 J        X          X        DEPOS      DEPOS    PCT TR.      TEMP.     PRESS.    IMPULSE      MATERIAL      COND       J        OX   ABSORBED
        INCH        CM      ERGS/GM     CAL/GM               DEG. C       KBAR      KTAPS                                     CM    CAL/CM2
 0  0.000000  0.000000 2.470E+07 5.900E-01  1000.000 0.          6.955E-01 0.         U-3PCTMO
 1  0.000000  0.000000 2.470E+07 5.900E-01   924.587 0.          8.797E-01 0.         U-3PCTMO      S H R      1 4.072E-02      4.291
 2   .016030   .040716 2.470E+07 5.900E-01   854.106 0.          8.797E-01 0.         U-3PCTMO      S N R      2 3.805E-02      8.301
 3   .031011   .078769 2.470E+07 5.900E-01   788.235 0.          8.797E-01 0.         U-3PCTMO      S N R      3 3.556E-02     12.048
 4   .045013   .114333 2.470E+07 5.900E-01   726.671 0.          8.797E-01 0.         U-3PCTMO      S N R      4 3.324E-02     15.551
 5   .058099   .147571 2.470E+07 5.900E-01   669.134 0.          8.797E-01 0.         U-3PCTMO      S N R      5 3.106E-02     18.824
 6   .070329   .178636 2.470E+07 5.900E-01   615.359 0.          8.797E-01 0.         U-3PCTMO      S N R      6 2.903E-02     21.884
 7   .081759   .207669 2.470E+07 5.900E-01   565.102 0.          8.797E-01 0.         U-3PCTMO      S N R      7 2.713E-02     24.743
 8   .092442   .234803 2.470E+07 5.900E-01   518.131 0.          8.797E-01 0.         U-3PCTMO      S N R      8 2.536E-02     27.416
 9   .102426   .260163 2.470E+07 5.900E-01   474.233 0.          8.797E-01 0.         U-3PCTMO      S N R      9 2.370E-02     29.913
10   .111757   .283864 2.470E+07 5.900E-01   433.205 0.          8.797E-01 0.         U-3PCTMO      S N R     10 2.215E-02     32.247
11   .120476   .306015 2.470E+07 5.900E-01   394.860 0.          8.797E-01 0.         U-3PCTMO      S N R     11 2.070E-02     34.429
12   .128629   .326717 2.470E+07 5.900E-01   359.023 0.          8.797E-01 0.         U-3PCTMO      S N R     12 1.935E-02     36.468
13   .136246   .346066 2.470E+07 5.900E-01   325.530 0.          8.797E-01 0.         U-3PCTMO      S N R     13 1.808E-02     38.374
14   .143366   .364149 2.470E+07 5.900E-01   294.228 0.          8.797E-01 0.         U-3PCTMO      S N R     14 1.690E-02     40.155
15   .150019   .381049 2.470E+07 5.900E-01   264.972 0.          8.797E-01 0.         U-3PCTMO      S N R     15 1.580E-02     41.819
16   .156238   .396845 2.470E+07 5.900E-01   237.630 0.          8.797E-01 0.         U-3PCTMO      S N R     16 1.476E-02     43.375
17   .162050   .411607 2.470E+07 5.900E-01   212.076 0.          8.797E-01 0.         U-3PCTMO      S N R     17 1.380E-02     44.828
18   .167482   .425403 2.470E+07 5.900E-01   188.193 0.          8.797E-01 0.         U-3PCTMO      S N R     18 1.289E-02     46.187
19   .172558   .438298 2.470E+07 5.900E-01   165.873 0.          8.797E-01 0.         U-3PCTMO      S N R     19 1.205E-02     47.457
20   .177303   .450349 2.470E+07 5.900E-01   145.012 0.          8.797E-01 0.         U-3PCTMO      S N R     20 1.126E-02     48.644
21   .181737   .461612 2.470E+07 5.900E-01   125.515 0.          8.797E-01 0.         U-3PCTMO      S N R     21 1.053E-02     49.753
22   .185881   .472138 2.470E+07 5.900E-01   107.293 0.          8.797E-01 0.         U-3PCTMO      S N R     22 9.838E-03     50.790
23   .189754   .481976 2.470E+07 5.900E-01    90.263 0.          8.797E-01 0.         U-3PCTMO      S N R     23 9.195E-03     51.759
24   .193374   .491171 2.470E+07 5.900E-01    74.347 0.          8.797E-01 0.         U-3PCTMO      S N R     24 8.593E-03     52.664
25   .196757   .499764 2.470E+07 5.900E-01    59.472 0.          8.797E-01 0.         U-3PCTMO      S N R     25 8.031E-03     53.511
26   .199919   .507795 2.470E+07 5.900E-01    45.569 0.          8.797E-01 0.         U-3PCTMO      S N R     26 7.506E-03     54.302
27   .202874   .515301 2.470E+07 5.900E-01    32.576 0.          8.797E-01 0.         U-3PCTMO      S N R     27 7.015E-03     55.041
28   .205636   .522316 2.470E+07 5.900E-01    20.433 0.          8.797E-01 0.         U-3PCTMO      S N R     28 6.556E-03     55.732
29   .208218   .528872 2.470E+07 5.900E-01     9.083 0.          8.797E-01 0.         U-3PCTMO      S N R     29 6.128E-03     56.378
30   .210630   .535000 0.        5.900E-01     9.083 0.          8.797E-01 0.         U-3PCTMO      S L P     30 0.            56.690

      YIELD= 9.100E+08 SOUND SP= 3.383E+05 DENSITY= 1.786E+01 TENS STR=-1.938E+09 INTERFACE STRENGTH= 0.

 0   .210630   .535000 9.879E+06 2.360E-01     9.083 0.          7.982E-03 0.         NA-600C
31   .210630   .535000 9.879E+06 2.360E-01     8.942 0.          7.982E-03 0.         NA-600C       S H R     31 4.233E-03     56.386
32   .212297   .539233 9.879E+06 2.360E-01     8.800 0.          7.982E-03 0.         NA-600C       S N R     32 4.233E-03     56.394
33   .213963   .543467 9.879E+06 2.360E-01     8.658 0.          7.982E-03 0.         NA-600C       S N R     33 4.233E-03     56.402
34   .215630   .547700 0.        2.360E-01     8.658 0.          7.982E-03 0.         NA-600C       S L R     34 0.            56.406

      YIELD= 0.         SOUND SP= 1.990E+05 DENSITY= 8.080E-01 TENS STR=-9.996E+10 INTERFACE STRENGTH= 0.

 0   .215630   .547700 9.879E+06 2.360E-01     8.658 0.          4.982E-02 0.               ZIRC
35   .215630   .547700 9.879E+06 2.360E-01     6.357 0.          4.982E-02 0.               ZIRC    S H P     35 8.467E-03     56.533
36   .218963   .556167 9.879E+06 2.360E-01     4.057 0.          4.982E-02 0.               ZIRC    S N R     36 8.467E-03     56.664
37   .222297   .564633 9.879E+06 2.360E-01     1.757 0.          4.982E-02 0.               ZIRC    S N R     37 8.467E-03     56.794
38   .225630   .573100 0.        2.360E-01     1.757 0.          4.982E-02 0.               ZIRC    S L R     38 0.            56.860

      YIELD= 3.000E+09 SOUND SP= 4.567E+05 DENSITY= 6.550E+00 TENS STR=-9.957E+10 INTERFACE STRENGTH= 0.

 0   .225630   .573100 9.879E+06 2.360E-01     1.757 0.          8.367E-03 0.         NAK-100C
39   .225630   .573100 9.879E+06 2.360E-01     1.610 0.          8.367E-03 0.         NAK-100C      S H R     39 4.167E-03     56.803
40   .227270   .577267 9.879E+06 2.360E-01     1.464 0.          8.367E-03 0.         NAK-100C      S N R     40 4.167E-03     56.811
41   .228911   .581433 9.879E+06 2.360E-01     1.318 0.          8.367E-03 0.         NAK-100C      S N R     41 4.167E-03     56.819
42   .230551   .585600 9.879E+06 2.360E-01     1.171 0.          8.367E-03 0.         NAK-100C      S N R     42 4.167E-03     56.828
43   .232192   .589767 9.879E+06 2.360E-01     1.025 0.          8.367E-03 0.         NAK-100C      S N R     43 4.167E-03     56.836
44   .233832   .593933 9.879E+06 2.360E-01      .878 0.          8.367E-03 0.         NAK-100C      S N R     44 4.167E-03     56.844
45   .235472   .598100 9.879E+06 2.360E-01      .732 0.          8.367E-03 0.         NAK-100C      S N R     45 4.167E-03     56.853
46   .237113   .602267 9.879E+06 2.360E-01      .586 0.          8.367E-03 0.         NAK-100C      S N R     46 4.167E-03     56.861
47   .238753   .606433 9.879E+06 2.360E-01      .439 0.          8.367E-03 0.         NAK-100C      S N R     47 4.167E-03     56.869
48   .240394   .610600 9.879E+06 2.360E-01      .293 0.          8.367E-03 0.         NAK-100C      S N R     48 4.167E-03     56.878
49   .242034   .614767 9.879E+06 2.360E-01      .146 0.          8.367E-03 0.         NAK-100C      S N R     49 4.167E-03     56.886
50   .243675   .618933 9.879E+06 2.360E-01      .000 0.          8.367E-03 0.         NAK-100C      S N R     50 4.167E-03     56.894
51   .245315   .623100 0.        2.360E-01      .000 0.          8.367E-03 0.         NAK-100C      S L R     51 0.            56.899

      YIELD= 0.         SOUND SP= 2.501E+05 DENSITY= 8.470E-01 TENS STR=-9.996E+10 INTERFACE STRENGTH=-1.000E+11
```

FIGURE 6.3   SAMPLE DEPOS LISTING OF THE CELL LAYOUT FOR A RADIATION PROBLEM

```
OUTPUT FROM PRESCR
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)
  SD1( 25)    Y( 20)    Y( 21)    Y( 22)    Y( 23)    Y( 24)    Y( 25)  NEM( 20)  NEM( 21)  NEM( 22)  NEM( 23)  NEM( 24)
  NEM( 25)  NET( 20)  NET( 21)  NET( 22)  NET( 23)  NET( 24)  NET( 25)
S-INT(  1)S-INT(  2)S-INT(  0)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(  8)COM 7( 10)COM 7( 13)
COM 7( 17)COM12(  1)COM12(  3)COM12(  3)COM12(  8)COM12( 10)COM12( 13)COM12( 17)
```

FIGURE 6.4 SAMPLE LISTING OF OUTPUT FROM PRESCR SHOWING VARIABLES FOR WHICH HISTORICAL LISTINGS WILL BE MADE

```
DATE =  77/02/17. IDENT FR 4 FRAGMTG RND OF HF-1 TO SIMULATE CROWES TESTS 1 AND 2.

  TIME EDIT NO. 30 AT N =  300, TIME = 5.03722E-05 SECS, JSTAR =   22, CALC TIME IS    48.348 SECS, DTNH = 1.189E-07 SECS

    J        X         U       RHL        PHL        SHL        EHL     DHL        CHL  COND                   MUM      FBURN       NET
CELL        CM    CM/SEC  DYN/CM2    DYN/CM2    DYN/CM2       ERGS  GM/CM3     CM/SEC                        TAPS
   1 0.000000        0. 1.162E+09 1.162E+09 1.162E+09 1.980E+09   .3187 1.0177E+05 S S B CCMPB      0.         1.000E+00  0.
   2  .885987    14551. 1.305E+09 1.305E+09 1.305E+09 2.123E+09   .3339 1.0538E+05 S N B CCMPB      1.817E+03  1.000E+00  0.
   3 1.741600    30913. 1.581E+09 1.581E+09 1.581E+09 2.410E+09   .3563 1.1228E+05 S N B OCMPB      1.884E+04  1.000E+00  0.
   4 2.558029    41600. 1.899E+09 1.899E+09 1.899E+09 2.717E+09   .3796 1.1922E+05 S N B CCMPB      6.411E+04  1.000E+00  0.
   5 3.340054    51712. 1.834E+09 1.834E+09 1.834E+09 2.655E+09   .3752 1.1785E+05 S N B CCMPB      1.456E+05  1.000E+00  0.
   6 4.141794    59596. 1.613E+09 1.613E+09 1.613E+09 2.443E+09   .3587 1.1303E+05 S N B OCMPB      2.707E+05  1.000E+00  0.
   7 4.981754    62292. 1.793E+09 1.793E+09 1.793E+09 2.618E+09   .3720 1.1703E+05 S N B CCMPB      4.381E+05  1.000E+00  0.
   8 5.792237    65167. 1.813E+09 1.813E+09 1.813E+09 2.639E+09   .3732 1.1748E+05 S N B OCMPB      6.449E+05  1.000E+00  0.
   9 6.603248    80913. 1.712E+09 1.712E+09 1.712E+09 2.548E+09   .3650 1.1545E+05 S N B CCMPB      9.185E+05  1.000E+00  0.
  10 7.433143    95163. 1.653E+09 1.653E+09 1.653E+09 2.523E+09   .3558 1.1488E+05 S N B OCMPB      1.292E+06  1.000E+00  0.
  11 8.282992   109313. 0.        0.        0.        4.469E+10  1.7200 7.9500E+05 S L B OCMPB      1.777E+06  1.000E+00  0.
    J        X         U       RHL        PHL        SHL        EHL     DHL        CHL  COND                   MUM  Y/NEM/RVV SD/NET/ENV
CELL        CM    CM/SEC  DYN/CM2    DYN/CM2    DYN/CM2       ERGS  GM/CM3     CM/SEC                        TAPS
  12 8.282992   109313. 1.489E+09 1.489E+09 1.489E+09 6.529E+08  7.8144 5.8440E+05 S R Y HF-1 SH B 1.777E+06  1.010E+00  1.747E+01
  13 8.388019   107964. 1.319E+09 1.319E+09 1.319E+09 6.376E+08  7.8146 5.8440E+05 S N Y HF-1 SH B 3.264E+06  1.014E+00  1.677E+01
  14 8.497585   106580. 1.095E+09 1.882E+09 1.059E+09 6.288E+08  7.8141 5.8440E+05 S N V HF-1 SH B 4.814E+06  9.592E-01  1.538E+01
  15 8.611523   105094. 1.020E+09 1.627E+09 9.481E+08 6.226E+08  7.8141 5.8440E+05 S N U HF-1 SH B 6.427E+06  9.181E-01  1.586E+01
  16 8.729654   103524. 1.260E+09 1.141E+09 1.141E+09 6.276E+08  7.8143 5.8440E+05 S N Y HF-1 SH B 8.097E+06  1.006E+00  1.499E+01
  17 8.851807   101873. 1.882E+09 1.595E+09 1.595E+09 6.260E+08  7.8167 5.8440E+05 S N Y HF-1 SH B 9.820E+06  1.003E+00  1.484E+01
  18 8.977780   100105. 1.899E+09 1.797E+09 1.797E+09 6.230E+08  7.8179 5.8440E+05 S N Y HF-1 SH B 1.159E+07  1.057E+00  1.365E+01
  19 9.107458    98673. 6.646E+08 0.        0.        6.141E+08  7.5276 5.8440E+05 S N Y HF-1 SH B 1.342E+07  1.057E+00  1.886E+01
  20 9.245722    96152. 1.508E+08 5.628E+08 1.508E+08 5.916E+08  7.8121 5.8440E+05 S N P HF-1 SH B 1.528E+07  6.731E-01  8.928E+00
  21 9.382373    94742. 1.288E+08-2.756E+09 0.        5.644E+08  7.8013 5.8440E+05 S N P HF-1 SH B 1.718E+07  6.978E-01  8.415E+00
  22 9.522458    93185. 0.        0.        0.        0.         7.8500 5.8440E+05 S L B HF-1 SH B 1.912E+07  6.867E+09  0.
```

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

```
DATE =  77/02/17. IDENT FR 4 FRAGMTG RND OF HF-1 TO SIMULATE CROWES TESTS 1 AND 2.
SCRIBE NO.  1 USUAL UNITS ARE DYN, CM, SEC, GRAM, EXCEPT TIME IN MICROSEC, DTNH IN NANOSEC
   N     TIME      DTNH    DELTIM  JTS S-INT( 11)    S1( 12)    S1( 13)    S1( 14)    S1( 15)    S1( 16)    S1( 17)
  1.     .010     9.585      .674  13.  7.076E+10 -9.314E-03  0.         0.         0.         0.         0.
  2.     .021    11.502      .030  12.  7.082E+10  1.411E+08 -9.314E-03  0.         0.         0.         0.
  3.     .035    13.802      .034  12.  7.100E+10  5.116E+08  9.008E+05 -9.314E-03  0.         0.         0.
  4.     .051    16.563      .037  12.  7.135E+10  1.241E+09  5.225E+06  6.855E+03 -7.337E-03  0.         0.
  5.     .071    19.875      .038  12.  7.196E+10  2.519E+09  1.921E+07  5.786E+04  1.383E+01  0.         0.
  6.     .095    23.850      .041  12.  7.296E+10  4.613E+09  5.725E+07  3.001E+05  2.000E+02 -7.337E-03  0.
  7.     .124    28.620      .043  12.  7.452E+10  7.899E+09  1.510E+08  1.239E+06  1.688E+03  7.356E-01  0.
  8.     .158    34.344      .043  12.  7.687E+10  1.288E+10  3.678E+08  4.463E+06  2.308E+04  1.200E+01  0.
  9.     .199    41.213      .049  12.  8.024E+10  2.004E+10  8.455E+08  1.479E+07  1.372E+05  2.275E+02 -9.314E-03
 10.     .249    49.456      .050  12.  8.366E+10  2.753E+10  1.854E+09  4.628E+07  6.486E+05  2.211E+03  2.666E+00
 11.     .308    59.347      .191  12.  8.794E+10  3.703E+10  3.854E+09  1.387E+08  2.780E+06  1.629E+04  4.612E+01
 12.     .379    71.216      .056  12.  9.359E+10  4.968E+10  7.615E+09  3.979E+08  1.118E+07  1.621E+05  5.237E+02
 13.     .465    85.460      .056  12.  1.009E+11  6.620E+10  1.436E+10  1.092E+09  4.322E+07  1.015E+06  7.169E+03
 14.     .567   102.551      .062  12.  1.100E+11  8.665E+10  2.351E+10  2.866E+09  1.611E+08  5.508E+06  6.947E+04
 15.     .690   123.062      .063  12.  1.197E+11  1.092E+11  3.542E+10  6.970E+09  5.771E+08  2.765E+07  7.720E+05
 16.     .838   147.674      .069  13.  1.276E+11  1.283E+11  5.400E+10  1.551E+10  1.940E+09  1.337E+08  5.782E+06
 17.    1.015   177.209      .075  13.  1.296E+11  1.357E+11  8.074E+10  2.696E+10  5.985E+09  6.116E+08  3.796E+07
 18.    1.195   179.582      .077  13.  1.247E+11  1.288E+11  1.076E+11  4.483E+10  1.479E+10  2.409E+09  2.182E+08
 19.    1.382   186.796      .079  14.  1.177E+11  1.169E+11  1.232E+11  6.956E+10  2.531E+10  6.983E+09  9.471E+08
 20.    1.569   187.279      .079  14.  1.138E+11  1.108E+11  1.242E+11  9.519E+10  4.048E+10  1.528E+10  3.160E+09
 21.    1.758   189.430      .258  15.  1.135E+11  1.115E+11  1.174E+11  1.123E+11  6.204E+10  2.428E+10  7.981E+09
 22.    1.952   193.410      .080  15.  1.152E+11  1.156E+11  1.099E+11  1.177E+11  8.548E+10  3.728E+10  1.574E+10
 23.    2.145   192.754      .083  16.  1.154E+11  1.164E+11  1.079E+11  1.146E+11  1.030E+11  5.636E+10  2.353E+10
 24.    2.343   198.306      .083  16.  1.138E+11  1.134E+11  1.106E+11  1.088E+11  1.110E+11  7.788E+10  3.494E+10
 25.    2.539   195.925      .083  17.  1.123E+11  1.103E+11  1.128E+11  1.056E+11  1.107E+11  9.526E+10  5.211E+10
 26.    2.741   202.260      .089  17.  1.123E+11  1.104E+11  1.115E+11  1.065E+11  1.070E+11  1.047E+11  7.191E+10
 27.    2.940   198.894      .086  17.  1.134E+11  1.126E+11  1.082E+11  1.089E+11  1.038E+11  1.066E+11  8.876E+10
 28.    3.144   204.391      .085  18.  1.136E+11  1.135E+11  1.072E+11  1.089E+11  1.035E+11  1.045E+11  9.905E+10
 29.    3.346   201.656      .086  18.  1.127E+11  1.121E+11  1.090E+11  1.065E+11  1.052E+11  1.019E+11  1.025E+11
 30.    3.552   205.712      .085  19.  1.116E+11  1.105E+11  1.104E+11  1.050E+11  1.058E+11  1.011E+11  1.018E+11
 31.    3.756   204.360      .267  19.  1.111E+11  1.103E+11  1.097E+11  1.058E+11  1.044E+11  1.021E+11  9.988E+10
 32.    3.964   207.900      .088  20.  1.110E+11  1.107E+11  1.079E+11  1.072E+11  1.030E+11  1.029E+11  9.883E+10
 33.    4.175   211.485      .084  20.  1.104E+11  1.103E+11  1.071E+11  1.070E+11  1.032E+11  1.021E+11  9.897E+10
 34.    4.390   214.613      .085  13.  1.093E+11  1.088E+11  1.075E+11  1.054E+11  1.043E+11  1.007E+11  9.906E+10
 35.    4.613   223.161      .083  21.  1.083E+11  1.074E+11  1.074E+11  1.044E+11  1.042E+11  1.001E+11  9.742E+10
 36.    4.845   232.241      .081  12.  1.078E+11  1.072E+11  1.061E+11  1.044E+11  1.025E+11  1.000E+11  9.377E+10
 37.    5.124   278.689      .079  12.  1.076E+11  1.073E+11  1.043E+11  1.043E+11  1.004E+11  9.804E+10  8.893E+10
 38.    5.403   279.226      .083  12.  1.068E+11  1.061E+11  1.044E+11  1.016E+11  9.910E+10  9.160E+10  8.234E+10
 39.    5.682   278.757      .083  12.  1.060E+11  1.052E+11  1.036E+11  9.922E+10  9.448E+10  8.408E+10  6.902E+10
 40.    5.960   278.217      .084  21.  1.056E+11  1.049E+11  1.010E+11  9.656E+10  8.618E+10  7.337E+10  5.245E+10
 41.    6.194   234.000      .265  21.  1.046E+11  1.035E+11  9.829E+10  9.043E+10  7.707E+10  5.854E+10  4.049E+10
 42.    6.431   236.616      .084  12.  1.031E+11  1.013E+11  9.452E+10  8.220E+10  6.621E+10  4.630E+10  3.023E+10
 43.    6.708   276.958      .084  12.  1.011E+11  9.813E+10  8.747E+10  7.135E+10  5.287E+10  3.587E+10  1.922E+10
 44.    6.985   276.557      .084  12.  9.801E+10  9.330E+10  7.621E+10  5.846E+10  4.062E+10  2.542E+10  9.509E+09
 45.    7.261   276.297      .084  12.  9.359E+10  8.610E+10  6.484E+10  4.634E+10  3.072E+10  1.513E+10  4.857E+09
 46.    7.537   276.194      .083  12.  8.902E+10  7.913E+10  5.574E+10  3.673E+10  2.121E+10  7.960E+09  6.044E+09
 47.    7.813   275.968      .084  12.  8.572E+10  7.529E+10  4.954E+10  2.947E+10  1.326E+10  6.852E+09  7.240E+09
 48.    8.088   275.420      .084  12.  8.319E+10  7.348E+10  4.640E+10  2.436E+10  1.078E+10  8.817E+09  5.025E+09
 49.    8.363   274.723      .084  21.  8.042E+10  7.143E+10  4.516E+10  2.355E+10  1.387E+10  8.564E+09  2.067E+09
 50.    8.597   234.145      .083  14.  7.749E+10  6.884E+10  4.530E+10  2.731E+10  1.697E+10  7.171E+09  8.775E+08
```

FIGURE 6.6 SAMPLE HISTORICAL LISTING OF VARIABLES FROM SCRIBE

PERIODIC PRINT

```
N=  300, JSTAR=  22, TIME= 5.037E-05, CALC TIME=   483.290 SECS, JTS=  14  DTNH= 1.189E-07  SMAX= 1.859E+09  JSMAX=   4
```

STOP CRITERION

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

N = 300, JCYCS = 300, TIME = 5.037E-05, TS = 8.000E-05, X(JSMAX) = 2.558E+00, CKS = 3.000E+01, LSUB(7)=  0, DTNH = 1.189E-07
```

SHEAR2 OUTPUT

```
K=   21 J=   21 IH3=   16 ROT= 0.         EN=  .841E+01 TAU=  .698E+00 EP=  .745E+00
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    .110E+01    .151E+01    .189E+01    .225E+01
NG=    5
CN=  .159E+01    .102E+01    .576E+00    .276E+00    .107E+00    .307E-01    .591E-02    .656E-03
CL=  .103E-01    .247E-01    .449E-01    .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    .684E-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       .256        .224        .219
TAUZ=  0.000000  0.000000  0.000000   .697120    .000435    .000293
```

BFRACT3 OUTPUT

```
FRAG  N=  4.008E+01  1.162E+01  1.162E+01  5.284E+01  5.284E+01  SUM=  1.696E+02  FF=  7.195E-01, CYCLE=   13
J= 1 C1=  2.945E-01  2.923E-01  2.923E-01  2.931E-01  2.931E-01  FRFR=  5.229E-02  4.174E-01  4.174E-01  2.349E-01
K= 2 CB=  5.332E-01  5.332E-01  5.332E-01  5.332E-01  5.332E-01  TAUZ=  2.829E-01  8.006E-02  8.006E-02  3.654E-01  3.654E-01
    VVA=  1.303E-02  3.452E-03  3.452E-03  1.611E-02  1.611E-02
```

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 TIME $\geq$ TS, X(JSMAX) $\geq$ CKS, LSUB(7) = 1, and DTNH < 1.E - 12. LSUB(7) is set to 1 in HSTRESS and FMELT to trigger an error stop.

Several material property subroutines provide regular listings in the cycle just preceding an EDIT. The samples in Figure 6.7 are from SHEAR2 and BFRACT3. EXPLODE also prints a line whenever the detonation is completed at a cell. If the iterations do not converge in CAP1, BFRACT3, REBAR, PEST, EQSTPF, TSQE, BECOM, or DFRACT, an error message and some information about the cause and location of failure is given. 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

$$\sigma_a \propto (h\nu)^{-3} \tag{A.1}$$

where σ_a is the mass absorption coefficient and $h\nu$ is the photon energy (ν is frequency and h is Planck's constant). In standard tables, such as those of McMasters et. al.[41] and Fisher and Wiehe,[42] the absorption coefficient is expanded in the following form between edges:

$$\ln \sigma_a = A_0 + A_1 w + A_2 w^2 + A_3 w^3 \tag{A.2}$$

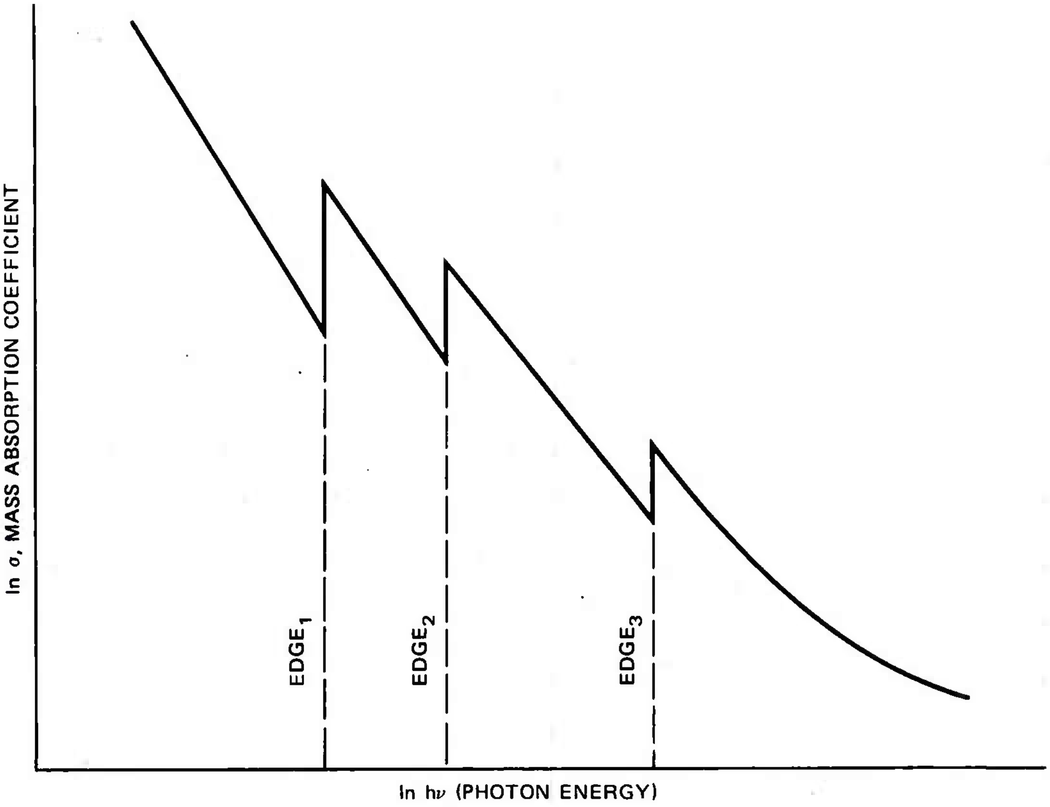
where σ_a the mass absorption coefficient, barn/atom

w = $\ln(h\nu)$
$h\nu$ = photon energy, keV
A_0 ... 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 ΔX is

$$\frac{I}{I_0} = \exp(-\mu_a \Delta X) \tag{A.3}$$



MA-6802-4

FIGURE A.1 TYPICAL VARIATION OF PHOTOELECTRIC MASS ABSORPTION COEFFICIENT WITH PHOTON ENERGY where μ_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 μ_a is related to σ_a as follows.

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

where ρ = 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

σ_a = mass absorption coefficient, barn/atom.

With the coefficients A_0, ... 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.

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

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 information 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

$$\mu_a = 0.602252 \sum_{n=1}^{N_c} \frac{n}{A_{wn}} \sum_{i=0}^{3} A_{in} \left[\ln(h\nu)\right]^i \qquad (A.6)$$

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 ρ_n is the weight fraction of the nth constituent times the composite density. Hence

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

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 standarization 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

$$dW = \frac{15}{\pi^4}\left(\frac{\omega^3}{e^{\omega} - 1}\right) d\omega \qquad (A.8)$$

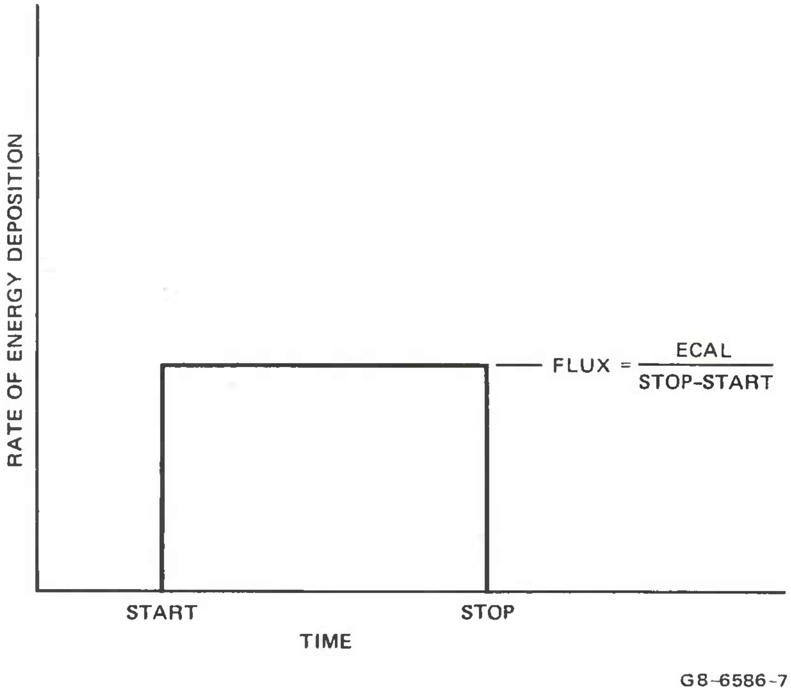


FIGURE A.2 HISTORY OF RADIATION SOURCES CONSIDERED IN THE PROGRAM

where

ω = $h\nu/kT$, a nondimensional quantity proportional to photon energy

h = Planck's constant

ν = frequency of the photons

$h\nu$ = photon energy, usually in keV

k = Boltzmann constant

T = Kelvin temperature

kT = 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 ΔW 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

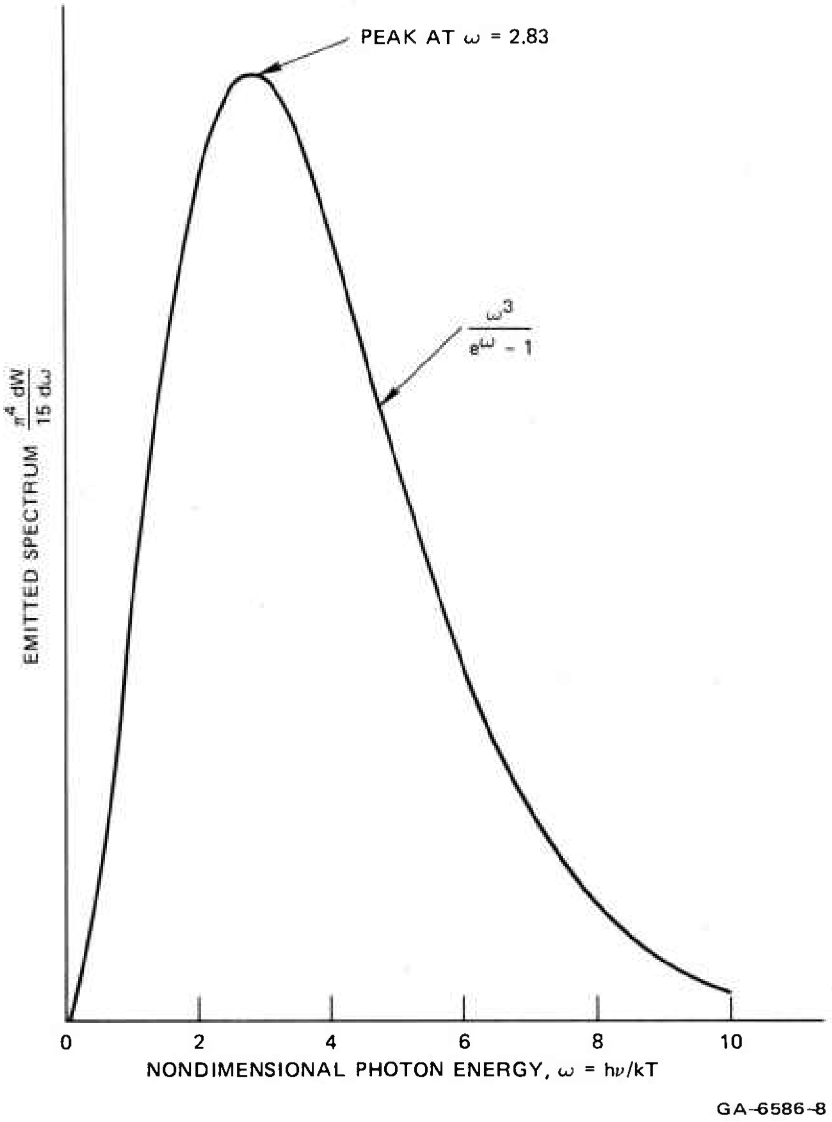


FIGURE A.3 EMITTANCE SPECTRUM FOR BLACK BODY

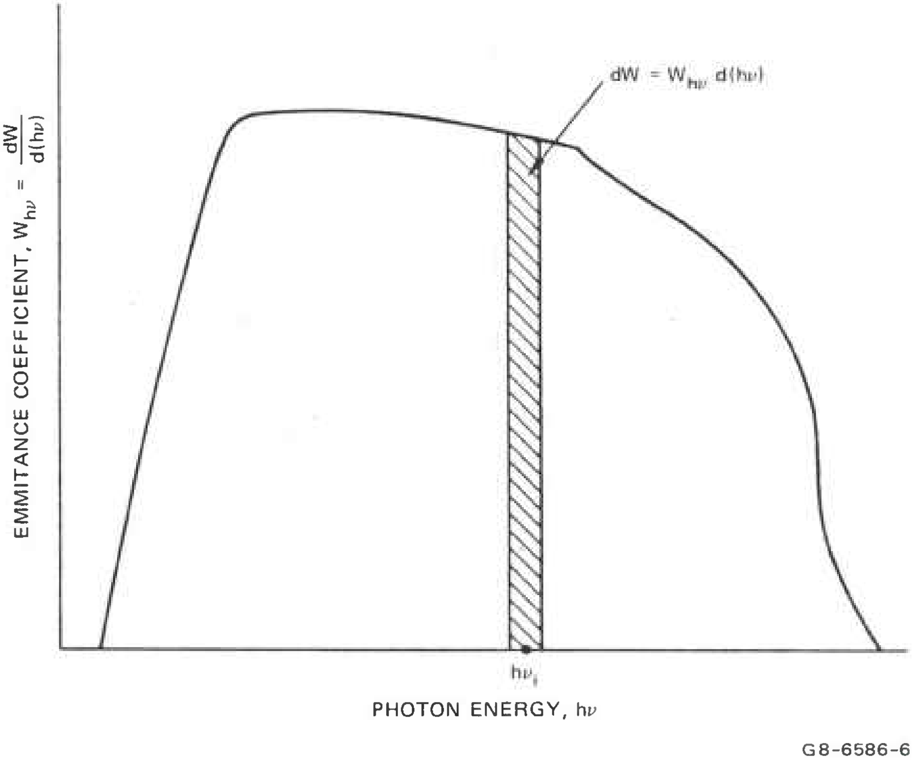


FIGURE A.4 SPECTRUM OF A RADIATION SOURCE

$$\Delta E = E_1 \left(1 - e^{-\mu_a \Delta X}\right) \tag{A.9}$$

where

E_1 = the amount of energy reaching the left face of the cell

ΔX = the thickness of the cell.

Because μ_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 μ_a for each abscissa, and the function of μ_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 jth cell in an increment of time Δt is given by

$$\Delta E_j = C_c \frac{E^n_{R,j} \; \Delta t}{Z_j \Delta T^n} \tag{A.10}$$

where

$E^n_{R,j}$ = the total energy in cal/cm^2 to be deposited in the jth cell from the nth source

C_c = a conversion factor, 4.186×10^7 erg/cal

Z_j = the mass of the jth cell, g/cm^2

ΔT^n = the duration of the nth source.

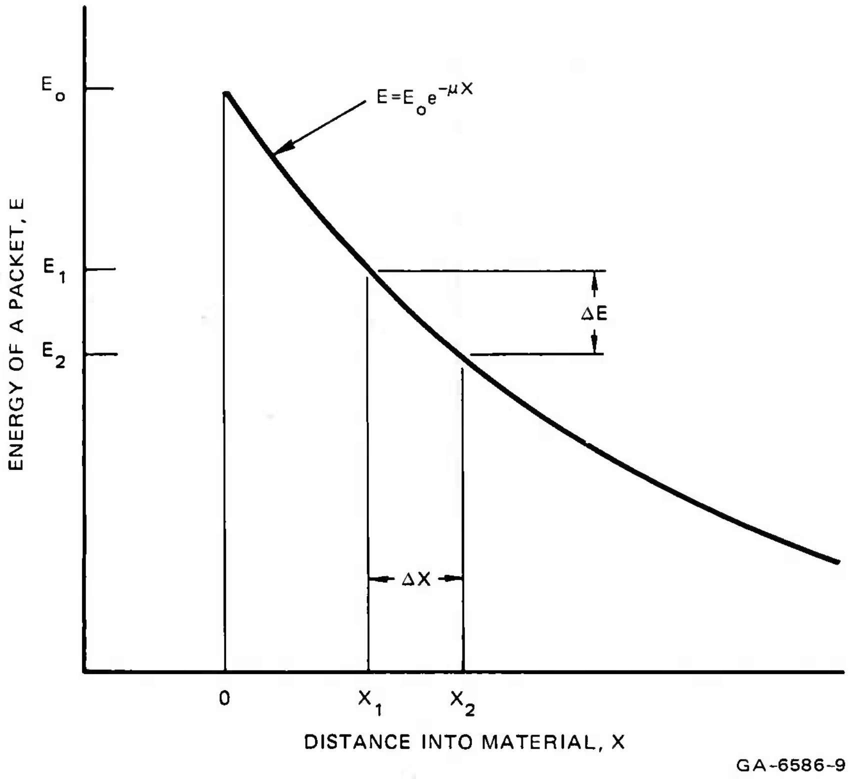


FIGURE A.5 PENETRATION OF ENERGY INTO A MATERIAL

Then the deposited energy ΔE_j is in erg/g. The array SS is defined to include all the constant quantities in Eq. (A.10), that is, all except Δt.

$$SS_j^n = \frac{C_c E_{R,j}^n}{Z_j \Delta T^n} \qquad (A.11)$$

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:

$$E = E_o e^{-\mu_a X} \qquad (A.12)$$

where

E_o = the incident energy

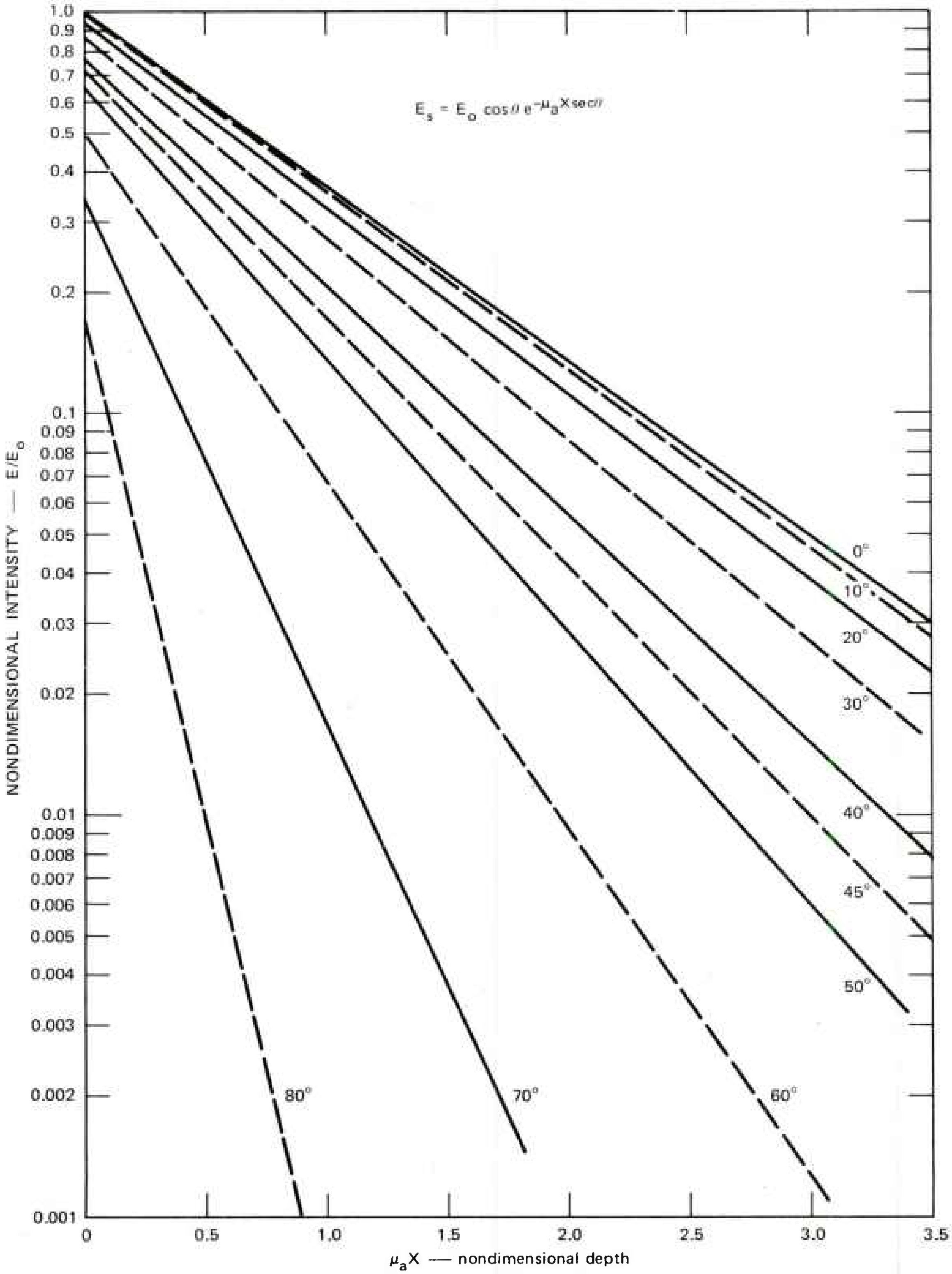
E = the intensity at any depth, X.

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

$$E_s = E_o \cos\theta \, e^{-\mu_a X \sec\theta} \qquad (A.13)$$

where θ 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 ΔX, ΔY, and ΔZ, the incident fluence is



MA-6802-5

FIGURE A.6 RADIATION INTENSITY AS A FUNCTION OF DEPTH FOR SEVERAL INCIDENT ANGLES

$$E_{si}\ \Delta Y\ \Delta Z = E_o \cos\theta\ e^{-\mu_a X_1 \sec\theta}\ \Delta Y \Delta Z$$

and the mass is

$$Z_j = \rho \Delta X \Delta Y \Delta Z$$

Therefore, the absorbed energy is

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

$$= \frac{E_o \cos\theta\ e^{-\mu_a \Delta X \sec\theta}}{Z_j} \qquad (A.14)$$

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.[42] 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/cm^2). 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

$$E_s = E_{s1}^{\xi_1} \cdot E_{s2}^{\xi_2} \cdot E_{s3}^{\xi_3} \tag{A.15}$$

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

$$E'_j = \frac{1}{X_{j+1} - X_j} \int_{X_j}^{X_{j+1}} E_s \, dX \tag{A.16}$$

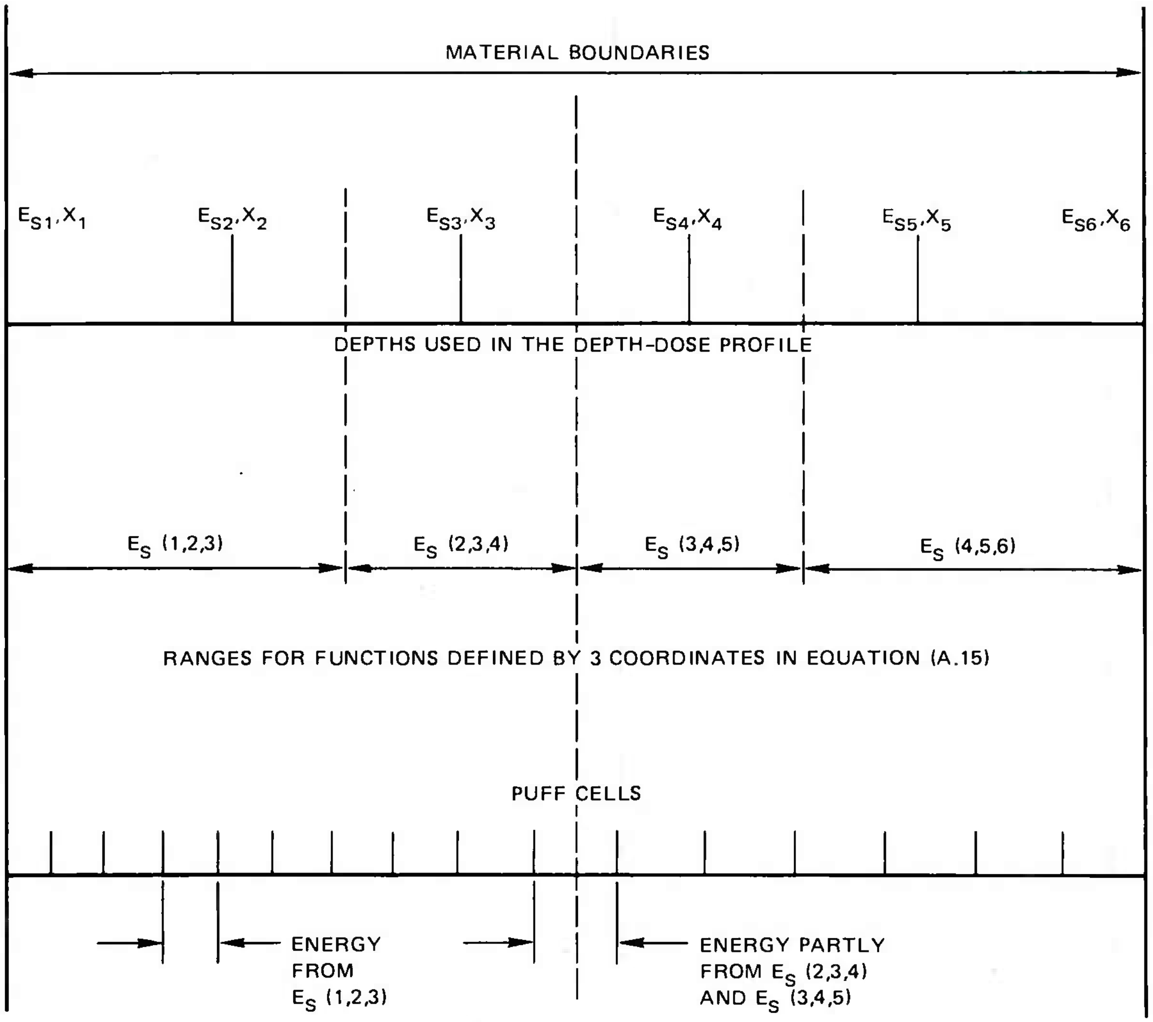
This integration is performed numerically using Simpson's rule.

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_{cal} is $E_j' \cdot E_{cal}$. Then the expression for computing values for the SS array is

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

The SS array is used in the function SSCALH to provide energy increments for each cell during the propagation calculations.



GA-8152-4A

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.[44] Then the types of detonation provided in PUFF, the input required, and the algebra of the code calculations are described.

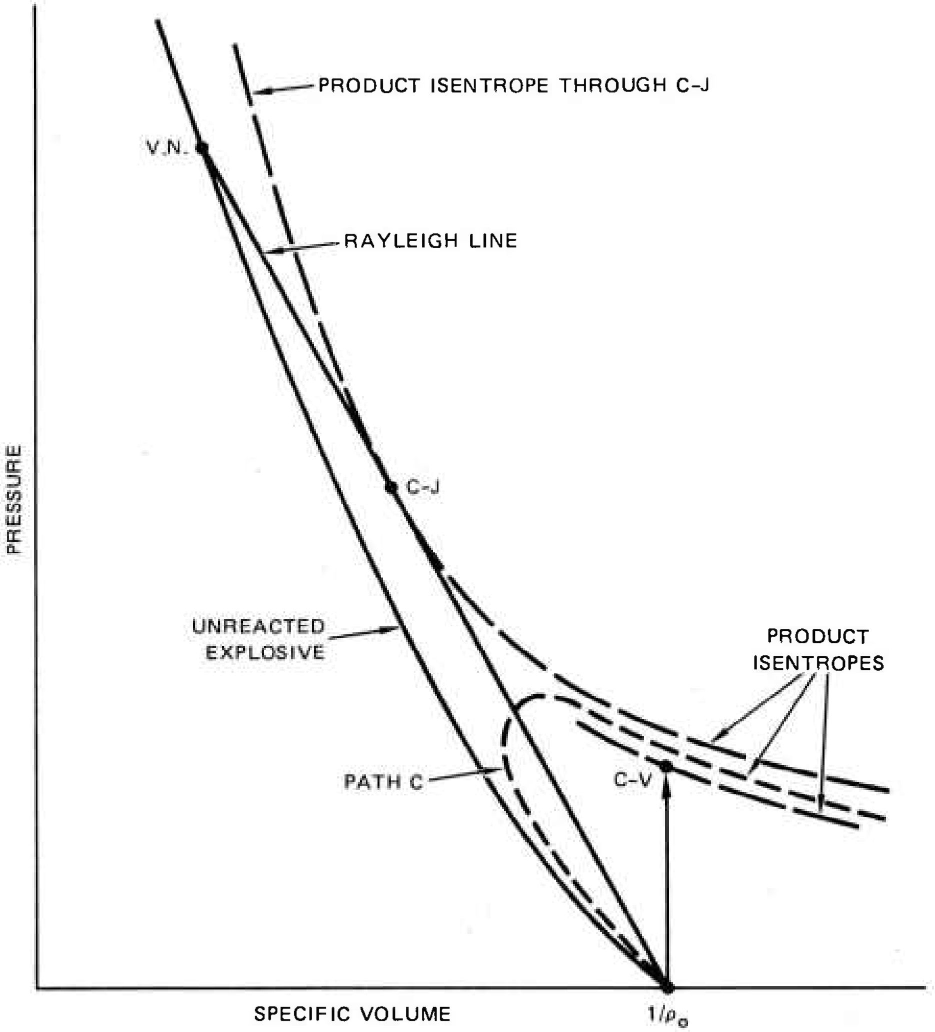
Background on Detonation Processes

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_{CJ}, V_{CJ}, and E_{CJ}. If the product gases are assumed to follow a polytropic gas equation of state, that is,

$$PV^{\gamma} = \text{constant} \tag{B.1}$$

then relations for the detonation velocity (D_x), P_{CJ}, V_{CJ}, E_{CJ}, and the particle velocity (u_{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.

$$D_x = \sqrt{2Q_x(\gamma + 1)\ (\gamma - 1)} \tag{B.2}$$



MA-6802-13

FIGURE B.1 PRESSURE-VOLUME PATHS FOLLOWED IN DETONATION PROCESS

$$P_{CJ} = 2Q_x(\gamma - 1)\,\rho_o \tag{B.3}$$

$$V_{CJ} = \frac{\gamma}{\rho_o(\gamma + 1)} \tag{B.4}$$

$$E_{CJ} = \frac{2Q_x\gamma}{\gamma + 1} \tag{B.5}$$

$$u_{CJ} = \sqrt{\frac{2Q_x(\gamma - 1)}{\gamma + 1}} \tag{B.6}$$

where Q_x = the energy of the explosive

ρ_o = the initial density.

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

$$\gamma = \Gamma + 1 \tag{B.7}$$

For many common explosives, γ 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 γ 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.

$$F_B = 1 - \frac{\left|\bar{X} - X_D\right|}{b\Delta X} \tag{B.8}$$

where $\bar{X}$ = the cell midpoint

ΔX = the cell length in the direction of propagation.

From eq. (B.8) it appears that the cell midpoint must be within a distance $b\Delta X$ 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

$$P = P_{CJ} F_B \tag{B.9}$$

$$\rho = \frac{\rho_o}{1 + F_B(V_{CJ}\rho_o - 1)} \tag{B.10}$$

$$E = Q_x + (E_{CJ} - Q_x)FB \tag{B.11}$$

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.

$$t_B = \frac{\left|\bar{X} - X_D\right| - b\,\Delta\,X}{D_X} \tag{B.12}$$

The fraction detonated is then

$$F_B = \frac{(t - t_B)\,D_X}{b\Delta\,X} \tag{B.13}$$

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

```
   -DATA- IN COLUMNS 2-5 INDICATES A DATA BANK FOR TAPE 4

HEADING    SHOCKEY
NTEDT =             0 NJEDIT =             1 NREZON =             0 NALPHA =             1
JEDIT =       14    16    18    20    21    22    23    24    26    28
NEDIT =            10 JCYCS =            200 CKS =              10. TS =                1.
NMTRLS =            2 MATFL =              1 UZERU =    0.

1145 AL              RHOS =     2.784      CFP = 000 DPY = 503
EQST =     8.560E+11 0.         1.000E+11 2.1        2.1
BAUSCH =   3.000E-02 4.000E+10 4.000E+10 2.000E-01
YIELD =    4.130E+09 3.000E+11
VISC =     4.0       0.05      0.05
EMELT =    6.600E+09 5.000E+09 0.         1.0        0.25

1145 AL HOT          RHOS =     2.70452    CFP = 010 DPY = 003
EQST =     8.560E+11 0.         1.000E+11 2.1        2.1
DFR1 =    -.00625   -4.000E+09 1.000E-04 2.000E+12-3.000E+09 0.960E+09
YIELD =    2.000E+09 3.000E+11
VISC =     4.0       0.05      0.05
EMELT =    6.600E+09 5.000E+09 0.         1.0        0.25

ARMCO SH BAND        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
SH2        3.000E+01 .2000E+00 1.100E-02 3.000E-04 0.17       0.070E+00 0.070E+00
           1.4       3.000E-08 3.000E+08         6.         .2         .17 7.000E+09
NSIZE         0    0     0    8     8    8     0    0     0
MELT =     1.085E+10 6.460E+09 5.700E-01 7.000E-01 1.850E-01
YIELD =    2.000E+09 8.190E+11

HF-1 SH BAND         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
SH2        3.000E+01 .2000E+00 1.100E-02 1.000E-03 0.17       0.070E+00 0.070E+00
           1.4       3.000E-08 3.000E+08         6.         .2         .17 7.000E+09
NSIZE         0    0     0    8     8    8     0    0     0
YIELD =    1.030E+10 8.190E+11
MELT =     1.085E+10 1.000E+08 0.E0       9.975E-01 1.000E-01

PMMA-8K8 (BARKER)  RHOS =       1.184E+00 CFP = 000 DPY = 001
EQST =     7.000E+10 4.050E+11 1.000E+10 1.000E+00 2.500E-01 39640E+11
YIELD =    1.000E+06 1.950E+10 2.850E+09

HF-1                 RHOS =     7.85E0     CFP=  000 DPY=  002                NCON=    0
EQST=      1.589E+12 5.170E+12 7.360E+10 1.69E0     0.25E0     5.170E+13
YIELD =    1.030E+10 8.190E+11
MELT =     1.085E+10 1.000E+08 0.E0       9.975E-01 1.000E-01

LEAD (KOHN)          RHOS =     11.355     CFP = 000 DPY = 000
EQST =     5.008E+11 4.986E+11 9.155E+09 2.2        0.25       2.019E+12

SPEC Z12             NHNU =               95 (4(F10.5,F7.5,3X))
    .03571 .00000        .10714 .00000        .17857 .00000        .25000 .00000
    .32143 .00001        .39286 .00002        .46429 .00003        .53571 .00004
    .60714 .00006        .67857 .00009        .75000 .00012        .82143 .00016
    .89286 .00021        .96429 .00026       1.07143 .00093       1.21429 .00156
   1.35714 .00209       1.50000 .00254       1.64286 .00290       1.78571 .00316
   1.92857 .00333       2.08333 .00436       2.25000 .00508       2.41667 .00565
   2.58333 .00608       2.75000 .00635       2.91667 .00648       3.12500 .00993
   3.37500 .01053       3.62500 .01145       3.87500 .01269       4.12500 .01420
   4.37500 .01563       4.62500 .01684       4.87500 .01782       5.16667 .02577
   5.50000 .02789       5.83333 .02884       6.16667 .02976       6.50000 .03075
   6.83333 .03069       7.16667 .03068       7.50000 .03062       7.83333 .02941
   8.25000 .04213       8.75000 .03797       9.25000 .03188       9.75000 .02682
  10.25000 .02138      10.75000 .01842      11.25000 .01734      11.75000 .01616
  12.25000 .01546      12.75000 .01464      13.25000 .01379      13.75000 .01311
  14.50000 .02460      15.50000 .02280      16.50000 .02070      17.50000 .01870
  18.50000 .01720      19.50000 .01610      20.50000 .01530      21.50000 .01480
  22.50000 .01450      23.50000 .01400      24.50000 .01160      25.50000 .00680
  26.50000 .00440      27.50000 .00580      28.50000 .00720      29.50000 .00730
  30.50000 .00680      31.50000 .00630      32.50000 .00590      33.50000 .00550
  34.50000 .00510      35.50000 .00480      36.50000 .00440      37.50000 .00410
  38.50000 .00370      39.50000 .00340      40.50000 .00320      41.50000 .00290
  42.50000 .00270      43.50000 .00250      44.50000 .00230      45.50000 .00210
  46.50000 .00200      47.50000 .00180      48.50000 .00170      49.50000 .00160
  51.25000 .00284      53.75000 .00376      57.50000 .00490
7/8/9
```

FIGURE C.1 DATA BANK CONTAINING GENERAL RUNNING INFORMATION, MATERIAL PROPERTY DATA, AND A SPECTRUM (ON TAPE 4)

```
ABS DATA
***   ABSORPTION EDGE AND COEFFICIENT DATA BORROWED MAINLY FROM THE FSCATT   ***
***   CODE OF S.S.S.   THE DATA ARE FROM THE LLL COMPILATION OF X-RAY CROSS  ***
***   SECTIONS, BY W. H. MCMASTERS, ET.AL., SECT.2, REV. 1, MAY, 1959       ***
*
***   VALUE OF COEFFICIENTS USED IN FIT OF CROSS SECTION DATA EQUATION -     ***
***        LN(SIGMA/SIGNAO)=A(0,I)+A(1,I)*X+A(2,I)*X**2+A(3,I)*X**3          ***
***   WHERE X=LN(HNU) SITH HNU IN KEV, SIGMAO=1(BARN/ATOM). SUBSCRIPT -1-    ***
***   REFERS TO THE FIT PAST THE ITH EDGE.                                   ***

HYDROGEN  X-RAY ABS NOE =                 1 ATWT =    1.008                 1 H
EDGE1        1.00000E  0                                                     1 H
COEF1        2.44950E  0,-3.34932E  0,-4.72054E -2, 7.10529E -3              1 H

HELIUM    X-RAY ABS NOE =                 1 ATWT      4.0026                2 HE
EDGE1        1.00000E  0                                                     2 HE
COEF1        6.06498E  0,-3.29055E  0,-1.07282E -1, 1.44502E -2              2 HE

LITHIUM   X-RAY ABS NOE =                 1 ATWT =    6.9390                3 LI
EDGE1        1.00000E  0                                                     3 LI
COEF1        7.75366E  0,-2.81798E  0,-2.41741E -1, 2.62541E -2              3 LI

BERYLLIUM X-RAY ABS NOE =                 1 ATWT =    9.0120                4 BE
EDGE1        1.00000E  0                                                     4 BE
COEF1        9.04503E  0,-2.83490E  0,-2.09990E -1, 2.29488E -2              4 BE

BORON     X-RAY ABS NOE =                 1 ATWT =    10.81                 5 B
EDGE1        1.00000E  0                                                     5 B
COEF1        9.95057E  0,-2.74173E  0,-2.15138E -1, 2.27845E -2              5 B

TITANIUM  X-RAY ABS NOE =                 2 ATWT =   47.90                 22 TI
EDGE1        1.00000E  0, 4.96500E  0                                       22 TI
COEF1        1.31074E  1,-2.53681E  0,-9.37662E -2,-8.07696E -4             22 TI
COEF2        1.43509E  1,-1.66361E  0,-3.31403E -1, 2.61935E -2             22 TI

VANADIUM  X-RAY ABS NOE =                 2 ATWT =   50.94                 23 V
EDGE1        1.00000E  0, 5.46500E  0                                       23 V
COEF1        1.32515E  1,-2.49745E  0,-1.06643E -1, 7.70206E -5             23 V
COEF2        1.47598E  1,-1.88849E  0,-2.71904E -1, 2.15824E -2             23 V

CHROMIUM  X-RAY ABS NOE =                 2 ATWT =   52.00                 24 CR
EDGE1        1.00000E  0, 5.98900E  0                                       24 CR
COEF1        1.34235E  1,-2.51606E  0,-1.01138E -1,-2.36898E -4             24 CR
COEF2        1.48015E  1,-1.82384E  0,-2.79236E -1, 2.17419E -2             24 CR

MANGANESE X-RAY ABS NOE =                 2 ATWT =   54.94                 25 MN
EDGE1        1.00000E  0, 6.54000E  0                                       25 MN
COEF1        1.35761E  1,-2.49626E  0,-1.07826E -1, 6.28831E -4             25 MN
COEF2        1.48969E  1,-1.79894E  0,-2.83640E -1, 2.22096E -2             25 MN

IRON      X-RAY ABS NOE =                 2 ATWT =   55.85                 26 FE
EDGE1        1.00000E  0, 7.11200E  0                                       26 FE
COEF1        1.36697E  1,-2.39272E  0,-1.36795E -1,-2.37212E -4             26 FE
COEF2        1.43458E  1,-1.23512E  0,-4.18728E -1, 3.21614E -2             26 FE

NICKEL    X-RAY ABS NOE =                 3 ATWT =    58.71                28 NI
EDGE1        1.00000E  0, 1.01200E  0, 8.33300E  0                          28 NI
COEF1        1.38363E  1,-2.47740E  0,   .00000E  0,   .00000E  0           28 NI
COEF2        1.39849E  1,-2.48097E  0,-8.88292E -2, 3.18989E -5             28 NI
COEF3        1.42375E  1,-9.66762E -1,-4.78299E -1, 3.66306E -2             28 NI

COPPER    X-RAY ABS NOE =                 3 ATWT =   63.54                 29 CU
EDGE1        1.00000E  0, 1.10000E  0, 8.97900E  0                          29 CU
COEF1        1.40954E  1,-2.59039E  0,   .00000E  0,   .00000E  0           29 CU
COEF2        1.42443E  1,-2.58831E  0,-6.51996E -2,-4.13025E -4             29 CU
COEF3        1.45807E  1,-1.18359E  0,-4.13899E -1, 3.12129E -2             29 CU

ZINC      X-RAY ABS NOE =                 5 ATWT =   65.37                 30 ZN
EDGE1        1.00000E  0, 1.02100E  0, 1.04400E  0, 1.19600E  0             30 ZN
EDGE2        9.65900E  0                                                    30 ZN
COEF1        1.20599E  0,-1.12290E  0,   .00000E  0,   .00000E  0           30 ZN
COEF2        1.38301E  1,-2.62547E  0,   .00000E  0,   .00000E  0           30 ZN
COEF3        1.41741E  1,-2.63124E  0,   .00000E  0,   .00000E  0           30 ZN
COEF4        1.43226E  1,-2.62555E  0,-2.50198E -2,-3.53392E -4             30 ZN
COEF5        1.44132E  1,-9.34286E -1,-4.77048E -1, 3.62589E -2             30 ZN
```

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,

```
XIDENT 4409-1    DUCTILE FRACTURE IN HOT 1145 AL UNDER IMPACT            SHOCKEY
 EXTRA
 $NLIST UZERO=1.46E4$
7/8/9
T1145 AL
T1145 AL HOT
 NLAYERS =          2 JMAT =        1     2
 NZONES= 1         10 CELLS IN  0.313E0    4409-1
 NZONES= 1         20 CELLS IN  6.350E-01 CM
 EXTRA
 $NLIST EHL(12)=20*4.46E9.RHOS(2)=2.784 $
7/8/9
```

FIGURE C.3 INPUT DECK FOR HOT ALUMINUM IMPACT CALCULATION

```
                   ****    5RI PUFF 8    ****

  DATE =  78/03/17.
IDENT 4409-1   -1- DAMAGE                                              SHOCKEY   IND=X , IN= 5
NTEDT =              0 NJEDIT =            I NREZON =          0 NALPHA =              1 IND=  , IN= 4
JEDIT =      I4   16   18   20   21   22   23   24   26   28                          IND=  , IN= 4
NEDIT =             10 JCYCS =             0 CK5 =      1.000E+01 TS =      1.000E+00 IND=  , IN= 4 , , , CM,  SEC
NMTRLS =             2 MATFL =             I UZERD =   0.                   ********** IND=  , IN= 4 , , , CM/SEC
        NSCRb INDICATORS  -  RAD. PLOTS  I-E, 2-T, 3-P, 4- ,         SCRIBE HISTORIES  I-6S, 2-I2S, 3-D5,Y, 4-R,V, 5-U,I, 6-NEM,NET
EXTRA                                                                           INPUT FROM -EXTRA- ROUTINE
$NLIST UZERO=1.46E4$                                                            INPUT FRDM -EXTRA- ROUTINE
EOF ENCOUNTERED BY EXTRA

1145 AL                      -0.            ***       ***         -0          -0 IND=T , IN= 5 G/CM3
1145 AL            RHOS =    2.784E+00 CFP = 000 DPY = 503        -0          -0 IND=  , IN= 4 G/CM3
EQST =    8.560E+II 0.        1.000E+II 2.100E+00 2.100E+00-0.        -0.        IND=  , IN= 4 DYN/CM2,=. ERG/G, , , DYN/CM2, ERG/G
BAUSCH =  3.000E-02 4.000E+10 4.000E+I0 2.000E-0I-0.         -0.      -0.        IND=  , IN= 4
YIELD =   4.130E+09 3.000E+11-0.         -0.         -0.         0.       0.        IND=  , IN= 4 DYN/CM2.  DYN/CM2,  DYN/CM2,
VISC =    4.000E+00 5.000E-02 5.000E-02 0.         0.         0.       0.        IND=  , IN= 4
EMELT =   6.600E+09 5.000E+09 0.        1.000E+00 2.500E-01-0.        -0.        IND=  , IN= 4 ERG/G,

1145 AL HOT                  -0.            ***       ***         -0          -0 IND=T , IN= 5 G/CM3
1145 AL HOT        RHDS =    2.705E+00 CFP = 0I0 DPY = 003        -0          -0 IND=  , IN= 4 G/CM3
EQST =    8.560E+II 0.        I.000E+11 2.100E+00 2.100E+00-0.        -0.        IND=  , IN= 4 DYN/CM2,=, ERG/G, , , DYN/CM2, ERG/G
DFR1 =   -6.250E-03-4.000E+09 1.000E-04 2.000E+I2-3.000E+09 9.600E+08-0.         IND=  , IN= 4
YIELD =   2.000E+09 3.000E+11-0.         -0.         -0.         0.       0.        IND=  , IN= 4 DYN/CM2,  DYN/CM2,  DYN/CM2,
VISC =    4.000E+00 5.000E-02 5.000E-02 0.         0.         0.       0.        IND=  , IN= 4
EMELT =   6.600E+09 5.000E+09 0.        1.000E+00 2.500E-01-0.        -0.        IND=  , IN= 4 ERG/G,

NLAYERS =           2 JMAT =       1    2   -0   -0   -0   -0   -0   -0   -0   -0 IND=  , IN= 5
NZONES= 1          I0 CELLS IN  3.130E-0I 4409-I   -0.                -0.        IND=  , IN= 5 CM, CM,
NZONES= I          20 CELLS IN  6.350E-01 CM       -0.                -0.        IND=  , IN= 5 CM, CM,
$NLIST EHL(12)=20*4.46E9,RHDS(2)=2.784 $                                        INPUT FROM -EXTRA- ROUTINE
EOF ENCOUNTERED BY EXTRA
```

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 x 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 deposites 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 labelled "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_o, 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 modecule, instead of the weight fraction (because PBW $\geq$ 1.0).

```
TIDENT AP  EXPERIMENT H 3116   X-RAY DEPOSITION INTO A HEAT SHIELD      APDS

TAP (ERLICH)

TFUSED SILICA-BARKER

TQUARTZ (GRAHAM)

 NLAYER =            3 JMAT =       1    2    3
 NZONES= 1         103 CELLS IN  3.550E-01 CM DX =    1.000E-04 RATIO =    1.05E0
 NZONES= 1          13 CELLS IN  6.350E-01 CM
 NZONES= 1          16 CELLS IN  8.000E-01 CM

 NSPEC =             1 ANGLE =   0.
TSPEC Z12  NHNU =    0 ECAL =    1.100E+02 START =   0.         SSTOP =   5.000E-09
7/8/9
```

(a) SAMPLE DATA DECK FOR RADIATION PROBLEM

```
AP (ERLICH)            RHOS =     1.843      CFP = 001 DPY = 001              NCON =  6
EQST =     3.800E+11-3.800E+11 8.200E+09 0.05       0.02       3.800E+11
RHO =      1.45
AK =       1.030E+10 MUP =     0.
NREG =             2
RHOP =     1.45      1.703     2.55      2.55       2.55       2.6
COSQ =     4.0       4.0       4.0       4.0        4.0        4.0
C1 =       0.3       0.3       0.3       0.3        0.3        0.3
P1 =       0.
1 P2 =     3.000E+09 DELP =   -4.000E+08
2 P2 =     3.500E+10 DELP =   -2.700E+09
EMELT =    4.000E+09 3.000E+09 0.15      0.25      -0.06
IRON       ITAPE=  2 PBW =     0.0061
OXYGEN     ITAPE=  2 PBW =     0.387
NICKEL     ITAPE=  2 PBW =     0.0017
SILICON    ITAPE=  2 PBW =     0.1052
MAGNESIUM  ITAPE=  2 PBW =     0.19
CARBON     ITAPE=  2 PBW =     0.31
```

(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

```
                       ****    SRI PUFF 8    ****

 DATE =  78/02/23.
IDENT APDS H 311B                                             APDS      IND=T , IN= 5
NTEDT =           4 NJEDIT =          2 NREZON=          4                 1 IND=   , IN= 4
TEDIT =    7.000E-09 1.000E-09 2.000E-08 5.000E-08-0.       -0.      -0.        IND=   , IN= 4  SEC
JEDIT =     50   62   70   76   81   84   87   90   93   95   97   99  100  101
           111  123  126  130                                                   IND=   , IN= 4
NTR   =      1    2    3    4    0    0    0    0    0    0    0    0    0    0 IND=   , IN= 4
JREZDN =    30   55   80  105    0    0    0    0    0    0    0    0    0    0 IND=   , IN= 4
NEDIT =              30 JCYCS =             0 CKS =      1.000E+01 TS =      5.000E-06 IND=   , IN= 4 , , , CM,  SEC
NMTRLS =              3 MATFL =             0 UZERO =    0.                ********** IND=   , IN= 4 , , , CM/SEC
        NSCRD INDICATORS  -  RAD. PLOTS  1-E, 2-T, 3-P, 4- ,       SCRIBE HISTORIES  1-6S, 2-12S, 3-DS,Y, 4-R,V, S-U,1, 6-NEM,NET

AP (ERLICH)                   -0.            ***         ***        -0           -0 IND=T , IN= 5 G/CM3
AP (ERLICH)           RHDS=    1.843E+00 CFP = 001 DPY = 001        -0 NCUN=     6 IND=   , IN= 4 G/CM3
EQST =     3.800E+11-3.800E+11 8.200E+09 5.000E-02 2.000E-02 3.800E+11-0.          IND=   , IN= 4 DYN/CM2,=, ERG/G, , , DYN/CM2, ERG/G
RHO =      1.450E+00
AK =       1.030E+10 MUP =      0.                  -0.                            IND=   , IN= 4 -POREQST-,DYN/CM2,=.=
NREG =             2                                                               IND=   , IN= 4 -POREQST-, ,G/CM3
RHOP =     1.450E+00 1.703E+00 2.550E+00 2.550E+00 2.550E+00 2.600E+00             IND=   , IN= 4 -POREQST-,G/CM3
COSQ =     4.000E+00 4.000E+00 4.000E+00 4.000E+00 4.000E+00 4.000E+00             IND=   , IN= 4 -POREQST-
C1    =    3.000E-01 3.000E-01 3.000E-01 3.000E-01 3.000E-01 3.000E-01             IND=   , IN= 4 -POREQST-
1 P1 =     0.                                                                      IND=   , IN= 4 -POREQST-,DYN/CM2,=
1 P2 =     3.000E+09 DELP =   -4.000E+08          -0.                              IND=   , IN= 4 -POREQST-,DYN/CM2,=,DYN-CM/G
2 P2 =     3.500E+10 DELP =   -2.700E+09          -0.                              IND=   , IN= 4 -POREQST-,DYN/CM2,=,DYN-CM/G
EMELT=     4.000E+09 3.000E+02 1.500E-01 2.500E-01-6.000E-02-0.        -0.
IRON       1TAPE=  2 PBW =       .0061000                                          IND=   , IN= 4 -DEPDS-
IRON       X-RAY ABS NOE =              2 ATWT =       55.850                      IND=   , IN= 2 -DEPDS-
EDGE     1.00000E+00  7.11200E+00                                                  IND=   , IN= 2 -DEPDS-
COEFS    1.36697E+01 -2.39272E+00 -1.36795E-01 -2.37212E-04
         1.43458E+01 -1.23512E+00 -4.18728E-01  3.21614E-02                        IND=   , IN= 2 -DEPDS-
OXYGEN     1TAPE=  2 PBW =       .3870000                                          IND=   , IN= 4 -DEPDS-
OXYGEN     X-RAY ABS NOE =              1 ATWT         15.999                      IND=   , IN= 2 -DEPDS-
EDGE     1.00000E+00                                                               IND=   , IN= 2 -DEPDS-
COEFS    1.17128E+01 -2.57213E+00 -2.05923E-01  1.99266E-02                        IND=   , IN= 2 -DEPDS-
NICKEL             2             .0017000                                          IND=   , IN= 4 -DEPDS-
NICKEL     X-RAY ABS NOE =              3 ATWT =       58.710                      IND=   , IN= 2 -DEPDS-
EDGE     1.00000E+00  1.01200E+00  8.33300E+00                                     IND=   , IN= 2 -DEPDS-
COEFS    1.38363E+01 -2.47740E+00  0.            0.
         1.39849E+01 -2.48097E+00 -8.88292E-02  3.18989E-05
         1.42375E+01 -9.66762E-01 -4.78299E-01  3.66306E-02                        IND=   , IN= 2 -DEPDS-
SILICON            2             .1052000                                          IND=   , IN= 4 -DEPDS-
SILICON    X-RAY ABS NOE =              2 ATWT =       28.090                      IND=   , IN= 2 -DEPDS-
EDGE     1.00000E+00  1.83900E+00                                                  IND=   , IN= 2 -DEPDS-
COEES    1.12236E+01 -2.73639E+00  1.26919E-01  0.
         1.32678E+01 -1.98148E+00 -3.17030E-01  2.74002E-02                        IND=   , IN= 2 -DEPDS-
MAGNESIUM          2             .1900000                                          IND=   , IN= 4 -DEPDS-
MAGNESIUM X-RAY ABS NOE =               2 ATWT =       24.310                      IND=   , IN= 2 -DEPDS-
EDGE     1.00000E+00  1.30500E+00                                                  IND=   , IN= 2 -DEPDS-
COEFS    1.05971E+01 -2.83633E+00  0.            0.
         1.28791E+01 -2.12553E+00 -2.99449E-01  2.67690E-02                        IND=   , IN= 2 -DEPDS-
CARBON             2             .3100000                                          IND=   , IN= 4 -DEPDS-
CARBON     X-RAY ABS NOE =              1 ATWT =       12.011                      IND=   , IN= 2 -DEPDS-
EDGE     1.00000E+00                                                               IND=   , IN= 2 -DEPDS-
COEFS    1.06880E+01 -2.71404E+00 -2.00516E-01  2.07238E-02                        IND=   , IN= 2 -DEPDS-
          DEPOS - CONST. DENSITIES (G/CM3), RHDC = 8.845E-03 5.612E-01 2.465E-03 1.525E-01 2.755E-01 4.405E-01

FUSED SILICA-BARKER           -0.            ***         ***        -0           -0 IND=T , IN= 5 G/CM3
FUSED SILICA-BARKER RHOS=      2.201E+00 CFP = 000 DPY = 000        -0 NCON=     2 IND=   , IN= 4 G/CM3
EQST =     7.760E+11-4.935E+12 8.900E+10 3.500E-02 2.500E-01 3.530E+13-0.          IND=   , IN= 4 DYN/CM2,=, ERG/G, , , DYN/CM2, ERG/G
SILICON    1TAPE=  2 PBW =      1.0000000                                          IND=   , IN= 4 -DEPDS-
```

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

```
SILICON   X-RAY ABS NOE =           2 ATWT =        28.090                       INO=   , IN= 2 -DEPOS-
EDGE      1.00000E+00  1.83900E+00                                               INO=   , IN= 2 -DEPOS-
COEFS     1.12236E+01 -2.73639E+00  1.26919E-01  0.
          1.32678E+01 -1.98148E+00 -3.17030E-01  2.74002E-02                     INO=   , IN= 2 -DEPOS-
OXYGEN     ITAPE=  2 PBW =      2.0000000                                        INO=   , IN= 4 -DEPOS-
OXYGEN    X-RAY ABS NOE =           1 ATWT          15.999                       INO=   , IN= 2 -DEPOS-
EDGE      1.00000E+00                                                            INO=   , IN= 2 -DEPOS-
COEFS     1.17128E+01 -2.57213E+00 -2.05923E-01  1.99266E-02                     INO=   , IN= 2 -DEPOS-
           DEPOS - CONST. DENSITIES (G/CM3), RHOC = 1.029E+00 1.172E+00

QUARTZ (GRAHAM)                 -0.              ***        ***         -0      -0 INO=T , IN= 5 G/CM3
QUARTZ (GRAHAM)      RHOS=      2.650E+00 CFP = 000 DPY = 000           -0 NCON=   2 INO=   , IN= 4 G/CM3
EQST =       8.695E+11-8.695E+11 8.900E+10 6.200E-01 2.500E-01 8.695E+11-0.          INO=   , IN= 4 DYN/CM2,=, ERG/G, , , DYN/CM2, ERG/G
SILICON    ITAPE=  2 PBW =      1.0000000                                        INO=   , IN= 4 -DEPOS-
SILICON   X-RAY ABS NOE =           2 ATWT =        28.090                       INO=   , IN= 2 -DEPOS-
EDGE      1.00000E+00  1.83900E+00                                               INO=   , IN= 2 -DEPOS-
COEFS     1.12236E+01 -2.73639E+00  1.26919E-01  0.
          1.32678E+01 -1.98148E+00 -3.17030E-01  2.74002E-02                     INO=   , IN= 2 -DEPOS-
OXYGEN     ITAPE=  2 PBW =      2.0000000                                        INO=   , IN= 4 -DEPOS-
OXYGEN    X-RAY ABS NOE =           1 ATWT          15.999                       INO=   , IN= 2 -DEPOS-
EDGE      1.00000E+00                                                            INO=   , IN= 2 -DEPOS-
COEFS     1.17128E+01 -2.57213E+00 -2.05923E-01  1.99266E-02                     INO=   , IN= 2 -DEPOS-
           DEPOS - CONST. DENSITIES (G/CM3), RHOC = 1.239E+00 1.411E+00

NLAYER =            3 JMAT =        1    2    3   -0    -0    -0    -0    -0    -0    -0 INO=   , IN= 5
NZONES= 1         103 CELLS IN  3.550E-01 CM DX =    1.000E-04 RATIO =    1.050E+00 INO=   , IN= 5 CM, CM,
NZONES= 1          13 CELLS IN  6.350E-01 CM          -0.                  -0.       INO=   , IN= 5 CM, CM,
NZONES= 1          16 CELLS IN  8.000E-01 CM          -0.                  -0.       INO=   , IN= 5 CM, CM,
NSPEC =             1 ANGLE =   0.          -0.        -0.          -0.    -0.       INO=   , IN= 5 -DEPOS-   ANGLE FROM NORMAL(DEG)
SPEC 05    NHNU =   0 FCAL =       .110E+03 START =    0.         SSTOP =     .500E-08 INO=T , IN= 5 -DEPOS-   , , ,CAL/CM2,SEC,SEC
SPEC 05               NHNU =              95 (4(F10.5,F7.5,3X))                     INO=   , IN= 4 -DEPOS-
    .035710.00000          .107140.00000          .178570.00000          .250000.00000
    .32143 .00001          .39286 .00002          .46429 .00003          .53571 .00004
    .60714 .00006          .67857 .00009          .75000 .00012          .82143 .00016
    .89286 .00021          .96429 .00026         1.07143 .00093         1.21429 .00156
   1.35714 .00209         1.50000 .00254         1.64286 .00290         1.78571 .00316
   1.92857 .00333         2.08333 .00436         2.25000 .00508         2.41667 .00565
   2.58333 .00608         2.75000 .00635         2.91667 .00648         3.12500 .00993
   3.37500 .01053         3.62500 .01145         3.87500 .01269         4.12500 .01420
   4.37500 .01563         4.62500 .01684         4.87500 .01782         5.16667 .02577
   5.50000 .02789         5.83333 .02884         6.16667 .02976         6.50000 .03075
   6.83333 .03069         7.16667 .03068         7.50000 .03062         7.83333 .02941
   8.25000 .04213         8.75000 .03797         9.25000 .03188         9.75000 .02682
  10.25000 .02138        10.75000 .01842        11.25000 .01734        11.75000 .01616
  12.25000 .01546        12.75000 .01464        13.25000 .01379        13.75000 .01311
  14.50000 .02460        15.50000 .02280        16.50000 .02070        17.50000 .01870
  18.50000 .01720        19.50000 .01610        20.50000 .01530        21.50000 .01480
  22.50000 .01450        23.50000 .01400        24.50000 .01160        25.50000 .00680
  26.50000 .00440        27.50000 .00580        28.50000 .00720        29.50000 .00730
  30.50000 .00680        31.50000 .00630        32.50000 .00590        33.50000 .00550
  34.50000 .00510        35.50000 .00480        36.50000 .00440        37.50000 .00410
  38.50000 .00370        39.50000 .00340        40.50000 .00320        41.50000 .00290
  42.50000 .00270        43.50000 .00250        44.50000 .00230        45.50000 .00210
  46.50000 .00200        47.50000 .00180        48.50000 .00170        49.50000 .00160
  51.25000 .00284        53.75000 .00376        57.50000 .00490                           INO=   , IN= 4 -DEPOS-
           DEPOS - ESUM = 1.100E+02 CAL/CM2
```

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)

Additional Input: EXTRA and H-DATA

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 4.46×10^{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   D,18   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

C	Sound speed, cm/sec
COM, COM2, COM12	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.
D	Density, g/cm^3
DPDD	$\partial P/\partial \rho$, $dyn/cm^2/(g/cm^3)$
DPDE	$\partial P/\partial E$, $dyn/cm^2/(erg/g)$
E	Internal energy, erg/g
H1	H(J,1), cell state indicator
H2	H(J,2), cell or coordinate type indicator
H3	H(J,3), cell state indicator
IMP	Impulse = $\int R dt$, $dyn\text{-}sec/cm^2$
NEM, NET	Special arrays; meaning depends on the material model
P	Pressure, dyn/cm^2
R	Mechanical stress in direction of propagation, dyn/cm^2
SDT	Deviator stress in the circumferential direction in cylindrical problems, dyn/cm^2
SD1, SD2, SD3	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
S1, S2, S3	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
S-INT	Interface stress--average of stresses in cells on either side of interface, dyn/cm^2

Table C.1 (concluded)

T	Spall strength, dyn/cm^2
U	Coordinate velocity, cm/sec
V	Specific volume, cm^3/g
X	Coordinate location, cm
XO	Initial coordinate location, cm
Y	Yield strength, dyn/cm^2
Z	Cell mass, g/cm^2, g/cm, or g for planar, cylindrical and spherical geometries, respectively
1,2, any integer	Cell or coordinate number
3.25	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

Table C.2

SAMPLE INPUT DIRECTIVES

```
  1*             10 11                                80
   JEDIT =            16   23   4   29   18   D,8,9,11
   JEDIT 2            S1,S2,S3,E,Y,   6,7,8    X,6, 9
   HIST 3             U, H2,NEM,16      C,U, SD1  24
   4th CARD           30,35,COM24     5   10   15   20
```

---

[*] 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 $4 + \text{NANG} + 2\sum_i \text{NSIZE}_i$, where NANG and NSIZE_i 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 guide lines 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 σ_1 is required. In Figure C.11, all three prinicpal 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).

```
 1DENT 847 I FRACTURE IN 1145 AL. FRACTURE 1MPACT EXPERIMENT AT 423 FT/SEC
 NED1T =           0 NJED1T =          2
 JED1TS =     27   28   29   30   31   32   15   16   17   18   19   20   21   22
             23   24   25   26
 NED1T =          20 JCYCS =         180 CKS =     3.0       TS =      3.000E-06
 NMTRLS =          2 MATFL =           1 UZERO =   1.289E+04
 AL 1145             RHOS =    2.7E0     CFP = 000 DPY = 003           NCON =  0
 EQST =    7.600E+11 1.500E+12 1.220E+11 2.04E0    0.25E0    0.
 TENS =   -1.000E+11 0.       -1.E0
 V1SC =    4.E0      0.05E0    0.
 Y1ELD =   2.000E+09 3.000E+11 0.
 AL 1145 FR          RHOS =    2.7E0     CFP = 010 DPY = 003           NCON =  0
 EQST =    7.600E+11 1.500E+12 1.220E+11 2.04E0    0.25E0    0.
 DFR1 1145-0.01     -4.000E+09 1.000E-04 3.000E+09-3.000E 09-4.000E+08
 TENS =   -1.200E+10 0.
 V1SC =    4.E0      0.05E0    0.
 Y1ELD =   2.000E+09 3.000E+11 0.
 NLAYER=           2 JMAT =        1    2
 NZONES= 1        10 CELLS 1N  0.236     CM
 NZONES= 1        25 CELLS 1N  0.635     CM
7/8/9
```

FIGURE C.7 INPUT DECK FOR IMPACT IN 1145 ALUMINUM, ILLUSTRATING DUCTILE FRACTURE DATA AND JEDITS WITH NO TYPE INDICATOR

```
 IDENT = S25 E SHOT 8678-1-S25 IN ARMCO IRON
C   STANDARD IMPACT SIMULATION USED TO CALIBRATE OR CHECK BRITTLE FRACTURE
C   MODELS
 NTEDT =           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 =      1.000E-05
 NMTRLS =          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-01 5.170E+13
 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
 Y0 =      2.000E+09 8.190E+11
 PMMA-8KB (BARKER)   RHOS =    1.184     CFP = 000 DPY = 001
 EQST=     7.000E+10 4.050E+11 1.000E+10 1.000E+00 2.500E-01 3.640E+11
 YIELD =   1.000E+06 1.950E+10 2.350E+09
 NLAYER =          3 JMAT =       1    1    2
 NZONES= 1        10 CELLS IN  1.138E-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
```

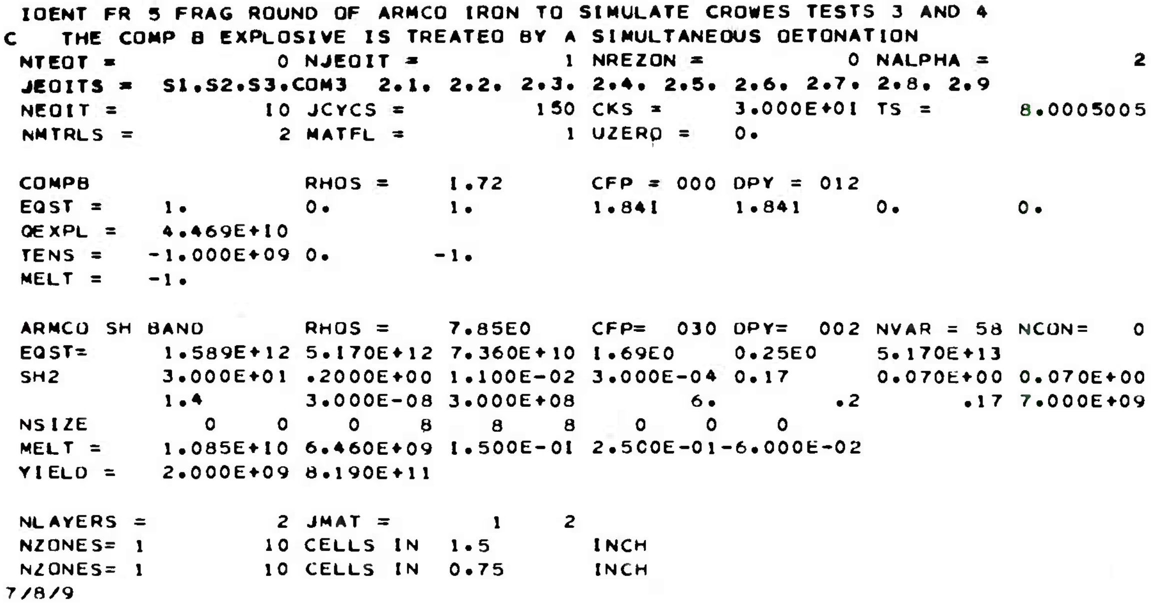
FIGURE C.8 INPUT DECK FOR IMPACT IN ARMCO IRON, ILLUSTRATING BRITTLE FRACTURE AND A GEOMETRIC CELL LAYOUT

```
 IDENT = 103A SYMMETRIC IMPACT OF TONALITE, COULOMB FRICTION AND POROSITY
 NTEDT =           0 NJEDIT =          1                   0                   C
 JEDITS =     1   11   21   31   36   41
 NEDIT =          50 JCYCS =         200 CKS =     6.0       TS =      8.000E-06
 NMTRLS =          1 MATFL =          -1 UZERO =   6.320E+04
 C C TONALITE        RHOS =    2.58E0    CFP = C01 DPY = 002           NCON =  C
 EQST =    2.940E+11 3.056E+12 1.000E+12 2.E0      0.25E0  -6.406E+12
 RHO =     2.56E0
 AK =      2.000E+11 MUP =     1.000E+11 YC =      0.
 NREG =            1 RHOP1     0.
 RHOP =    2.56E0    2.762 E0  2.762E0   2.762E0   2.762E0   2.8E0
 COSQ =    4.E0      4.E0      4.E0      4.E0      4.E0      4.E0
 C1 =      0.2E0     0.2E0     0.2E0     0.2E0     0.5E0     0.5E0
 P1 =      1.000E+08
 IP2 =     3.300E+10 DELP =   -3.500E+09 YADDP =   0.
 TENS =   -1.000E+11-1.000E+11-1.E0
 YIELD =   0.        1.000E+11 0.        0.056E0
 NLAYER =          1 JMAT =          1
 NZONES= 3        10 CELLS IN  0.269     CM
                  20 CELLS IN  0.576     CM
                  10 CELLS IN  0.297     CM
7/8/9
```

FIGURE C.9 INPUT DECK FOR ASYMMETRIC IMPACT OF TONALITE, SHOWING COULOMB FRICTION WITH TANϕ = 0.056, MULTIPLE ZONES IN LAYER, AND USE OF THE POREQST MODEL

```
IDENT PROJECTILE IMPACT ON CONCRETE AT 22.34M/SEC - TEST REBAR
NTEDT =              0 NJEDIT =            1
JEDIT =      5      8     11     14     16     18     20     22     25     27     29
NEDIT =             10 JCYCS =           150 CKS =      1.000E+02 TS =        1.000E-04
NMTRLS=              4 MATFL =             1 UZERO =    2.234E+03
IMPACTOR STEEL      RHOS=           7.85E0 CFP=  0000PY=    001
EQST =      1.5889E12  5.170E12  7.360E10    1.69E0     0.25E0  5.170E13
YIELD=       1.222E10  8.188E11
REBAR STEEL         RHOS=           7.85E0 CFP=  0000PY=    001
EQST =      1.5889E12  5.170E12  7.360E10    1.69E0     0.25E0  5.170E13
YIELD=       1.030E10  8.188E11
CONCRETE            RHOS =      2.85 E0   CFP = 004 DPY = 000
EQST =      2.830E+11 0.         1.000E+11 2.000E+00 .25        0.
RHO =       2.22E0     AMU =     2.033E+11
AK =        7.000E+10 AK2 =    -0.550E+02 MUP =      5.250E+10 MUP2 =     0.125E+03
MC =        1.040E+09-8.300E+08 2.703E+09 2.500E+08 1.E0
SCRIT(M)= 2.300E+08 DAMG =     0.100E-02
EVP =       0.          -1.200E-02-3.500E-02-5.000E-02-2.230E-01
NREG =               4 NPRCAP =            0 P1 =      -3.500E+08 W2 =        1.25E0
P2 =      -1.000E+09 DELP =        0.
P2 =      -2.400E+09 DELP =    0.
P2 =      -3.400E+09 DELP =    0.
P2 =      -1.533E+10 DELP =    0.
REBAR               RHOS=        2.5015E0 CFP=  100DPY=    000 VAR=      7
FS=          0.05E0THET=              0.01MC=                  31MS=                 2
NLAYER =             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 CROWES TESTS 3 AND 4
C   THE COMP B EXPLOSIVE IS TREATED BY A SIMULTANEOUS DETONATION
 NTEDT =              0 NJEDIT =             1 NREZON =             0 NALPHA =             2
 JEDITS =  S1.S2.S3.COM3  2.1. 2.2. 2.3. 2.4. 2.5. 2.6. 2.7. 2.8. 2.9
 NEDIT =             10 JCYCS =            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.         1.          1.841       1.841       0.          0.
 QEXPL =    4.469E+10
 TENS =    -1.000E+09 0.        -1.
 MELT =    -1.

 ARMCO SH BAND          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
 SH2        3.000E+01 .2000E+00 1.100E-02 3.000E-04 0.17        0.070E+00 0.070E+00
            1.4       3.000E-08 3.000E+08         6.          .2        .17 7.000E+09
 NSIZE         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 JMAT =       1     2
 NZONES= 1           10 CELLS IN  1.5         INCH
 NZONES= 1           10 CELLS IN  0.75        INCH
7/8/9
```

FIGURE 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

```
 IDENT FR 4 FRAGMTG RNO OF HF-1 TO SIMULATE CROWES TESTS 1 AND 2.
 NTEOT =           0 NJEDIT =          3                     NALPHA =           2
 JEDIT =  S1 2.1.2.2.2.3.2.4.2.5.2.6.2.7.2.8.2.9
 JEOIT =  COM2.COM3.COM4.12.13.14.15.16.17.18.19
 JEDIT 3  S2.S3.U.0.17.18.19
 NEDIT =          10 JCYCS =         300 CKS =     30.       TS =      8.000E-05
 NMTRLS =          2 MATFL =           1 UZERO =   0.

 COMPB               RHOS =    1.72      CFP = 000 DPY = 012
 EQST =    1.0       0.        1.0       1.841     1.841
 QEXPL =   4.469E+10
 TENS =   -1.000E+09 0.       -1.
 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       .17 7.000E+09
 NSIZE         0    0    0    8    8    8    0    0    0
 YIELO =   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
7/8/9
```

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 KEOUGH ON 6172-10
NTEDT =              0 NJEDT =              1 NREZON =            0 NALPHA =             2
JEDIT =     S1.1.1.S.1.75.2.1.2.2.2.3.2.4.2.5.2.6.2.8 S-INT.0
NEDIT =             10 JCYCS =            100 CKS =       100.      TS =        4.000E-04
NMTRLS =             2 MATFL =             -2 UZERO =    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 Y0 =        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 =      2.680E+08
P2 =        3.400E+09 DELP =      0.          YADD =      1.220E+08
P2 =        1.533E+10 DELP =      0.          YADD =      2.690E+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+10 MUP =       3.887E+10 Y0 =        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
7/8/9
      FUNCTION SIGMAT(LS,T)
      DIMENSION PS(10),TS(10)
      DATA PS/0.,4.E9,4.E9,7*0./
      DATA TS/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

```
 IDENT = SPHERICAL PUFF*/AL6061T6/PETN/PMMA/GROUT(SRIRMG-2C3) - CAP
C   A SPHERICAL EXPLOSION WITH A CENTAL BALL OF AL, THEN A SHELL OF PETN
C   IN A PLASTIC CONTAINER IN A LARGE SPHERE OF ROCK-MATCHING GROUT
 NTEDIT =          0 NJEDIT =          1 NREZONE =          0 NALPHA =          3
 JEDIT =      7    9   11    13   15   17   19   21   23   25
 NEDIT =          10 JCYCS =         120 CKS =     1.000E+02 TS =      1.000E-04
 NMTRLS =          4 MATFL =          -3 UZERO =   0.

 AL6061-T6           RHOS =    2.707E0   CFP = 000 DPY = 001           NCON =  1
 EQST =    6.670E+11 1.000E+12 1.220E+11 2.040E0   2.500E-01 0.        0.
 YIELD =   3.210E+09 2.670E+11 3.790E+10 0.        0.        0.        0.

 PETN =              RHOS =    1.E0      CFP = 000 DPY = 063
 EQST =    1.E0      0.        1.E0      1.45E0    1.45E0    0.        0.
 IMAX =           21
 RHOP =    1.E0      0.501E+11 0.794E0   0.300E+11 0.633E0   0.183E+11
           0.5E0     0.114E+11 0.398E0   0.725E+10 0.316E0   0.469E+10
           0.251E0   0.309E+10 0.2E0     0.207E+10 0.159E0   0.140E+10
           0.126E0   0.967E+09 0.1E0     0.675E+09 0.794E-01 0.477E+09
           0.633E-01 0.341E+09 0.500E-01 0.245E+09 0.398E-01 0.182E+09
           0.316E-01 0.130E+09 0.251E-01 0.960E+08 0.200E-01 0.711E+08
           0.156E-01 0.530E+08 0.126E-01 0.397E+08 0.100E-01 0.299E+08
 TENS =   -1.000E+09 0.       -1.000E+09 0.        0.        0.        0.
 MELT =   -1.000E0  0.         0.        0.        0.        0.        0.
 VISC =    4.E0      2.500E-01 2.500E-01 0.        0.        0.        0.

 PMMA-8KB (HARKER)   RHOS =    1.184E0   CFP = 000 DPY = 001
 EQST =    7.000E+10 4.050E+11 1.000E+10 1.000E0   2.500E-01 3.640E+11 0.
 YIELD =   1.000E+06 1.950E+10 2.850E+09 0.        0.        0.        0.

 SRIRMG-2C3 GROUT    RHOS =    2.148E0   CFP = 004 DPY = 002
 EQST =    1.450E+11 0.        1.000E+11 2.E0      2.500E-01 0.        0.
 RHO =      2.1E0    AMU=      9.000E+10
 AK =      1.046E+11 AK2 =     0.        MUP =     6.897E+10 MUP2 =    0.
 MC =      2.600E+08-1.000E+08 1.321E+09 1.000E+09 2.E0
 SCRIT =   2.300E+08 DAMAG =   1.000E-03
 EVP=      0.E0     -4.531E-03-6.806E-03-1.257E-02
 NREG =            3 NPRCAP =          0 P1 =     -1.000E+08 W2 =      1.000E+04
 1P2 =    -.3200E+09 DELP =   -3.000E+07
 2P2 =    -1.150E+09 DELP =   +.2000E+09
 3P2 =    -4.000E+09 DELP =   +.1300E+09
 YIELD =   4.345E+08 9.000E+10 0.        0.        0.        0.        0.
 VISC =    4.E0      2.500E-01 2.500E-01 0.        0.        0.        0.

 NLAYERS =         4 JMAT =       1    2    3    4
 NZONES= 1         1 CELLS IN  9.068E-02 CM
 NZONES= 1         5 CELLS IN  4.021E-01 CM
 NZONES= 1         3 CELLS IN  1.422E-01 CM
 NZONES= 1        36 CELLS IN  1.461E+01 CM DX =   1.000E-01 RATIO =   1.100E0

 EXTRA
$NLIST RHL(9)=40*6.89E7,PHL(9)=40*6.89E7,SHL(9)=40*6.89E7,P6(1)=6.89E7,
   T6(1)=1.0$
7/8/9
```

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

```
 IDENT = 254 H EXPLOSIVE. FLYER. AND TARGET.  SIMULTANEOUS DETONATION
C  ELIMINATE C. D. AND S FROM EOS FOR EXPL. LET GAMMA BE CONSTANT.
C  ADD 2 CM TO TARGET. CHECK WHETHER RIGHT IMPULSE IS OBTAINED (27.3 KBAR)
 NTEDT =             0 NJEDIT =            2                   0                   0
 JEDITS =     11    12    14    15    20    24    28    32    36    40    44    48    52    56
              60    64    68    72
 NEDIT =            10 JCYCS =           170 CKS =       10.0       TS =       2.500E-05
 NMTRLS =            3 MATFL =             1 UZERO =   0.
 EL-506D               RHOS =     1.4        CFP = 000 DPY = 002
 EQST =     1.0        0.         1.         2.         2.         0.
 TENS =    -1.         0.        -1.000E+11
 MELT =    -1.         0.         0.         0.         0.

 AL6061-T6             RHOS =     2.707E0    CFP = 000 DPY = 002                  NCON =  0
 EQST =     6.670E+11 1.000E+12 1.220E+11 2.04E0     0.25E0     0.
 TENS =    -1.000E+11 0.        -5.000E+08
 YIELD =    3.210E+09 2.670E+11 3.790E+10

 OTWR                  RHOS =     1.63E0     CFP = 000 DPY = 001                  NCON =  0
 EQST =     7.490E+10 1.500E+11 1.600E+10 0.74E0     0.25E0     1.310E+11
 YIELD =    2.400E+08 3.000E+10 0.

 NLAYER =            4 JMAT =          1     2     0     3
 NZONES= 1          10 CELLS IN  0.202      CM
 NZONES= 1           2 CELLS IN  0.0406     CM
 NZONES= 1           0 CELLS IN  1.3700     CM
 NZONES= 2          37 CELLS IN  1.0        CM. DX =  1.000E-02 RATIO =    1.05
                    25 CELLS IN  3.0        CM. DX =  6.100E-02 RATIO =    1.05
 EXTRA
 $NLIST EHL(I)=10*3.64E+10 $
7/8/9
```

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, COMP 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
 JEDITS=        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      TS =       1.000E-04
 NMTRLS =           4 MATFL =             1 UZERO =    0.

 HMX                  RHOS =    1.84      CFP = 000 DPY = 012
 EQST =    1.0        0.        1.0       1.89      1.89      0.
 QEXPL=    5.690E+10 0.0625    2.0
 TENS =   -1.0        0.       -1.000E+11
 MELT =   -1.0        0.        0.        0.        0.

 COMP B               RHOS =    1.68E0    CFP = 000 DPY = 012                 NCON =  0
 EQST =    1.E0       0.        1.E0      1.63E0    1.63E0    0.
 QEXPL =   5.190E+10 0.0625E0  2.E0
 TENS =   -1.E0       0.       -1.000E+11
 MELT =   -1.E0       0.        0.        0.

 TNT                  RHOS =    1.56E0    CFP = 000 DPY = 012                 NCON =  0
 EQST =    1.E0       0.        1.E0      1.44E0    1.44E0    0.
 QEXPL =   4.520E+10 0.0625E0  2.0
 TENS     -1.E0       0.       -1.000E+11
 MELT =   -1.E0       0.        0.        0.        0.

 AL6061-T6            RHOS =    2.707E0   CFP = 000 DPY = 001                 NCON =  0
 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         80 CELLS IN  10.0      CM
 NZONES= 1         24 CELLS IN  3.0       CM
 NZONES= 1         24 CELLS IN  3.0       CM
 NZONES= 1         50 CELLS IN  6.0       CM
7/8/9
```

FIGURE C.16 INPUT DECK FOR A RUNNING DETONATION THROUGH THREE EXPLOSIVES, ILLUSTRATING THAT PUFF PERMITS OVERDRIVING OF EXPLOSION

```
 IDENT = DC5-A-1 SINTERED AL203 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.
 NTEDIT =          1 NJEDIT =          1 NREZONE =       -30
 TEDIT =   5.000E-09
 JEDITS =      5   17   26   40   48   61   66   68   70   72   74   76   78   80
 DTMAX =   5.000E-09 TREZON =  5.000E-09
 NEDIT =          30 JCYCS =         200 CKS =           2.0 TS =      1.500E-06
 NMTRLS =          3 MATFL =           0 UZERO =          0.

 BERYLLIUM           RHOS =    1.85E0    CFP = 000 DPY = 001
 EQST =    1.203E+12 1.524E+12 3.550E+11 1.45E0    2.500E-01 5.130E+11 0.
 MELT =    3.955E+10 1.978E+10 1.500E-01 2.500E-01-6.000E-02 4.500E-01

 ALUMINA SINTERED    RHOS =        3.969 CFP = 001 DPY = 004
 EQST =    2.655E+12 4.200E+12 3.080E+11 1.320E+00 2.500E-01 2.090E+12
 RHO =     3.16E0
 AK =      1.700E+12 MUP =     1.000E+12 Y0 =      3.000E+09
 NREG =            3
 RHOP =    2.8E0     3.E0      3.92E0    4.44E0    4.44E0    4.5E0
 C1 =           .050      .050      .050      .050      .050      .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
 MELT =    4.500E+10 1.350E+10 2.000E-02   .9        .2
 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       .79       .20 2.213E+11
 EMELT =   6.000E+09 4.000E+09        .1        .6   -.15

 NLAYERS=          4 JMAT =      1    2    3    3
 NZONES= 1        31 CELLS IN         .1 CM DX =         .0076 RATIO =       .935
 NZONES= 1        31 CELLS IN  .1        CM DX =          .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 = DC5 AL2D3   TH = NP=          7 (8E10.3)
 0.        1.073E+01 8.333E-03 9.178E-01 1.667E-02 6.342E-01 2.500E-02 4.954E-01
 5.000E-02 3.090E-01 7.500E-02 2.317E-01 1.000E-01 1.866E-01
 IDENT = DC5 AL203   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 = DC5 AL203   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-01 7.091E-02 3.250E-01 7.016E-02 3.500E-01 6.940E-02
 IDENT = DC5 AL2D3   TH = NP=          7 (8E10.3)
 3.500E-01 6.940E-02 4.333E-01 6.681E-02 5.167E-01 6.414E-02 6.000E-01 6.141E-02
 6.833E-01 5.861E-02 7.667E-01 5.572E-02 8.500E-01 5.270E-02
7/8/9
```

FIGURE C.17 INPUT DECK FOR RADIATION INTO BERYLLIUM AND ALUMINA, SHOWING THE USE OF A DEPTH-DOSE PROFILE, GEOMETRIC LAYOUT, AND AUTOMATIC REZONING

```
 IDENT I002  X-RAY DEPOSITION INTO AL  FOR POST TEST STUOY
C  THE X-RAY DEPOSITION HAS BEEN CALCULATED BY -FSCATT- AND IS PROVIDED AS
C   A DEPTH-DOSE PROFILE FOR THE PUFF CALCULATION
 NTEDIT =           0 NJEOIT             2 NREZON =           0
 JEDITS =     28    31    34    37    40    43     I     5    I0    IS    20    2S    45    50
              SS    60    6S    70
 NEOIT =           30 JCYCS =          150 CKS =      2.000E+0I TS =        S.000E-07
 NMTRLS =           I MATFL =            0 UZERO =    0.

 AL 1145              RHOS =     2.7E0      CFP = 0I0 DPY = 003
 EQST =    7.600E+11 I.500E+12 1.220E+1I 2.04E0     2.S00E+01 0.        0.
 DFRI 1145-1.000E-02-4.000E+09 I.000E-04 3.000E+09-3.000E+09-4.000E+08 0.
 VISC =    4.000E0   S.000E-02 0.        0.        0.        0.        0.
 YIELD =   2.000E+09 3.000E+11 0.        0.        0.        0.        0.
 MELT =    6.590E+09 2.400E+09 I.S00E-01 2.S00E-0I-6.000E-02 0.

 NLAYERS =          I JMAT        1
 NZONES= 3         27 CELLS IN  I.I00E-0I CM OX =    3.S00E-03 RATIO =    9.S00E-0I
                   17 CELLS IN  3.400E-02
                   27 CELLS IN  I.I00E-0I CM DX =    1.000E-03 RATIO =    1.05E0

 NSPEC =            1 ANGLE =   0.
 DQPOST     NARB    0 ECAL =    7.440E+02 START =   0.          SSTOP =   5.000E-09
 IDENT=DQPOST NOS.   TH = NP =            S (8EI0.3)
 2.4I3E0   6.108E-02 2.476E0   S.707E-02 2.54E0    S.33SE-02 2.603E0   4.98IE-02
 2.667E0   4.978E-02
7/8/9
```

FIGURE C.18 INPUT DECK FOR RADIATION INTO ALUMINUM, SHOWING A DEPTH-DOSE PROFILE AND MULTIPLE ZONES IN ONE LAYER

```
 IDENT = 610 200 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.
 NTEDIT =          9 NJEDIT =          1 NREZON =          4 NALPHA =          1
 TEDIT =   1.000E-08 1.000E-07 2.000E-07 4.000E-07 7.000E-07 1.000E-06 1.400E-06
           2.000E-06 3.000E-06
 JEDIT =     25   42   54  100  128  142
 NTR =        2    3    4    5
 JREZON =    45   60   80  110
 NEDIT =           50 JCYCS =         200 CKS =      6.E0      TS =      6.000E-06
 NMTRLS =          1 MATFL =           0 UZERO =   0.

 ALUMINUM            RHOS =    2.785E0   CFP = 000 DPY = 004           NCON =  1
 EQST =    7.550E+11 1.290E+12 1.220E+11 2.04E0    2.500E-01 1.197E+12 3.110E+10
 TENS =   -2.000E+10 0.       -2.000E+10
 COSQ =    3.24E0    2.500E-01 0.
 YO =      2.500E+09 2.870E+11
 EMELT =   1.060E+10 2.400E+09 1.500E-01 2.500E-01-6.000E-02
 ALUMINUM  ITAPE = 5 PBW =     1.
 ALUMINUM X-RAY ABS NDE =              2 ATWT =   26.98
 EDGEI=       1.00000E  0, 1.56000E  0
 COEF1        1.08710E  1,-2.78415E  0, 1.89848E -1,  .00000E  0
 COEF2        1.31739E  1,-2.18214E  0,-2.58940E -1, 2.22834E -2

 NLAYERS =         1 JMAT =       1
 NZONES= 3        40 CELLS IN  1.224E-02 CM DX =   1.000E-04 RATIO =   1.051E0
                  33 CELLS IN  5.776E-02 CM DX =   7.187E-04 RATIO =   1.05E0
                  76 CELLS IN  2.93E0    CM DX =   3.618E-03 RATIO =   1.05E0

 NSPEC =           3 ANGLE =   0.
 NHNU      NBB =   1 ECAL =    4.2E0     START =   0.        SSTOP =   3.500E-09
 200 CAL             TEMP =    3.7E0     ECAL =    4.2E0
 NHNU      NBB =   1 ECAL =    8.7E0     START =   3.500E-09 SSTOP =   4.500E-09
 200 CAL             TEMP =    2.370     ECAL =    8.7E0
 NHNU      NBB =   1 ECAL =    1.380E+01 START =   4.500E-09 SSTOP =   5.500E-09
 200 CAL             TEMP =    2.08E0    ECAL =    1.380E+01
7/8/9
```

FIGURE C.19 INPUT DECK FOR RADIATION FROM THREE BLACK BODIES INTO ALUMINUM, SHOWING FOUR REZONES AND MULTIPLE ZONES IN ON ONE LAYER

```
IDENT = NEUTRON GEN/CONFIGURATION W/ZRCLY-2 CLADDING
NTEDT =           1 NJEDIT =          1 NREZON =          0
TEDIT =   1.000E-09
JEDIT =  S1,1.03.1.06.1.09,1.2,1.4,1.6,1.8.2.5,3.5,4.5
NEDIT =           5 JCYCS =         200 CKS =     10.E0     TS =      6.000E-06
NMTRLS =          4 MATFL =           0 UZERO =   0.

U-3PCTMO            RHOS =    17.8608E0 CFP = 000 DPY = 004
EQST =    1.202E+12 1.459E+12 2.059E+10 2.03E0    2.500E-01 5.840E+12
MELT  =   2.050E+09 1.271E+09 1.160E-01 5.000E-02 0.
YIELD =   9.100E+08 6.314E+11
TENS =   -8.000E+09-1.000E+11 0.
VISC =    4.E0      5.000E-02 5.000E-02

NA-600C             RHOS =    8.080E-01 CFP = 000 DPY = 001
EQST =    3.200E+10 0.0       1.000E+11 1.0E0     2.500E-01
TENS =   -1.000E+11-1.000E+11 0.
ZIRCALOY-2          RHOS =    6.55E0    CFP = 000 DPY = 004
EQST =    9.031E+11 0.        6.714E+10 7.700E-01 2.500E-01 0.
EMELT =   6.615E+09 2.381E+09-7.200E-02 2.400E-01-6.000E-02
VISC =    4.E0      5.000E-02 5.000E-02
TENS =   -1.000E+11-1.000E+11 0.
YIELD =   3.000E+09 3.472E+11

NAK-100C            RHOS =    8.470E-01 CFP = 000 DPY = 001
EQST =    5.300E+10 0.0       1.000E+11 1.0E0     2.500E-01
TENS =   -1.000E+11-1.000E+11-1.000E+11

NLAYERS =         4 JMAT =       1    2    3    4
NZONES= 1        29 CELLS IN  5.350E-01 CM DX =   6.000E-03 RAT1D =   9.346E-01
NZONES= 1         3 CELLS IN  1.270E-02 CM
NZONES= 1         3 CELLS IN  2.540E-02 CM
NZONES= 1        12 CELLS IN  5.000E-02 CM

NSPEC =           1 ANGLE     0.
PROTON    NARB =  0 ECAL =    5.900E-01 START =   0.        SSTOP =   2.000E-07
IDENT = U           TH = NP =         4 (1P8E10.3)
0.        1.E0      2.000E-01 1.E0      4.000E-01 1.E0      5.350E-01 1.E0
IDENT = NA          NPOINTS =         3 (1P8E10.3)
0.E0      0.4E0     .00635E0  0.4E0     0.0127E0  0.4E0
IDENT = ZRCLY-2     NPOINTS =         3 (1P8E10.3)
0.E0      0.4E0     0.0127E0  0.4E0     0.0254E0  0.4E0
IDENT = NAK         NPOINTS =         3 (1P8E10.3)
0.        0.4E0     0.02E0    0.4E0     0.05E0    0.4E0

H-DATA
H =          1    2         M H =         52    2         M

EXTRA
$NLIST RHOS(1)=1.845E+01.EQSTN(1)=1.55867E0.EHL(1)=30*1.065E+09$
7/8/9
```

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 NZEZON =              0
 JEDITS=       1    11    22    33    43    54    65    75    86    97   107   118   129   139
             160   174   179   183
 NEDIT =             10 JCYCS =             120 CKS =      3.00       TS=          7.000E-06
 NMTRLS=              3 MATFL =              -2 UZERO=     0.

 OTWR                   RHOS =      1.63E0     CFP = 000 DPY = 001                NCON =   0
 EQST =      7.490E+10 1.500E+11 1.600E+10 0.174E0    0.25E0      1.310E+11
 YIELD =     2.400E+08 3.000E+10 0.

 BOND                   RHOS =      1.1E0      CFP = 000 DPY = 000                NCON =   0
 EQST =      2.990E+10 1.588E+11 1.000E+10 0.5E0      0.25E0     -1.450E+11

 AL6061-T6              PHOS =      2.707E0    CFP = 000 DPY = 001                NCON =   0
 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 =             3 JMAT =        1     2     3
 NZONES= 1          150 CELLS IN  1.016      CM
 NZONES= 1           17 CELLS IN  0.076      CM
 NZONES= 1           19 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

```
IDENT JIM GRANS SHOCK TUBE
NTEDT             0 NJEDT             1                     NALPHA             1
JEDIT =   P, 10,15,20,23,30
NEDIT            10 JCYCS           150 CKS       1.000E+03 TS        3.300E-03
NMTRLS            2 MATFL            -2 UZERO     0.

AIR                 RHOS      1.169E-03 CFP   000 DPY   012
EQST      1.        0.        0.         .4        .4       0.
Q =       2.170E+09
TENS     -1.000E+10-1.000E+10-1.000E+10
VISC      4.         .02

STEEL               RHOS      7.903     CFP   000 DPY   001
EQST      1.648E+12 2.932E+12 7.360E+10 1.17       .25      4.658E+12
YIELD     7.013E+09 7.892E+11

NLAYER            2 JMAT         1   -2
NZONES  1        30 CELLS IN  15.5      INCH
NZONES  1         5 CELLS IN  25.       CM

EXTRA
$NLIST P6(1)=1.379E7,T6(1)=-1.,RHOS(2)=7.902995137$
7/8/9
```

FIGURE C.22 INPUT DECK FOR SIMULATING AN AIR SHOCK BY APPLYING A PRESSURE BOUNDARY THROUGH NAMELIST

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 ith interval, the coefficients are

$$F = F_{ai+1} + F_{bi+1}\, E + F_{ci+1}\, E^2$$

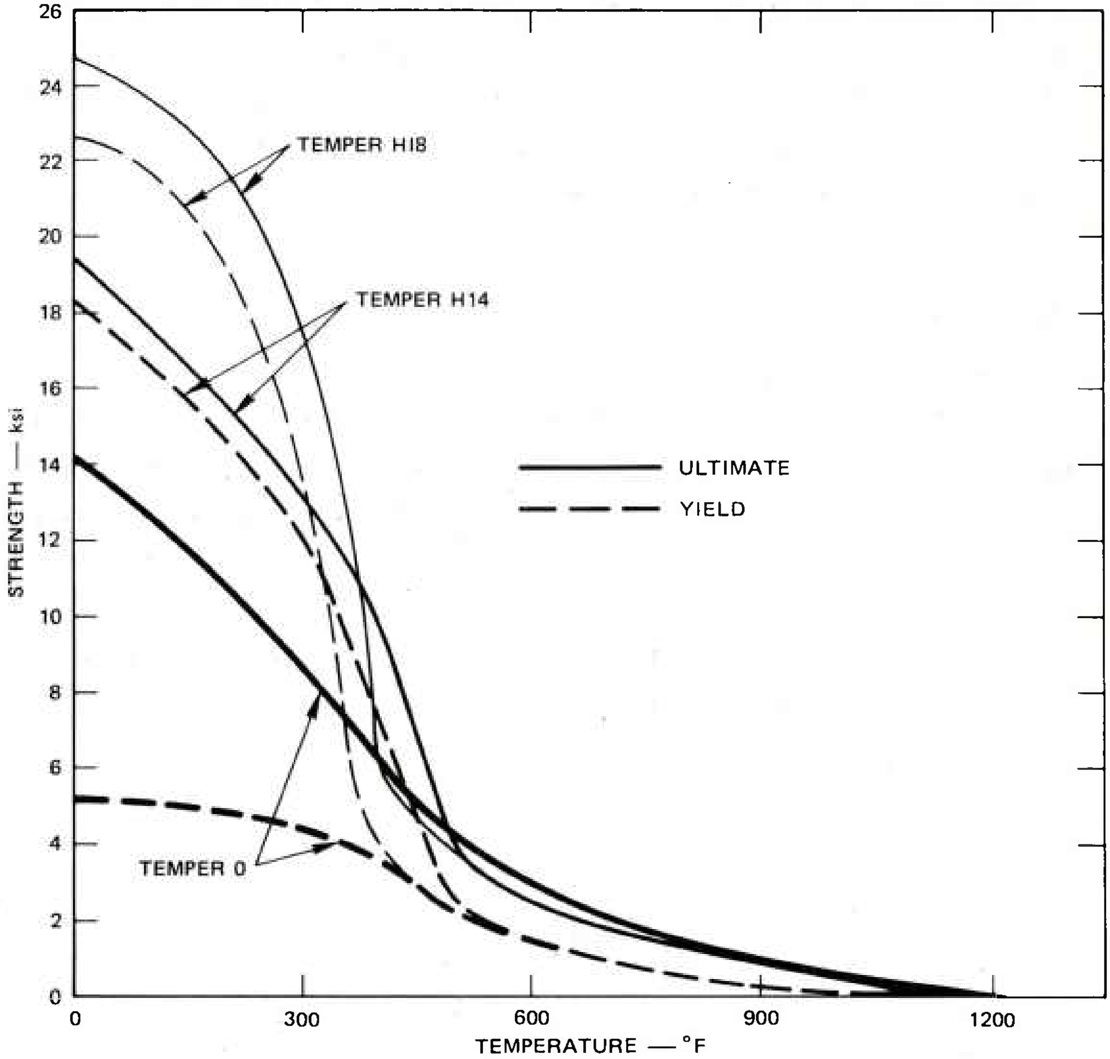
where

$$F_{ai+1} = F_{Li} - E_{Li}\,(F_{Ri} - F_{Li} + 4\Delta F \cdot E_{Ri}/\Delta E)/\Delta E$$

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

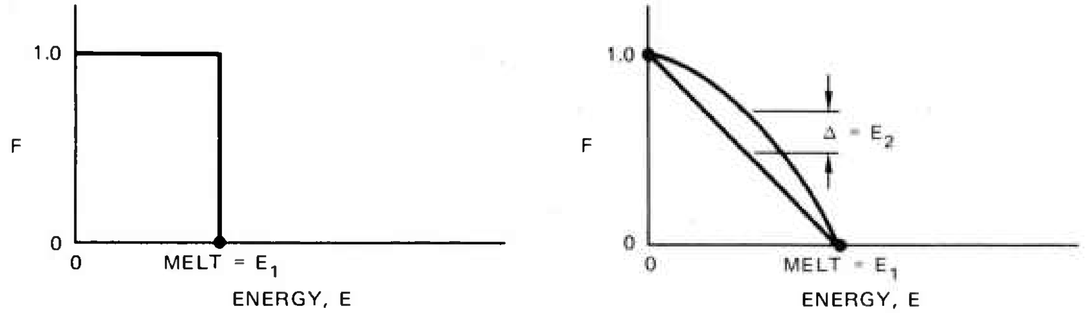
$$F_{ci+1} = -4\Delta F/\Delta E^2$$

$$\Delta E = E_{Ri} - E_{Li}$$



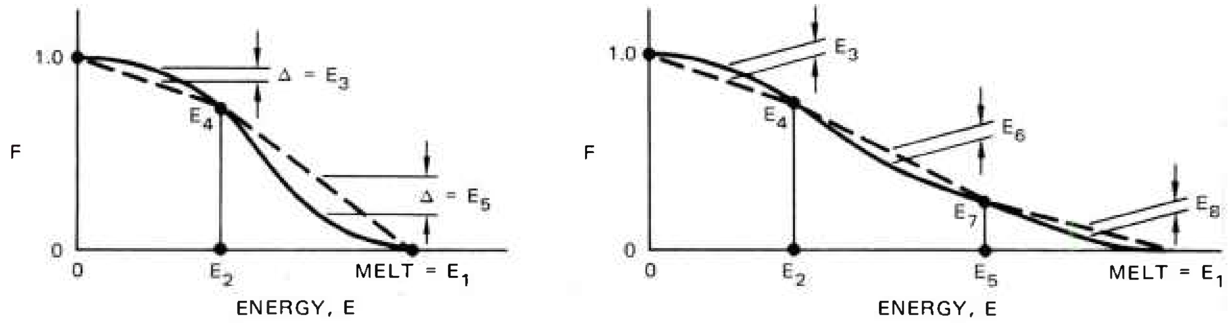
SOURCE: Metals Handbook, Vol. I,(American Society of Metals, 1961), pp. 936, 940. GA-6586-30

FIGURE D.1 VARIATION OF STRENGTH WITH TEMPERATURE FOR ALUMINUM 1100



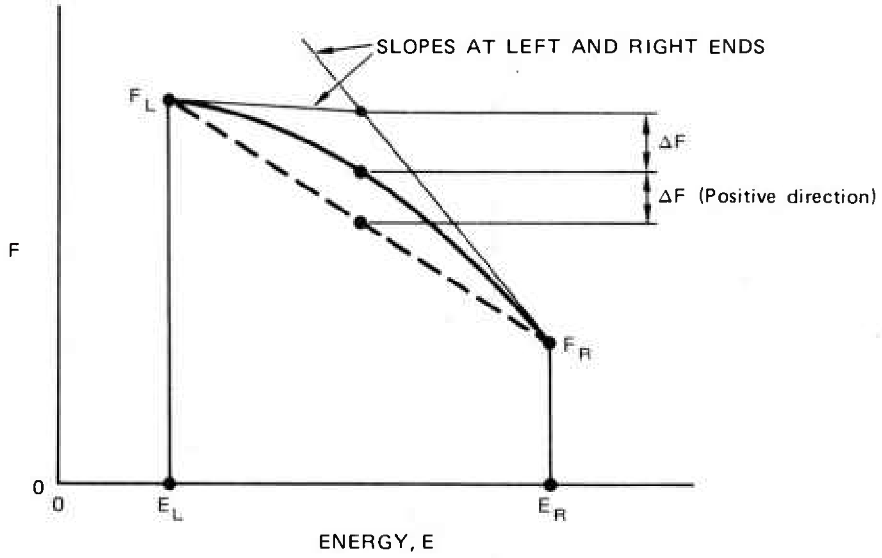
NO PARABOLIC REGIONS

ONE PARABOLIC REGION



TWO PARABOLIC REGIONS

THREE PARABOLIC REGIONS



STANDARD PARABOLA AND DEFINITION OF TERMS IN DERIVATION

MA-6802-6

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

ΔF = 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 ΔF 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,

$$\frac{\Delta X_1}{C_1} = \frac{\Delta X_2}{C_2} \tag{E.1}$$

where

ΔX_1, ΔX_2 = cell thicknesses

C_1, C_2 = 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

$$E_{NEW} = \frac{\Sigma\, E_{OLD}\rho\Delta X}{\Sigma\rho\Delta X} \qquad (E.2)$$

- 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.

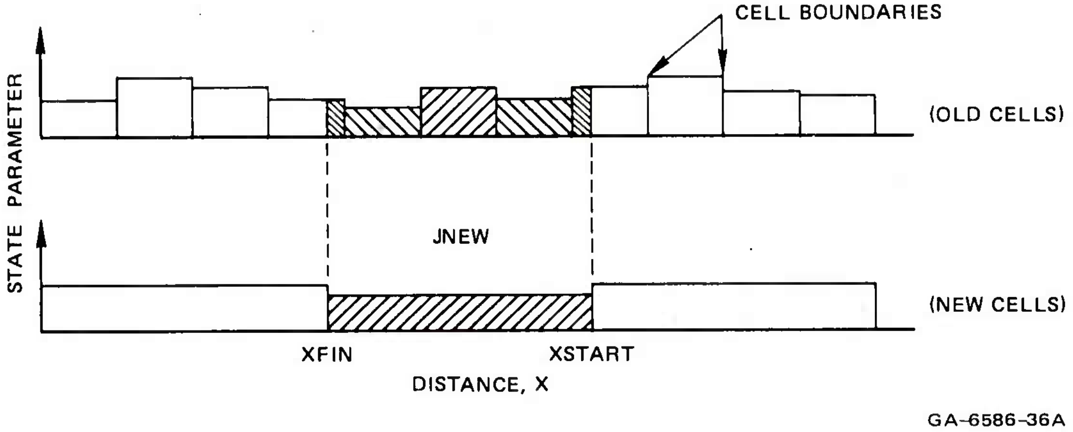


FIGURE E.1 LAYOUT FOR COMPUTING PROPERTIES AT REZONED COORDINATES

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.)

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

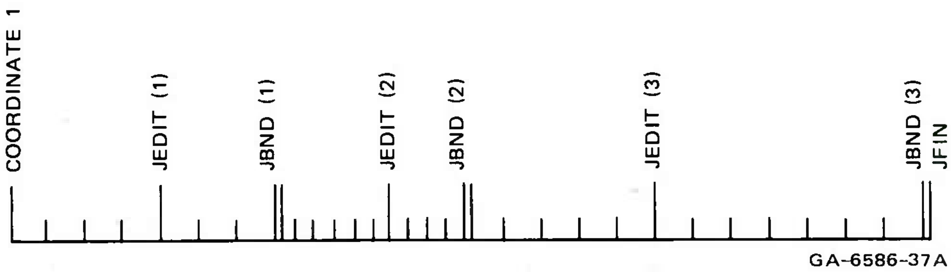


FIGURE E.2 REPRESENTATIVE LAYOUT OF COORDINATES BEFORE REZONING

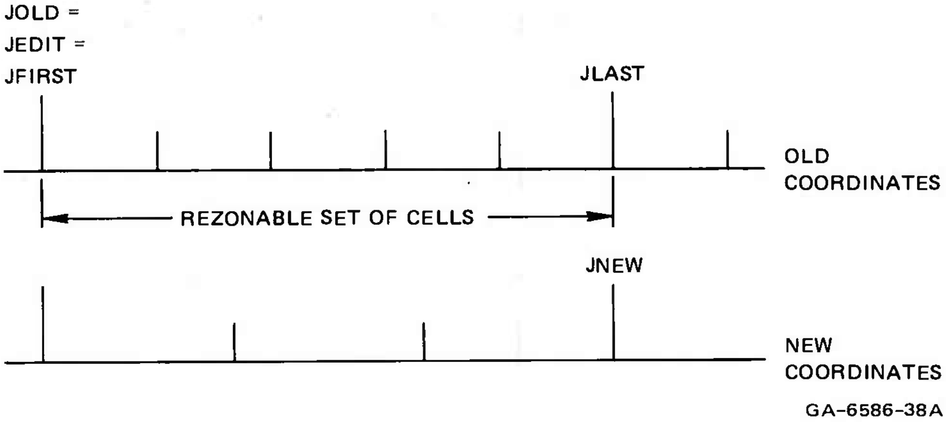


FIGURE E.3 REZONABLE SET OF CELLS TERMINATED ON LEFT BY A JEDIT

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:

$$M_{12} = M_{a12} + M_{b12} \tag{E.3}$$

$$M_{a12} = 1/2\int_{X_1}^{X_1 + \Delta X} \rho u d\xi \tag{E.4}$$

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

where

M_{a12} = one-half the momentum of the cell between coordinates 1 and 2

M_{b12} = the contribution to coordinate 1 of the variation of momentum in the cell 1-2

X_1 = the location of coordinate 1

ΔX = 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 ρu 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

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

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

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

and $M_{a12} = 1/4\rho\Delta X(U_1 + U_2)$

$M_{b12} = 1/2\rho\Delta X(U_1 - U_2)$

Hence,

$M_{12} = 1/2\rho\Delta XU_1$

$M_{21} = 1/2\rho\Delta XU_2$

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

$$U_1' = U_j + (U_{j+1} - U_j)\,\frac{X_1' - X_j}{X_{j+1} - X_j} \tag{E.8}$$

$$U_2' = U_j + (U_{j+1} - U_j)\,\frac{X_2' - X_J}{X_{j+1} - X_j} \tag{E.9}$$

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

$$\xi_1 = \frac{X_1' + X_2'}{2} - X_1 - \frac{\Delta X}{2} \tag{E.10}$$

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

$$M_{a12} = \frac{1}{4}\,\rho\,\xi_2(U_1' + U_2') \tag{E.11}$$

$$M_{b12} = -\,\frac{1}{4}\,\frac{\rho\,\xi_2}{\Delta X}\,[6\xi_1(U_1' + U_2') + \xi_2(U_2' - U_1')] \tag{E.12}$$

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

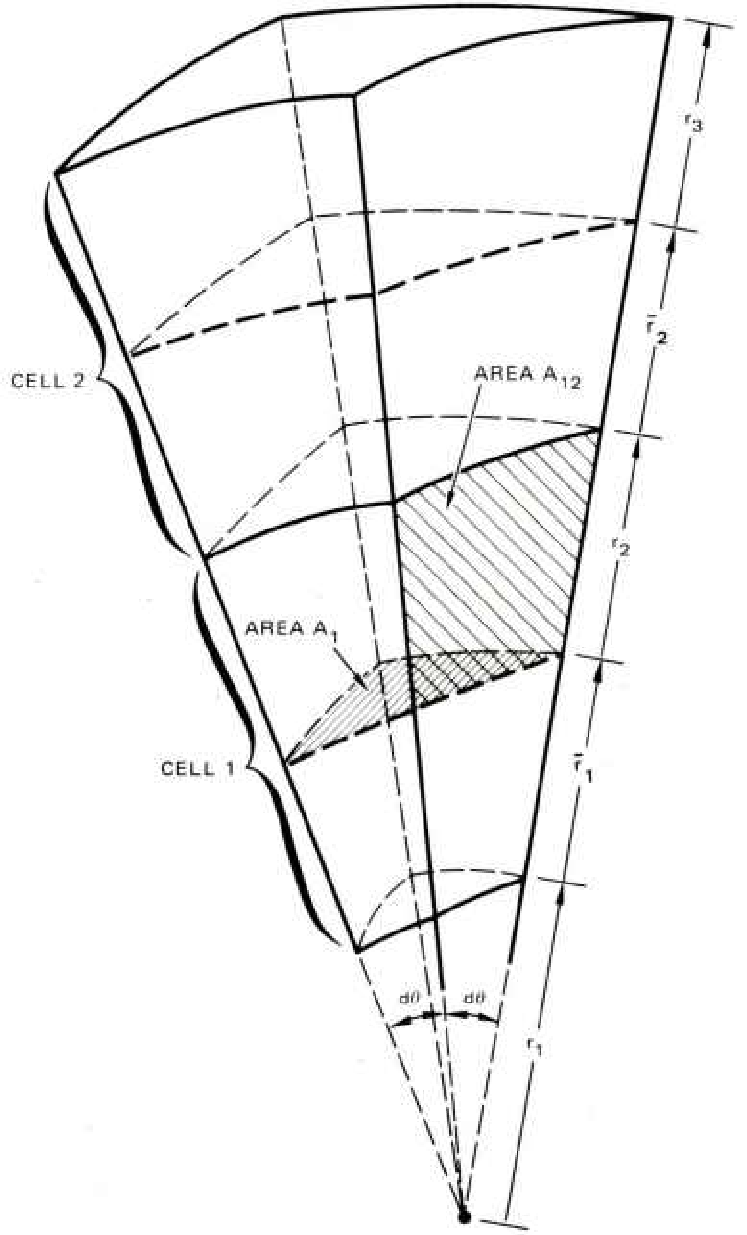
$$M_1 = \frac{\rho_o}{3} d\theta^2 \, (r_2^3 - r_1^3) \qquad \text{(F.1)}$$

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

$$\rho = \frac{3Z_1}{r_2^3 - r_1^3} \qquad \text{(F.2)}$$

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

$$\bar{r}_1^3 = \frac{r_2^3 + r_1^3}{2} \qquad \text{(F.3)}$$



MA-6802-7

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

$$A_1 = d\theta^2 \bar{r}_1^2 \tag{F.4}$$

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:

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

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

$$\begin{aligned} \sigma_{r1} d\theta^2 \bar{r}_1^2 - \sigma_{r2}\, d\theta^2 \bar{r}_2^2 + 4 \Bigg[\sigma_{\theta 1} \frac{d\theta}{2} \left(\frac{\bar{r}_1 + r_2}{2} \right) (r_2 - \bar{r}_1) d\theta \\ + \sigma_{\theta 2} \frac{d\theta}{2} \left(\frac{\bar{r}_2 + r_2}{2} \right) (\bar{r}_2 - r_2) d\theta \Bigg] = \frac{\rho d\theta^2}{3} \; \frac{r_3^3 - r_1^3}{2} \; \frac{\Delta U_2}{\Delta t} \end{aligned} \tag{F.6}$$

Here ΔU_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

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

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

$$\Delta E = V \int \sigma d\varepsilon \; = - \int P dV + V \sum_i \int \sigma'_i d\varepsilon_i \tag{F.8}$$

Through use of the stress and strain definitions in Section 4.1, the energy change reduces to

$$\Delta E = - \int \sigma_1 dV + \frac{3}{2} \int \sigma'_1 (dV + V d\varepsilon_1) \qquad \text{(F.9)}$$

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

$$M_1 = \frac{\rho_o}{2} d\theta \left(r_2^2 - r_1^2 \right) \qquad \text{(F.10)}$$

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 ρ at any time from the geometry and Z as in Eq. (F.10):

$$\rho = \frac{2Z_1}{r_2^2 - r_1^2} \qquad \text{(F.11)}$$

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

$$\bar{r}_1^2 = \frac{r_1^2 + r_2^2}{2} \qquad \text{(F.12)}$$

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

$$A_1 = \bar{r}_1 \, d\theta \qquad \text{(F.13)}$$

and the area for the circumferential stress $\sigma\theta_1$ is

$$A_{12} = d\theta(r_2 - \bar{r}_1) \tag{F.14}$$

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

$$\sigma_{r1} d\theta \cdot \bar{r}_1 - \sigma_{r2}\, d\theta \cdot \bar{r}_2 + \sigma_{\theta 1} d\theta(r_2 - \bar{r}_1)$$

$$+ \sigma_{\theta 2}\, d\theta(\bar{r}_2 - r_2) = \frac{\rho d\theta}{2}\,(r_3^2 - r_1^2)\,\frac{\Delta U_2}{\Delta t} \tag{F.15}$$

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

$$\Delta U_2 = 4\Delta t\,\frac{\sigma_{r1}\,\bar{r}_1 - \sigma_{r2}\bar{r}_2 + \sigma_{\theta 1}(r_2 - \bar{r}_1) + \sigma_{\theta 2}(\bar{r}_2 - r_2)}{Z_1 + Z_2} \tag{F.16}$$

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

$$\Delta E = - \int \sigma_1 dV + \int (\sigma_1' - \sigma_2')(dV + Vd\varepsilon_1) \tag{F.17}$$

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 ε_2 and ε_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

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

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 $\sigma_{,}'^{N}$ from

$$\sigma_1' = \sigma_1'^{N} = \sigma_{10}' + \frac{4}{3}\mu\Delta\varepsilon_1 \qquad \text{(F.19)}$$

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)

$$\sigma_1' = \frac{M\sigma_1'^{N} + 2G\,Y_o^{*}}{M + 3G} \qquad \text{(F.20)}$$

where Y_o^{*} has the same sign as σ_1' and $\left|Y_o^{*}\right| = Y_o$, the yield strength

Table F.1

STRESS AND STRAIN QUANTITIES IN ONE-DIMENSIONAL FLOW

Quantity	Definitions		
	Planar	Cylindrical	Spherical
$d\varepsilon_1$	$\frac{\partial u}{\partial x}$	$\frac{\partial u}{\partial r}$	$\frac{\partial u}{\partial r}$
$d\varepsilon_2$	0	$\frac{u}{r}$	$\frac{u}{r}$
$d\varepsilon_3$	0	0	$\frac{u}{r}$
$d\varepsilon_1'$	$\frac{2}{3}\,d\varepsilon_1$	$d\varepsilon_1 - \frac{1}{3}\frac{d\rho}{\rho}$	$d\varepsilon_1 - \frac{1}{3}\frac{d\rho}{\rho}$
$d\varepsilon_2'$	$-\frac{1}{3}\,d\varepsilon_1$	$-d\varepsilon_1 + \frac{2}{3}\frac{d\rho}{\rho}$	$-\frac{1}{2}\,d\varepsilon_1 + \frac{1}{6}\frac{d\rho}{\rho}$
$d\varepsilon_3'$	$-\frac{1}{3}\,d\varepsilon_1$	$-\frac{1}{3}\frac{d\rho}{\rho}$	$-\frac{1}{2}\,d\varepsilon_1 + \frac{1}{6}\frac{d\rho}{\rho}$
$d\bar{\varepsilon}$	$\frac{2}{3}\lvert d\varepsilon_1\rvert$	$\sqrt{\frac{4}{3}[d\varepsilon_1^{\,2} - d\varepsilon_1 \cdot \frac{d\rho}{\rho} + (\frac{1}{3}\frac{d\rho}{\rho})^2]}$	$\lvert d\varepsilon_1'\rvert$
$d\bar{\varepsilon}^P$	$\lvert d\varepsilon_1^P\rvert$	$\sqrt{\frac{4}{3}[(d\varepsilon_1^P)^2 + (d\varepsilon_2^P)^2 + d\varepsilon_1^P d\varepsilon_2^P]}$	$\lvert d\varepsilon_1^P\rvert$
$\bar{\sigma}$	$\frac{3}{2}\lvert\sigma_1'\rvert$	$\sqrt{3(\sigma_1'^{\,2} + \sigma_2'^{\,2} + \sigma_1'\,\sigma_2')}$	$\frac{3}{2}\lvert\sigma_1'\rvert$
σ_2'	$-\frac{1}{2}\,\sigma_1'$	-	$-\frac{1}{2}\,\sigma_1'$

Notes: Subscript 1 is in direction of propagation
2 is in θ 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
$\sigma, \sigma', \bar{\sigma}$	are stress, deviator stress, equivalent or Mises stress
γ, γ^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
ρ	is density.

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.

$$\Delta\varepsilon_1^P = \frac{2G\Delta\varepsilon_1 - Y_o^* + \frac{3}{2}\sigma_{10}}{M + 3G} \qquad \text{(F.21)}$$

and $$Y = Y_o + M\Delta\bar{\varepsilon}^P = Y_o + \frac{3}{2}M|\Delta\varepsilon_1^P|.$$

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

$$\Delta\varepsilon_1^P = \Delta\varepsilon_1' = \frac{2}{3}\Delta\varepsilon_1 \qquad \text{(F.22)}$$

and there is no change in the elastic deviator strain.

For cylindrical flow, two deviator stresses $\sigma_1'^N$ and $\sigma_2'^N$ are calculated from Eq. (G.26) and then $\bar{\sigma}^N$ is evaluated from Eq. (G.4). If $\bar{\sigma}^N$ exceeds Y, then $\bar{\sigma}$ is reduced from $\bar{\sigma}^N$ as follows:

$$\bar{\sigma} = \frac{M\bar{\sigma}^N + 3GY_o}{M + 3G} \qquad \text{(F.23)}$$

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

$$\sigma_1' = \sigma_1'^N \frac{Y}{\bar{\sigma}^N} \qquad \text{(F.24)}$$

$$\sigma_2' = \sigma_2'^N \frac{Y}{\bar{\sigma}^N} \qquad \text{(F.25)}$$

where $Y = Y_o + M\Delta\bar{\varepsilon}^P$.

The plastic shear strain is obtained as in Eq. (F.21):

$$\Delta\bar{\varepsilon}^P = \frac{3G\Delta\bar{\varepsilon} - Y + \bar{\sigma}_o}{3G + M} \tag{F.26}$$

and from Eqs. (G.24) and (G.25):

$$\Delta\varepsilon_i^P = \Delta\varepsilon_i' - \Delta\varepsilon_i'^E = \Delta\varepsilon_i'(1 - \frac{Y}{\bar{\sigma}^N}) \tag{F.27}$$

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

$$\sigma_1' = \frac{M\sigma_1'^N + 2GY_o^*}{M + 3G} \tag{F.28}$$

and the plastic strain is

$$\Delta\varepsilon_1^P = \frac{3G\Delta\varepsilon_1' - Y_o^* + \bar{\sigma}_o}{M + 3G} \tag{F.29}$$

Note that in spherical flow, the relations for σ_1' 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

$$\Delta E^P = V \sum_i \sigma_i' \, d\varepsilon_i^P \tag{F.30}$$

where V is specific volume and ΔE^P is the increase in specific internal energy. For planar and spherical geometries, the energy change is

$$\Delta E^P = \frac{3}{2} V \sigma_1' \, d\varepsilon_1^P \tag{F. 31}$$

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

$$\Delta E^P = V \left[\sigma_1' \, (2d\varepsilon_1^P + d\varepsilon_2^P) + \sigma_2' \, (2d\varepsilon_2^P + d\varepsilon_1^P)\right] \tag{F. 32}$$

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[45] 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

$$d\varepsilon = d\varepsilon^E + d\varepsilon^P \qquad (G.1)$$

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

$$\frac{d\varepsilon^P_{12}}{\sigma'_{12}} = \frac{d\varepsilon^P_{23}}{\sigma'_{23}} = \frac{d\varepsilon^P_{13}}{\sigma'_{13}} = \frac{d\varepsilon^P_{11}}{\sigma'_{11}} = \ . \ . \ . = d\lambda \qquad (G.2)$$

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.,

$$d\varepsilon^P_1 + d\varepsilon^P_2 + d\varepsilon^P_3 = 0 \qquad (G.3)$$

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 $\bar{\sigma} = Y$ at yield. The usual definition for the effective stress has the following forms:

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

$$= \sqrt{\frac{3}{2}\left[(\sigma'_1)^2 + (\sigma'_2)^2 + (\sigma'_3)^2\right]} \qquad \text{(G.4b)}$$

$$= \sqrt{\frac{3}{2}\left[{\sigma'_x}^2 + {\sigma'_y}^2 + {\sigma'_z}^2 + 2(\tau_{yz}^2 + \tau_{zx}^2 + \tau_{xy}^2)\right]} \qquad \text{(G.4c)}$$

$$= \sqrt{3\left[(\sigma'_1)^2 + (\sigma'_2)^2 + \sigma'_1\sigma'_2\right]} \qquad \text{(G.4d)}$$

$$= \sqrt{\frac{3}{2}\,\sigma'_{ij}\,\sigma'_{ij}}$$

where

$\sigma_1, \sigma_2, \sigma_3$ = principal stresses

$\sigma'_1, \sigma'_2, \sigma'_3$ = deviator stresses in the principal directions

$\sigma'_x, \sigma'_y, \sigma'_z$ = deviator stresses in the coordinate directions

τ = shear stress.

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

$$d\bar{\varepsilon}^P = \sqrt{\frac{2}{9}\left[d\varepsilon_1^P - d\varepsilon_2^P)^2 + (d\varepsilon_2^P - d\varepsilon_3^P)^2 + (d\varepsilon_3^P - d\varepsilon_1^P)^2\right]} \qquad \text{(G.5a)}$$

$$= \sqrt{\frac{2}{3}\left[d\varepsilon_1^P)^2 + (d\varepsilon_2^P)^2 + (d\varepsilon_3^P)^2\right]} \qquad \text{(G.5b)}$$

$$= \sqrt{\frac{2}{3}\left[(d\varepsilon_x^P)^2 + (d\varepsilon_y^P)^2 + (d\varepsilon_z^P)^2 + 2(d\varepsilon_{xy}^P)^2 + 2(d\varepsilon_{yz}^P)^2 + 2(d\varepsilon_{zx}^P)^2\right]} \quad (G.5c)$$

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

$$= \sqrt{\frac{2}{3}\, d\varepsilon_{ij}^P\, d\varepsilon_{ij}^P}$$

where

$d\varepsilon_1^P$, $d\varepsilon_2^P$, $d\varepsilon_3^P$ = plastic strains in the principal directions

$d\varepsilon_x^P$, $d\varepsilon_y^P$, $d\varepsilon_z^P$ = plastic strains in the coordinate directions

$d\varepsilon_{xy}^P$, $d\varepsilon_{yz}^P$, $d\varepsilon_{zx}^P$ = plastic shear strains (tensor components).

4. The yield condition describes yielding as a function ϕ of the second invariant J_2' of the deviator stress tensor

$$\phi = J_2' - \kappa^2 = 0 \quad (G.6)$$

where

$$J_2' = \frac{\bar{\sigma}^2}{3} = \frac{1}{6}\left[(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2\right]$$

and κ 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 ϕ. That is, Eq. (G.2) can be put into the form

$$d\varepsilon_{ij}^P = \frac{\partial\phi}{\partial\sigma_{ij}}\, d\lambda \quad . \quad (G.8)$$

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:

$$\sigma'_{ij} = 2G(\varepsilon^E_{ij} - \frac{\delta_{ij}}{3} \Sigma\, \varepsilon_{ii}) \qquad (G.9)$$

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

$$\varepsilon'_{ij} = \varepsilon_{ij} - \frac{\delta_{ij}}{3} \Sigma\, \varepsilon_{ii} \qquad (G.10)$$

Then Eq. (G.9) takes the form

$$\sigma'_{ij} = 2G\varepsilon'^{E}_{ij} \qquad (G.11)$$

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 $d\varepsilon^P_{ij} = \sigma'_{ij} d\lambda$, squared, and all the equations are added, the result is

$$\frac{9}{4}(d\bar{\varepsilon}^P)^2 = \bar{\sigma}^2 (d\lambda)^2 \qquad (G.12)$$

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

$$\sigma'_{ij} = \frac{2}{3}\bar{\sigma}\, \frac{d\varepsilon^P_{ij}}{d\bar{\varepsilon}^P} \quad . \qquad (G.13)$$

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

$$dY = M d\bar{\varepsilon}^P \qquad (G.14)$$

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 σ'_{ij}. To aid in visualizing this problem, we introduce a vector notation for both principal deviator stress and principal deviator strain:

$$\vec{\sigma} = \sigma'_1 \vec{i} + \sigma'_2 \vec{j} + \sigma'_3 \vec{k} \qquad \text{(G.15)}$$

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

$$\vec{\sigma} = 2G\vec{\varepsilon} \qquad \text{(G.16)}$$

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 $\bar{\sigma}_o = Y_o$. The equation of the ellipse is given by Eq. (G.4d) and (G.11).

$$(\varepsilon'^E_1)^2 + (\varepsilon'^E_2)^2 + \varepsilon'^E_1 \varepsilon'^E_2 = \frac{Y_o^2}{12G^2} \qquad \text{(G.17)}$$

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

$$\varepsilon_{ij} = \varepsilon^E_{ijo} + \Delta\varepsilon_{ij} \qquad \text{(G.18)}$$

---

*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.

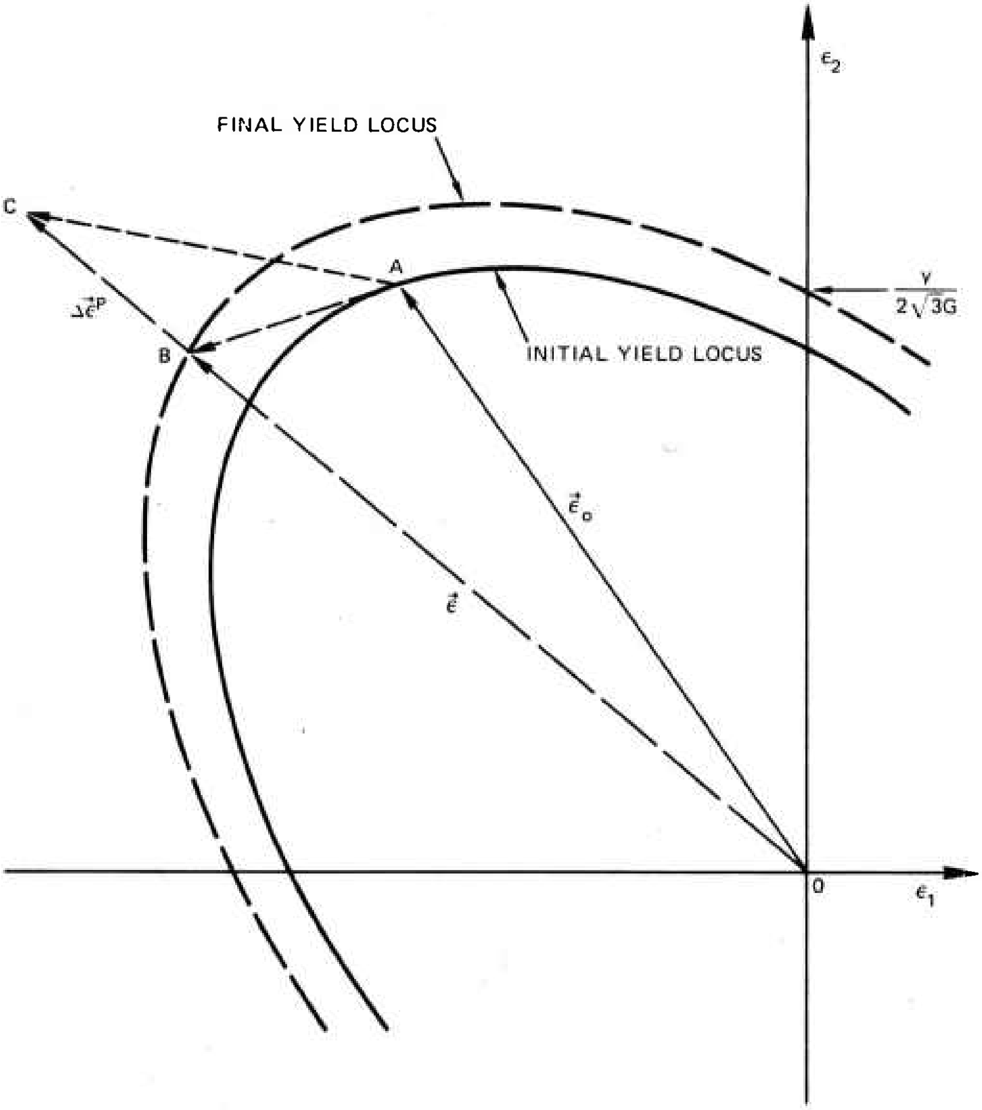


FIGURE G.1 VECTORIAL REPRESENTATION OF PRINCIPAL STRAINS DURING AN INTERVAL OF PLASTIC FLOW WITH WORK HARDENING where none of these tensors is necessarily oriented in such a way that the components are principal. When the new total strain tensor is diagonalized to obtain the principal deviator strains, 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

$$\Delta\vec{\varepsilon}^{P} = \vec{\varepsilon} - \vec{\varepsilon}^{E}$$

$$= \vec{\varepsilon} - \frac{\vec{\sigma}}{2G} \qquad (G.19)$$

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 co-axial

$$\Delta\bar{\varepsilon}^{P} = \bar{\varepsilon} - \frac{1}{3G}\left(\bar{\sigma}_{o} + M\Delta\bar{\varepsilon}^{P}\right) \qquad (G.20)$$

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

$$\bar{\varepsilon} = \sqrt{\frac{2}{3}}\,|\vec{\varepsilon}| \quad \text{and} \quad \bar{\sigma} = \sqrt{\frac{3}{2}}\,|\vec{\sigma}| \qquad (G.21)$$

Solving for $\Delta\bar{\varepsilon}^{P}$ provides

$$\Delta\bar{\varepsilon}^{P} = \frac{3G\bar{\varepsilon} - \bar{\sigma}_{o}}{M + 3G} \qquad (G.22)$$

With $\Delta\bar{\varepsilon}^P$ known, the yield value can be found from Eq. (G.14). The elastic strain is

$$\bar{\varepsilon}^E = \bar{\varepsilon} - \Delta\bar{\varepsilon}^P \tag{G.23}$$

and the individual strains and stresses are

$$\varepsilon'^E_{ij} = \varepsilon'_{ij}\,\frac{\bar{\varepsilon}^E}{\bar{\varepsilon}} \tag{G.24}$$

$$\sigma'_{ij} = \sigma'^N_{ij}\,\frac{\bar{\varepsilon}^E}{\bar{\varepsilon}} = \sigma'^N_{ij}\cdot\frac{Y}{\bar{\sigma}^N} \tag{G.25}$$

where

$$\sigma'^N_{ij} = 2G\varepsilon'_{ij} \tag{G.26}$$

and $\bar{\sigma}^N$ is calculated from the σ'^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

$$\sigma'_{ij} = \sigma'^N_{ij}\cdot\frac{M + 3G\dfrac{\bar{\sigma}_o}{\bar{\sigma}^N}}{M + 3G} \tag{G.27}$$

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.

$$d\varepsilon^P_{ij} = \frac{3d\bar{\varepsilon}^P}{2\bar{\sigma}}\,\sigma'_{ij}$$

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 $d\vec{\varepsilon}$ is calculated as if it were proportional to and in the direction of the final values of $\vec{\sigma}$ in the increment. In a more accurate calculation $d\vec{\varepsilon}^P$ would be directed toward an average $\vec{\sigma}$ 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 τ, the stress-strain relation is

$$\frac{\partial \tau}{\partial t} = G \frac{\partial \gamma}{\partial t} - \frac{\tau - Y_\tau}{T} \tag{G.28}$$

where T is the time constant, γ is the shear strain, and Y_τ is the yield stress in shear. With this form, a very rapid loading proceeds elastically, because the first two terms dominate. For gradual loading, τ must remain near Y_τ in the plastic range, so the behavior is like rate-independent plastic flow. At intermediate rates, an initial overshoot of τ above Y_τ occurs, and then τ gradually reduces to Y_τ. For computational purposes, we consider a short time interval, Δt, 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)

$$\tau = \tau_1 e^{-t/T} + (G\dot{\gamma}T + Y_\tau)(1 - e^{-t/T}) \tag{G.29}$$

where τ_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.

$$\frac{d\sigma'_{ij}}{dt} = 3G\,\frac{d\varepsilon'_{ij}}{dt} - \frac{\sigma'_{ij} - \frac{2}{3}Y\,\frac{d\varepsilon^{P}_{ij}}{d\bar{\varepsilon}^{P}}}{T} \tag{G.30}$$

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 $2Yd\varepsilon^{P}_{ij}/3d\bar{\varepsilon}^{P}$. Equation (G.30) is then integrated, holding all strain rates constant in the interval:

$$\sigma'_{ij} = \sigma'_{ijo} + \left[-\sigma'_{ijo} + \frac{2Yd\varepsilon^{P}_{ij}}{3d\bar{\varepsilon}^{P}} + 2GT\,\frac{d\varepsilon'_{ij}}{dt}\right]\left[1 - e^{-(t - t_{o})/T}\right] \tag{G.31}$$

where σ'_{ijo} and t_o 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 $d\varepsilon^{P}_{ij}/d\bar{\varepsilon}^{P}$ can be obtained from Eq. (G.13). In our calculations, the first estimate is

$$\frac{2d\varepsilon^{P}_{ij}}{3d\bar{\varepsilon}^{P}} \doteq \frac{\sigma'^{N}_{ij}}{\bar{\sigma}^{N}} \tag{G.32}$$

where the stress quantities are computed elastically. Subsequent estimates are based on the results of the evaluation of σ'_{ij} 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, 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

No.	Definition
1	Original density, g/cm^3
2	Bulk modulus (C), dyn/cm^2
3,4	D and S in the pressure equation: $P = C\mu + D\mu^2 + S\mu^3$ where μ = density/ESC (M,1) - 1
5	Shear modulus (G), dyn/cm^2
6	YADD, work hardening modulus, dyn/cm^2/(g/cm^3)
7	Initial solid density, g/cm^3
9	Grüneisen ratio
10	Initial yield strength, dyn/cm^2

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,
     1  TAPE4,TAPE7,TAPE2=400)
C
C        SRI PUFF 8, VERSION OF OCTOBER 1975
C        WRITTEN AT STANFORD RESEARCH INSTITUTE 8Y L. SEAMAN
C        CODE HANDLES FRACTURING, EXPLOSIVES, POROUS MATERIALS,
C        BAUSCHINGER EFFECT, AND STRESS RELAXATION IN RADIATION OR
C        IMPACT PROBLEMS.
C
C     MAIN PROGRAM  * * *
C        *  CALLS GENRAT TO READ DATA AND INITIALIZE ARRAYS
C        *  CALLS HYDRO FOR EACH CYCLE OF CALCULATIONS
C        *  SETS TIME STEP
C        *  CALLS EDIT AND REZONE AS REQUIRED
C        *  CALLS SCRIBE TO STORE RESULTS AND FOR TERMINAL PRINTOUT
C
      INTEGER H,POROUS,PRESS,RINTER,SOLID,SPALL
      REAL MATL,NEM,NET,NEMH,NETH
C              MISCELLANEOUS
      COMMON AZERO(1),CEF,CKS,DAVG,DELTIM,DISCPT(10),DOLD,DRHO,DTMAX,
     1  DTMIN,DTN,DTNH,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),NSEPRAT,NSPALL,NTEDT,
     4  NTEX,NTR(15),POLD,P6(20),R(30),RLAST,SLAST,SMAX,TEDIT(50),
     5  TF,TIME,TJ,TREZON,TS,T6(20),ULAST,UOLD,UZERO,XLAST,XNOW,XOLD
     I  ,XJDIT(20)
C              HALFSTEP VALUES
      COMMON DH,DHLAST,DUH,EH,PH,RH,RHLAST,SH,SHLAST,UH,UHLAST,XH,XHLAST
     1  ,NEMH,NETH
C              CONDITION INDICATORS
      COMMON INF,LINTER,MIRROR,NORMAL,POROUS,PRESS,RINTER,SOLID,SPALL
C              CELL LAYOUT
      COMMON DXX(30),JBND(30),JMAT(30),NAUTO,MATL(6,2),NLAYER,NMTRLS,
     1  THK(30)
C
C              COORDINATE ARRAYS
      COMMON/COORD/X(200),X0(200),CHL(200),DHL(200),DPDD(200),DPDE(200),
     I  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 /RAD/ SSTOP(5),START(5),SDURM,SSTOPM,NSPEC,SSJ,JSS,IPLOT(4)
     1  ,XMAX(4),XMIN(4),YMAX(4),YMIN(4),IA(7),ITITLE(24),NARZ,TARZ
C
C
100   CALL SECOND(FIRST)  $  XIN=FIRST
      CALL GENRAT
C        QUICK STOP FOR PROBLEM LAYOUT ONLY
      IF (JCYCS .LE. 0) GO TO I00
      NPERN=MAX0(NPERN,1)
      CN=NCYCS=NPERN  $  IT=MIN0(0,NTEDT-1)  $  NT=0  $  SF=0.8
      N=NR=1
C
C              CALCULATE AND STORE HYDRODYNAMIC DATA
200   CALL HYDRO
      XINL=XIN  $  CALL SECOND(XIN)  $  DELTIM=XIN-XINL
C          PERIODIC EDITS, PRINTS
      IF (MOD(N,    25) .EQ. 0) 205,210
205   CALTIM=XIN-FIRST
      WRITE(6,889)N,JSTAR,TIME,CALTIM,JTS,DTNH,SMAX,JSMAX
 210  IF (MOD(N,NEDIT) .EQ. 0 .AND. N .NE. JCYCS) CALL EDIT
      IF (LSUB(7) .EQ. 1)  GO TO 390
C
C              STORE DATA IN BUFFER
      CALL STORR
      JTS=MOD(JTS,1000)
C
C        STOP PARAMETERS
      IF (TIME .LT. TS) 304,400
304   IF (N .EQ. JCYCS) 400,305
```

```
 305  IF (X(JSMAX) - CKS) 500,400,400
C
C      ERROR FINISH
390   N=N-I
400   WRITE (6,84I)
      WRITE (6,840) 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=AMINI(SF*DTMIN,AMAX1(1.2*DTNH,.035*SF*DTMIN))
      IF (NSPEC .EQ. 0 .OR. SDURM .EQ. 1.) GO TO 530
      SOURM=1.
      DO 510 NS=1,NSPEC
      IF (TIME+DTNH .GT. START(NS) .ANO. TIME .LT. SSTOP(NS)) DTNH =
     I  AMIN1(DTNH,AMAX1(START(NS)-TIME,0.)+0.03*(SSTOP(NS)-START(NS)))
      IF (TIME-.5*DTN .LT. SSTOP(NS)) SDURM=0.
510   CONTINUE
530   CN=NCYCS=NPERN
C
C     PERIODIC REZONE
      IF (NREZON .GE. 0)  GO TO 534
      IF (TIME .LT. TREZON) GO TO 534
      IF (OTNH .GE. DTMAX)  GO TO 534
      ENARZ=NARZ  $  ENR=NR
      IF (ENARZ+TARZ .NE. 0. .AND. ENR .GT.ENARZ .AND. TIME .GT. TARZ)GO
     1TO 534
      IF (NR .EQ. 1) JCR=N
      IF (N .LT. JCR+(NR-1)*IABS(NREZON)) GO TO 534
      CALL EDIT
      CALL REZONE  $      NR=NR+1
C
C          TIME EDIT AND REZONE CALL
534   IF (IT) 560,550,535
 535  CALL EDIT  $NT=NT+I
      IF (NREZON .LE. 0)  GO TO 538
      IF (NT .EQ. NTR(NR)) 537,538
537   CALL REZONE  $      NR=NR+1
538   IF (NT .EQ. NTEDT) 540,545
540   IT=-1  $  GO TO 560
545   IT=0
550   IF (TIME+CN*DTNH .LT. TEDIT(NT+1)) 560,555
555   NCYCS=(TEOIT(NT+I)-TIME)/OTNH+1  $  CN=NCYCS
      DTNH=AMAX1((TEDIT(NT+I)-TIME)/CN,0.1*DTNH)   $   IT=1
560   N=N+1
      IF (DTNH .GE. 1.E-I4) 200,565
565   N=N-1  $  GO TO 400
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  ,8H, OTNH =1PE10.3)
841   FORMAT (/4X,28H**** CRITERION FOR STOP ****)
 889  FORMAT (5H   N=I5,8H, JSTAR=I4,7H, TIME=1PE10.3,12H, CALC TIME=
     1  F10.3,11H SECS, JTS=I4,7H  DTNH=1PEI0.3,7H  SMAX=1PE10.3,
     2  8H  JSMAX=I4/)
      END
```

SUBROUTINE BANDRLX

```
      SUBROUTINE BANDRLX(ICON,SD,Y1,ORO,COEF,N,J,M,NM,NT,DT,TSR,MUM,
     1   YO,INSR)
C
C     CALLED BY *HSTRESS* TO COMPUTE DEVIATOR STRESS FOR BAND AND
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     BAND       2     T1     T2    BEE    VM     GEE   EPS    NM    NT
C     GILMAN     3     CEE    PHI   BEE    VM     BNMO   -     NM    GAM
C      NOTE.  MEOW=MUM,   YAO=YADO,   INSR=NSR
C             NEM 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 - SD, ICON, NM. NT, YNOT.
C
C
      REAL NM,NT,NMO,NTO,MEOW,MUM
      DIMENSION TSR(6,30)
      YAO=0.6667 *YD
      YNOT=0.6667*Y1
C
      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.*MUM
C
      ICOR = ICON    $ YNOTO=YNOT
      NTO = NT    $    NMO = NM    $    SDO = SD
      NIT=4
      L=0     $ ENT=FLOAT(NIT)  $IT=0
      SIGHN = SIGN(1.,SDO)
      IF (ICON .EQ. 2)2,10
C     INITIAL CONDITIONS INSIDE ELASTIC ZONE
2     SD=SDO+COEF
      IF (ABS(SD).GT.YNOT)5,66
C     DEVIATOR LEAVES ELASTIC ZONE. CALCULATE RELAXATION
5     L = 1
      S = .5*(ABS(SDO+COEF)-YNOT)
      DELT = (SD-SIGN(YNOT,COEF))/(SD-SDO)*DT
      SIGHN= SIGN(1.,COEF)
      ENT=1.  $ SD=SDO
      GO TO 40
C     INITIAL CONDITION OUTSIDE OF ELASTIC ZONE
   10 L=2
      IT=IT+1  $SDI=SD+COEF/(2.*ENT)
      S=ABS(SDI)-YNOT   $ DELT=DT/ENT
      IF(S.LE.0.)18,11
C     AVERAGE DEVIATOR REMAINS OUTSIDE ELASTIC ZONE. CALCULATE  RELAXATI
   11 L=3
      IF(SIGHN.EQ.SIGN(1.,SDI))40,17
   13 IF(ABS(SD).GT.YNOT)14,16
   14 L=4
      IF(SIGN(1.,SD).EQ.SIGHN)15,16
C     DEVIATOR REMAINS OUTSIDE ELASTIC ZONE AFTER RELAXATION
   15 L=5
      IF(IT.EQ.NIT) 62,10
16    SD=SDI-COEF/(2.*ENT)
17    L=6
C     DEVIATOR REENTERS ELASTIC ZONE. RECALCULATE RELAXATION
   18 S = .5*(ABS(SD)-YNOT)   $YSTAR = SIGN(YNOT,SDO)
      DELT=(YSTAR-SD)/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 CROSSES OVER TO OPPOSITE SIDE OF ELASTIC ZONE. RECALCULATE
```

```
21      IF (SIGN(1.,SD).EQ.SIGHN) GO TO 62
        SIGHN = -SIGHN  $ L=7
        DELT = (SD+YSTAR)/COEF*OT
        S = .5*(ABS(SD)-YNOT)
        SD=SD-COEF    $ ENT=1.
40      ARG=4.*BEE/3./S
        IF (ARG.GT.20)42,43
42      XPO = 0.    $  GO TO 45
43      XPO = EXP(-ARG)
45      GO TO (96,51,52)INSR
C        PERFORM RELAXATION CALCULATIONS - BAND MODEL
51      TP=NT
        NT=NT+(EPS*GEE*S*(NT-NM)-1./T2*NM*XPO)*DELT
        NM=NM+(GEE*S*(TP-NM)-(1./T1+1./T2)*NM*XPO)*DELT
        GO TO 54
C        PERFORM RELAXATION CALCULATIONS - GILMAN MODEL
52      CEE=T1  $ PHI=T2  $ BNMO=GEE  $ GAM=GAMO=NT
        NM=BNMO*(1.+CEE*GAM)*EXP(-PHI*GAM)
54      SD=SO+COEF/ENT-1.333*MUM*NM*VM*XPO*DELT*SIGHN
55      GO TO (60,19,13,19,96,19,60)L
60      ICON = 2-IFIX(SIGN(1.,SD))
C       RECALCULATE YIELD STRENGTH IN CASES OF STRAIN HARDENING
62      YNOT = AMIN1(ABS(SD),YNOT+YAO*ABS(DRO))
        IF (YNOT.EQ.ABS(SD)) 64,66
64      ICON=2  $ L= L+50
66      CONTINUE
        GO TO (96,90,78)INSR
 78     OGAM = ABS(SDO+COEF-SD)/2.667/MUM
        NT=GAM=GAM+DGAM
        Y1 = 1.5 *YNOT
     90 RETURN
96      WRITE ( 6,199) INSR,L
        Y1 = 1.5 *YNOT
        RETURN
199     FORMAT (25H ERROR IN BANORLX,INSR = I5,5H, L = I5)
        ENO
```

SUBROUTINE BAUSCHI

```
      SUBROUTINE BAUSCHI(INO,SD,OS,YC,YHL,EP,EPM,HT,HC,XP,G)            BAUSCHI2
C                                                                       BAUSCHI3
C        ROUTINE PROVIOES A BAUSCHINGER EFFECT WHEN OEVIATOR CHANGES    BAUSCHI4
C        SIGN.  FUNCTION HAS THE FORM (S/SO)**N = EP/EPM                BAUSCHI5
C     INO=H(J,3),    2 = COMPRESSION,    0 = TENSION                    BAUSCHI6
C     SD =DEVIATOR STRESS, INPUT AS SDO, OUTPUT AS SDH OR SDJ           BAUSCHI7
C     DS = CHANGE IN DEVIATOR,  SD=SD+DS, INPUT AS ELASTIC CHANGE       BAUSCHI8
C     YC =NEM, CURRENT YIELO, SET TO ZERO WHEN SIGN OF DEVIATOR CHANGES BAUSCHI9
C     Y  =YHL, YIELD  (USED AT 2/3ROS ACTUAL VALUE)                     BAUSCH10
C     EP =NET(), PLASTIC STRAIN, RESET TO ZERO WHEN OEV. CHANGES SIGN   BAUSCH11
C     EPM=TSR(1), PLASTIC STRAIN AT WHICH BAUSCHINGER EFFECT CEASES     BAUSCH12
C     HT =TSR(2), WORK HARDENING MODULUS IN TENSION                     BAUSCH13
C     HC =TSR(3), WORK HARDENING MOOULUS IN COMPRESSION                 BAUSCH14
C     XP =TSR(4)=1/N IN OEFINING EQUATION, EXPONENT                     BAUSCH15
C     M  =4/3RDS ELASTIC SHEAR MODULUS, M=MU + EXMAT(M,1) * (D/RHO-1)   BAUSCH16
C                                                                       BAUSCH17
      REAL M                                                            BAUSCH18
      Y = 0.6667 *YHL                                                   BAUSCH19
      M = 1.333 * G                                                     BAUSCH20
      IF (DS*SD .GE. 0.) GO TO 100                                      BAUSCH21
C*****    BEGIN ROUTE FOR CHANGE IN DIRECTION OF LOADING           *****BAUSCH22
      IF (SO*(SO+OS) .GE. 0.) GO TO 400                                 BAUSCH23
C     STRESS HAS CHANGED SIGN.  PREPARE FOR BAUSCHINGER EFFECT          BAUSCH24
      IF (ABS(EP) .LE. 0.) GO TO 100                                    BAUSCH25
      YC=EP=0.  $  IND=1+SIGN(1.,SD+DS)  $  GO TO 300                   BAUSCH26
C*****    BEGIN ROUTE FOR CONTINUEO LOADING IN SAME OIRECTION      *****BAUSCH27
C     BRANCH TO ELASTIC PATH IF YIELO IS NOT EXCEEDED                   BAUSCH28
  100 IF (ABS(SD+DS) .LT. YC) GO TO 400                                 BAUSCH29
C     BRANCH TO BAUSCHINGER PATH IF PLASTIC STRAIN IS LESS THAN EPM     BAUSCH30
      IF (YC .LT. Y .AND. ABS(EP) .LT. EPM) GO TO 300                   BAUSCH31
C                                                                       BAUSCH32
C*****    LINEAR WORK HARDENING PATHS                              *****BAUSCH33
C     COMPRESSION                                                       BAUSCH34
  200 IF (SD .LT. 0.) GO TO 220                                         BAUSCH35
      SD=YC=Y=SD+(OS*HC+M*(Y-SD))/(HC+M)                                BAUSCH36
      DEP=(SD+OS-YC)/(M+H)  $  EP=EP+DEP                                BAUSCH37
      YHL = 1.5 * Y                                                     BAUSCH38
      RETURN                                                            BAUSCH39
C     LINEAR WORK HARDENING IN TENSION                                  BAUSCH40
  220 SO=SD+(DS*HT-M*(Y+SD))/(HT+M)                                     BAUSCH41
      YC=Y=-SO                                                          BAUSCH42
      DEP=(SD+DS-YC)/(M+H)  $  EP=EP+OEP                                BAUSCH43
      YHL = 1.5 * Y                                                     BAUSCH44
      RETURN                                                            BAUSCH45
C                                                                       BAUSCH46
C*****    BAUSCHINGER - NONLINEAR WORK HARDENING - PATH            *****BAUSCH47
C     SET INITIAL PLASTIC MOOULUS AND WORK HARDENING MODULI             BAUSCH48
  300 IF (ABS(EP) .LT. 1.E-4) GO TO 310                                 BAUSCH49
      HO=AMIN1(YC*XP/ABS(EP),1.E14)                                     BAUSCH50
      DEPA=ABS((SD+DS-SIGN(YC,SD+DS))/(HO+M))                           BAUSCH51
      HO=0.5*(HO+(YC+HO*DEPA)*XP/(ABS(EP)+DEPA))                        BAUSCH52
      GO TO 315                                                         BAUSCH53
  310 HO=ABS(SD+DS-SIGN(YC,SD+OS))/(EPM*(ABS(SD+OS)/Y)**(1./XP)-ABS(EP))BAUSCH54
  315 H=HC                                                              BAUSCH55
      IF (SD+OS .LT. 0.) H=HT                                           BAUSCH56
      L=0                                                               BAUSCH57
C         INITIAL ESTIMATES OF -EP- ANO -YC-                            BAUSCH58
      DEP=(SO+OS-SIGN(YC,SD+DS))/(HO+M)                                 BAUSCH59
      EPABS=ABS(EP+OEP)                                                 BAUSCH60
      YC=Y*AMIN1(1.,(EPABS/EPM)**XP)+H*AMIN1(ABS(DEP),AMAX1(0.,EPABS    BAUSCH61
     1  -EPM))                                                          BAUSCH62
      HO=YC*XP/EPABS                                                    BAUSCH63
      IF (EPABS .GT. EPM) HO=H                                          BAUSCH64
  330 DSE=SIGN(YC,DS)-SD                                                BAUSCH65
```

SUBROUTINE BAUSCHI (Concluded)

```
C         BEGIN ITERATIONS FOR PLASTIC STRAIN AND YIELD                 BAUSCH66
      L=L+1                                                             BAUSCH67
      DEP2=DEP                                                          BAUSCH68
      DEP=(DS-DSE+HO*DEP)/(HO+M)                                        BAUSCH69
      IF (DEP*DEP2 .LT. 0.) DEP=DEP2/3.                                 BAUSCH70
      EPABS=ABS(EP+DEP)                                                 BAUSCH71
      YC=Y*AMIN1(1.,(EPABS/EPM)**XP)+H*AMIN1(ABS(DEP),AMAX1(0.,EPABS    BAUSCH72
     1  -EPM))                                                          BAUSCH73
      HO=YC*XP/EPABS                                                    BAUSCH74
      IF (EPABS .GT. EPM) HO=H                                          BAUSCH75
      IF (L .GT. 10) GO TO 350                                          BAUSCH76
      IF (ABS(SIGN(YC,DS)-SD-DSE) .GT. 1.E6) GO TO 330                  BAUSCH77
350   SD=SIGN(YC,DS)                                                    BAUSCH78
      Y=AMAX1(Y,YC)                                                     BAUSCH79
      EP=EP+DEP                                                         BAUSCH80
      YHL = 1.5 * Y                                                     BAUSCH81
      RETURN                                                            BAUSCH82
C                                                                       BAUSCH83
C*****    ELASTIC ROUTE                                            *****BAUSCH84
400   SD=SD+DS                                                          BAUSCH85
      RETURN                                                            BAUSCH86
      END                                                               BAUSCH87
```

```
      SUBROUTINE BECOM(D,SDH,DTNP1,J,N)                                 BECOM   2
      COMMON /S2/ ALF,CO,EEN,EENP1,EPN,KS,TAUEL,TAUI,TAUN,TAUO,VELS,VMU,ALCOM   2
     1  ZAM,ZAMUSV,ZEP,ZEPDSV,ZEPMAXC,ZEPMAXS,ZEPSAVE,ZTAUY,ZTAUYMX     ALCOM   3
      DIMENSION TEMP(30)                                                BECOM   4
C                                                                       BECOM   5
      DATA    ALF,    BETA,    BOB2,   BVIOB,      CB,      CS,      EM/ BECOM   6
     1      4.E13, 5.2E10,1.B9E11, 3.85E9,  2.E-4, 8.86E5,    6.25/     BECOM   7
      DATA   RBVI,      SM,     SMA,     SMB,      VO/                  BECOM   8
     1       28.2,    0.16,    17.0, 2.3E-8, 1.18E8/                    BECOM   9
C                                                                       BECOM  10
C     *** INITIALIZATION OF FLAGS AND CONSTANTS - BASED UPON THE        BECOM  11
C         STRESS-STRAIN PROPERTIES AT TIME(N). ***                      BECOM  12
C                                                                       BECOM  13
C     *** KK .EQ. 0 - INITIAL LOADING PHASE                             BECOM  14
C         KK .EQ. 1 - UNLOADING OR RELOADING PHASES                     BECOM  15
C                                                                       BECOM  16
      C1=ZAMUSV                                                         BECOM  17
      ICONV=0                                                           BECOM  18
      ITERT=ITH=FT=1.                                                   BECOM  19
      KK=0                                                              BECOM  20
      SIGNT=SIGN(1.,TAUN)                                               BECOM  21
      IF (ZEPMAXS .GT. 0.) KK=1                                         BECOM  22
      IF (KK .EQ. 1) GO TO 2                                            BECOM  23
      SIGNE=SIGN(1.,EEN)                                                BECOM  24
      STAUO=SIGNE*TAUO                                                  BECOM  25
      STAUI=SIGNE*TAUI                                                  BECOM  26
      GO TO 3                                                           BECOM  27
    2 STAUO=SIGNT*TAUO                                                  BECOM  28
      STAUI=SIGNT*TAUI                                                  BECOM  29
C                                                                       BECOM  30
C     *** TRANSFER TO EITHER STATEMENT 90, WHICH BEGINS THE CALCULATION BECOM  31
C         OF THE PLASTIC STRESS-STRAIN PROPERTIES AT TIME(N+1), OR TO   BECOM  32
C         ONE OF THE SPECIAL CASES DETERMINED BY THE STRESS ROUTINE. ***BECOM  33
C                                                                       BECOM  34
    3 KSP1=KS+1                                                         BECOM  35
      GO TO (90,10,20,40,50), KSP1                                      BECOM  36
C                                                                       BECOM  37
C     *** KS .EQ. 1 - INITIAL CROSSING OF YIELD POINT, TAUO.  THE       BECOM  38
C         FRACTION OF THE TIME STEP IN THE PLASTIC REGIME IS            BECOM  39
C         CALCULATED. ***                                               BECOM  40
C                                                                       BECOM  41
   10 FT=(TAUEL-STAUO)/(TAUEL-TAUN)                                     BECOM  42
C       RECOMPUTE TAUEL, USING UPDATED MODULUS                           BECOM  43
      C1=AMIN1(ZAMUSV,AMAX1(ZAM-ALF*ABS(EENP1+FT*VELS),1.))             BECOM  44
      TAUEL=C1*EENP1                                                    BECOM  45
      FT=(TAUEL-STAUO)/(TAUEL-TAUN)                                     BECOM  46
      TAUN=STAUO                                                        BECOM  47
      ZTAUY=STAUO                                                       BECOM  48
      GO TO 90                                                          BECOM  49
C                                                                       BECOM  50
C     *** KS .EQ. 2 - CROSSING FROM POSITIVE TO NEGATIVE TAU OR VICE    BECOM  51
C         VERSA. CALCULATED QUANTITIES ARE THE FRACTION OF THE TIME STEPBECOM  52
C         IN THE PLASTIC REGIME, THE PLASTIC STRAIN AT THE CROSSING     BECOM  53
C         POINT, AND THE CUMULATIVE TOTAL OF THE PLASTIC STRAIN. ***    BECOM  54
C                                                                       BECOM  55
20    TAUN=TAUJ=ZTAUY=ZEPDSV=0.                                         BECOM  56
      STAUO=-STAUO                                                      BECOM  57
      STAUI=-STAUI                                                      BECOM  58
      SIGNT=-SIGNT                                                      BECOM  59
      FT=AMAX1(0.,AMIN1(1.,-TAUEL/(ZAM*VELS)))                          BECOM  60
      EPN=2.*(EENP1+FT*VELS)/3.                                         BECOM  61
      ZEPMAXS=ZEPMAXS+ABS(ZEPMAXC-EPN)                                  BECOM  62
      ZEPMAXC=EPN                                                       BECOM  63
      IF (KK.GT. 0) GO TO 30                                            BECOM  64
      KK=1                                                              BECOM  65
      ZEPSAVE=ABS(ZEPMAXC)                                              BECOM  66
   30 IF (FT .GE..001) GO TO 90                                         BECOM  67
      EPNP1=EPN                                                         BECOM  68
      GO TO 340                                                         BECOM  69
```

```
C
C     *** KS .EQ. 3 - RELOADING FROM AN ELASTIC POINT TO A PLASTIC
C         POINT.  THE FRACTION OF THE TIME STEP IN THE PLASTIC REGIME
C         IS CALCULATED. ***
C
40    FT=(TAUEL-ZTAUYMX)/(TAUEL-ZTAUY)
C       RECOMPUTE TAUEL, USING UPDATED MODULUS
      C1=AMIN1(ZAMUSV,AMAX1(ZAM-ALF*ABS(EENP1+FT*VELS),1.))
      TAUEL=C1*(EENP1-1.5*EPN)
      FT=(TAUEL-ZTAUYMX)/(TAUEL-ZTAUY)
      ZTAUY=TAUN=ZTAUYMX
      GO TO 90
C
C     *** KS .EQ. 4 - FIRST ELASTICALLY CALCULATED POINT IN UNLOADING
C         PHASE.  THE TAU AND TAUY VERSUS STRAIN CURVES CROSS BEFORE
C         TIME(N+1). ***
C
50    IF (KK .GT. 0) GO TO 54
C
C     ***   UNLOADING FROM INITIAL LOADING PHASE   ***
C
C       PLASTIC STRAIN IN FIRST PART OF TIME STEP IS FROM
C                       EPDOT = 4/3*PHI*PSI/(PHI+PSI) = A
      DTAU=TAUN-ZTAUY
      A=0.
      IF (ABS(DTAU) .LT. 1.E7) GO TO 52
      EPTOT=ABS(ZEPMAXC-EPN)
      PHI=SIGNT*AMAX1(1.E-6,AMIN1(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)/BVIOB**2))
52    TAUY1=STAUO*SQRT(1.+CO*ABS(ZEPMAXC-EPN-A*DTNP1))
      FT=DTAU/(TAUY1-ZTAUY+C1*(VELS+1.5*A*DTNP1))
      FT=AMAX1(0.,AMIN1(1.,FT))
      EC=EEN-FT*VELS
      EPNP1=EPN+A*DTNP1*FT
      ZTAUYMX=STAUO*SQRT(1.+CO*ABS(ZEPMAXC-EPNP1))
      GO TO 56
C
C     ***   UNLOADING FROM A LOADING PHASE OTHER THAN THE INIT. LOADING
C
54    DTAU=TAUN-ZTAUY
      A=0.
      IF (ABS(DTAU) .LT. 1.E7)  GO TO 55
      EPTOT=ABS(ZEPMAXC-EPN)
      PHI=SIGNT*AMAX1(1.E-6,AMIN1(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)/BVIOB**2))
55    TAUY1=STAUI*(1.-EXP(-SMA*SQRT(ABS(ZEPMAXC-EPN-A*DTNP1))))
      FT=DTAU/(TAUY1-ZTAUY+C1*(VELS+1.5*A*DTNP1))
      FT=AMAX1(0.,AMIN1(1.,FT))
      EC=EEN-FT*VELS
      EPNP1=EPN+A*DTNP1*FT
      ZTAUYMX=TAUN-C1*(FT*VELS+1.5*(EPNP1-EPN))
C
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
```

```
C
C        ***********
C     ***   BEGIN ITERATION LOOP FOR STRESS AND PLASTIC SHEAR STRAIN
C        ***********
90    EPNP1=EPN
      EPTOT=ABS(ZEPMAXC-EPN)
      IF (FT .LT. 0.) GO TO 300
      FTDT=FT*DTNP1
      FTVELS=FT*VELS
C
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=(TAUN+ZTAUY)/BVIOB
      CALL BEMOD(KK,J,SIGNT,EPN,ANM)
      PSI=RBVI/ANM*(T13/(SQRT(T13**2+1.)-1.)+T14/(SQRT(T14**2+1.)-1.))
205   EPDO=1./(PSI+1./PHI)
C        NEXT ESTIMATE OF PLASTIC STRAIN IS BASED ON EPDO
      EPNP1=EPN+EPDO*FTDT
      EPNSUM=EPN
      NTIMES=MAX1(1.,AMIN1(5.,3.*ABS(EPNP1-EPN)/(ABS(EPTOT)+1.E-12)))
      IF (ABS(EPTOT) .LT. 1.E-12) NTIMES=5
      DEPB=DEPA=ZEPDSV*FTDT/NTIMES
      DEPB=DEPA-SIGN(1.E-12,VELS)
      DEPAA=EPDO*FTDT/NTIMES
      DO 280 NNN=1,NTIMES
      ITERT=1
262   EPTOT=ABS(ZEPMAXC-EPNSUM-0.5*DEPB)
      EEC=EENP1+(1.-(NNN-0.5)/NTIMES)*FTVELS
      C1=AMIN1(ZAMUSV,AMAX1(ZAM-ALF*ABS(EEC),1.))
      TAUJ=C1*(EEC-1.5*(EPNSUM+0.5*DEPB))
      IF (KK .EQ. 0) TAUYJ=STAUO*SQRT(1.+CO*EPTOT)
      IF (KK .EQ. 1) TAUYJ=STAUI*(1.-EXP(-SMA*SQRT(EPTOT)))
      DTAU=TAUJ-TAUYJ
      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 265
      T14=(TAUJ+TAUYJ)/BVIOB
      CALL BEMOD(KK,J,SIGNT,EPNSUM+0.5*DEPB,ANM)
      PSI=RBVI/ANM*(T13/(SQRT(T13**2+1.)-1.)+T14/(SQRT(T14**2+1.)-1.))
265   EPDJ=1./(PSI+1./PHI)
      DEPBB=EPDJ*FTDT/NTIMES
      DEP=(DEPA*DEPBB-DEPAA*DEPB)/(DEPBB-DEPAA+DEPA-DEPB+1.E-12)
      LOC=265
      EPNP1=EPN+DEP
      IF (ABS(DEP-DEPBB) .LT. 0.02*ABS(DEPBB) .OR. ABS(DEPB-DEPBB)
     1 .LT. 1.E-10) GO TO 275
      IF (ITERT .GE. 20) GO TO 295
      IF (ITERT .EQ. 1) GO TO 267
      IF (ABS(DEPB-DEP) .GT. ABS(DEPA-DEP) .AND. MOD(ITERT,3) .NE. 3)
     1 GO TO 270
      GO TO 268
267   DEP=DEPBB
268   DEPA=DEPB  $    DEPAA=DEPBB
270   ITERT=ITERT+1
      DEPB=DEP
      GO TO 262
275   DEPA=DEPB   $   DEPAA=DEPBB
      EPNSUM=EPNSUM+DEP
      IF (ITERT .EQ. 1 .AND. DEP .EQ. 0.) EPNSUM=EPNSUM+DEPBB
280   CONTINUE
      EPNP1=EPNSUM
      GO TO 300
```

SUBROUTINE BECOM (Concluded)

```
295   LOC=295                                                           BECOM206
      EPNP1=EPN+EPDO*FTDT                                               BECOM207
      PRINT 1295, LOC,J,N,EPNP1,DEPA,DEPB,DEPAA,DEPBB,EPN,EPDO,FTDT     BECOM208
 1295 FORMAT(* LOC=*I4,* J,N=*2I4,*, EPNP1=*1P9E11.3)                   BECOM209
300   CONTINUE                                                          BECOM210
      C1=AMIN1(ZAMUSV,AMAX1(ZAM-ALF*ABS(EENP1),1.))                     BECOM211
      TAUJ=C1*(EENP1-1.5*EPNP1)                                         BECOM212
      EPTOT=ABS(ZEPMAXC-EPNP1)                                          BECOM213
      IF (KK .EQ. 0) TAUYJ=STAUO*SQRT(1.+CO*EPTOT)                      BECOM214
      IF (KK .EQ. 1) TAUYJ=STAUI*(1.-EXP(-SMA*SQRT(EPTOT)))             BECOM215
C                                                                       BECOM216
  310 IF (ABS(TAUJ) .GT. ABS(TAUYJ)) GO TO 330                          BECOM217
      KS=4                                                              BECOM218
      GO TO 50                                                          BECOM219
C                                                                       BECOM220
330   IF (ABS(TAUYJ) .LE. 0.) TAUYJ=SIGN(1.,TAUJ)                       BECOM221
      ZTAUY=TAUYJ                                                       BECOM222
      ZEPDSV=EPDJ                                                       BECOM223
  340 SDH=4.*TAUJ/3.                                                    BECOM224
      ZEP=EPNP1                                                         BECOM225
C                                                                       BECOM226
      RETURN                                                            BECOM227
C                                                                       BECOM228
      END                                                               BECOM229
```

SUBROUTINE BEMOD

```
      SUBROUTINE BEMOD(KK,J,SIGNT,EPNPH,ANM,DNMDEP)
C
      COMMON /S2/ ALF,CO,EEN,EENP1,EPN,KS,TAUEL,TAUI,TAUN,TAUO,VELS,VMU,
     1 ZAM,ZAMUSV,ZEP,ZEPDSV,ZEPMAXC,ZEPMAXS,ZEPSAVE,ZTAUY,ZTAUYMX
C
      DATA ANMO,C8,ANMI2,A2/2.75E6,1.E12,1.E6,1.E4/
C
C     *** SUBROUTINE BEMOD CALCULATES THE MOBILE DISLOCATION DENSITY
C         AND ITS DERIVATIVE WITH RESPECT TO PLASTIC STRAIN FOR
C         BERYLLIUM ***
C
      IF (KK .GT. 0) GO TO 10
C
C     *** LOADING PHASE ***
C
      ANM=ANMO+CB*ABS(ZEPMAXC-EPNPH)**2
      RETURN
C
C     *** UNLOADING OR RELOADING PHASE ***
C
10    EPB=ZEPMAXS-ABS(ZEPSAVE)+ABS(ZEPMAXC-EPNPH)
      ANMS=ANMO+C8*ABS(ZEPSAVE)**2
      ANM=ANMI2+(ANMS-ANMI2)*EXP(-A2*EPB**2)
      RETURN
C
      END
```

SUBROUTINE BFRACT

```
      SUBROUTINE BFRACT(LS,SXXEN,SYYEN,STTEN,TXYEN,EXX1,EYY1,ETT1,EXY1, BFRACT2
     1  P,NM,NT,VO,VOLD,DTO,E,EEST,EQSTCM,EQSTGM,ELMU,TSR,Y,YD,F,KS,JS,  BFRACT2
     2  M,NN,RHOS,DROT,ROT,FU2D,CL,CN)                                   BFRACT2
C                                                                        BFRACT2
C     NEM -- RELATIVE VOLUME OF CRACKS                                   BFRACT2
C     NET -- NUMBER OF CRACKS/UNIT VOLUME                                BFRACT2
C     T1 -- CRACK GROWTH COEFFICIENT, CM2/DYN/SEC                        BFRACT2
C     T2 -- THRESHOLD STRESS FOR GROWTH, DYN/CM2                         BFRACT2
C     T3 -- PARAMETER OF NUCLEATION DISTRIBUTION, CM                     BFRACT2
C     T4 -- NUCLEATION RATE COEFFICIENT                                  BFRACT2
C     T5 -- THRESHOLD STRESS FOR NUCLEATION                              BFRACT2
C     T6 -- DENOMINATOR OF EXPONENTIAL STRESS FUNCTION                   BFRACT2
C     T7 -- NOT USED                                                     BFRACT2
C     T8 -- THRESHOLD STRESS FOR ENTERING BFRACT                         BFRACT2
C     T9 -- SWITCH TO INDICATE WHETHER S OR SDH GOVERNS NUCLEATION       BFRACT2
C           0 STRESS GOVERNS                                             BFRACT2
C           1 DEVIATOR STRESS GOVERNS                                    BFRACT2
C     T10-- BETA, RATIO OF NO. OF FRAGMENTS TO NO. OF CRACKS             BFRACT2
C     T11-- GAMMA, RATIO OF FRAGMENT RADIUS TO CRACK RADIUS              BFRACT2
C     T12-- VALUE OF CRACK VOLUME WHICH DEFINES THRESHOLD OF COALESENCE  BFRACT2
C     T13-- TF, WHERE FRAGMENT VOLUME = TF*RF**3                         BFRACT2
C     CN -- CRACK DENSITY, NUMBER/CM3                                    BFRACT2
C     CL -- CUBE OF CRACK RADIUS,CM3                                     BFRACT2
C                                                                        BFRACT2
      DIMENSION TSR(6,30),FN(7),CL(1),CN(1),COS2TH(4),SIN2TH(4),CL3(5), BFRACT2
     1  FNUC(5),STH(5),INIT(6),VCR(6),VFR(6),VCN(6)                      BFRACT2
      REAL NM,NT                                                         BFRACT2
      DATA ALF,SMF,NANG/1.0,1.88,5/                                      BFRACT2
      IF (LS .GT. 0) GO TO 20                                            BFRACT2
C        **********                               ***********            BFRACT2
C               I N I T I A L I Z A T I O N                              BFRACT2
C        **********                               ***********            BFRACT2
C ***    INITIALIZE GENERAL ARRAYS - COS2TH, SIN2TH, ROT, CN, CL, FNUC   BFRACT2
      LS=1                                                               BFRACT2
      DO 5 I=1,6                                                         BFRACT2
      INIT(I)=0                                                          BFRACT2
      DO 5 J=1,7                                                         BFRACT2
      FN(J)=0.                                                           BFRACT2
    5 CONTINUE                                                           BFRACT2
      NANG1=NANG-1                                                       BFRACT2
      FNUC(1)=0.707107/NANG1                                             BFRACT2
      FNUC(NANG)=0.292893                                                BFRACT2
      COS2TH(1)=1.0                                                      BFRACT2
      SIN2TH(1)=0.                                                       BFRACT2
      DO 10 NG=2,NANG1                                                   BFRACT2
      FNUC(NG)=FNUC(1)                                                   BFRACT2
      TWOTH=6.2831853*FLOAT(NG-1)/FLOAT(NANG1)                           BFRACT2
      COS2TH(NG)=COS(TWOTH)                                              BFRACT2
10    SIN2TH(NG)=SIN(TWOTH)                                              BFRACT2
C ***    INITIALIZE -TSR- COEFFICIENTS FOR EACH MATERIAL                 BFRACT2
20    IF (INIT(M) .EQ. M) GO TO 25                                       BFRACT2
      TSR(M,3)=TSR(M,3)**3                                               BFRACT2
      VCR(M)=8.*(1./ELMU+1./(EQSTCM+ELMU/3.))                            BFRACT2
      VFR(M)=6.*TSR(M,13)*TSR(M,10)*TSR(M,11)**3                         BFRACT2
      VCN(M)=-TSR(M.3)*TSR(M.4)                                          BFRACT2
      INIT(M)=M                                                          BFRACT2
      PRINT 1025,M,(TSR(M,I),I=1,14),VCR(M),VFR(M),VCN(M)                BFRACT2
 1025 FORMAT(* INITIALIZE BFRACT FOR M=*I2,* TSR=* 1P7E11.3/4X,1P7E11.3/ BFRACT2
     1  * VCR,VFR,VCN=*1P3E11.3)                                         BFRACT2
25    CONTINUE                                                           BFRACT2
      IF(LS .EQ. 3)GO TO 500                                             BFRACT2
C        **********                               ***********            BFRACT2
C              C O M P U T A T I O N S                                   BFRACT2
C        **********                               ***********            BFRACT2
      IF (NM .LT. 0.) GO TO 410                                          BFRACT2
      IF (NT .EQ. 0.) FU2D=1.                                            BFRACT2
      FUO=FU2D                                                           BFRACT2
      VSO=VOLD*(1.-NM)/FUO/RHOS                                          BFRACT2
      VVO=VOLD/RHOS-VSO                                                  BFRACT2
      DV=DVO=(VO-VOLD)/RHOS                                              BFRACT2
      DOLD=RHOS/VOLD                                                     BFRACT2
      PSO=P/(VSO*FUO*DOLD)                                               BFRACT2
      R=ROT   $   PO=P                                                   BFRACT2
C ***    SET VALUES FOR MULTIPLE LOOPS IN CASE OF LARGE STRAIN           BFRACT2
C        MULTIPLE LOOPS IF STRAIN CORRESPONDS TO A STRESS GREATER THAN   BFRACT2
```

```
C          0.33*TSR(M,5)
      SDH=AMIN1(SXXEN,SYYEN,STTEN)
      NLOOP=MAX1(1.,-4.*DV*EQSTCM/VSO/TSR(M,5)+0.5,DTO*20.*TSR(M,1)*(PSO
     1  +SDH-TSR(M,2))+0.5)
      NLOOP=MIN0(NLOOP,10)
      PS=(EQSTCM/RHOS+EQSTGM*EEST)/(VSO+DV)-EQSTCM
      IF(PSO .GT. 0. .AND. PS .GT. 0.) NLOOP=1
      DPJ=0.5*(ABS(TSR(M,5))+ABS(PSO))
      NTRY=0
100   DELV=DV/NLOOP
      IF(ABS(DVO) .LT. 1.E-9) DVO=1.E-9
      EXX=EXX1/NLOOP*DV/DVO
      EYY=EYY1/NLOOP*DV/DVO
      ETT=ETT1/NLOOP*DV/DVO
      EXY=EXY1/NLOOP*DV/DVO
      VH=1./DOLD   $   YT=Y
      DE=(EEST-E)/NLOOP
      E1=E
      TEMP1=1./RHOS+EQSTGM*E/EQSTCM
      DR=DELV/DVO*DROT
      DT=DELV/DVO*DTO
      A1=3.*TSR(M,1)*DT
C
C ***    BEGIN -DO- LOOP FOR EACH STEP IN STRAIN
      DO 380 NL=1,NLOOP
      VH=VH+DELV
      DH=1./VH
      E1=E1+DE
      TEMPO=TEMP1
      TEMP1=1./RHOS+EQSTGM*E1/EQSTCM
      SDH=AMIN1(SXXEN,SYYEN,STTEN)
      VOPO=0.
      DO 120 NA=1,NANG
120   VOPO=VOPO+CN(NA)*CL(NA)
      TAUO=VFR(M)*VOPO
      VOPO=-VCR(M)*VOPO
C     *****                                                 *****
C     *****     ESTIMATE SOLID PRESSURE TO BEGIN ITERATIONS     *****
C     *****     STRAIN BASIS FOR PRESSURE ESTIMATE              *****
C     *****                                                 *****
      PS=PG=PN=EQSTCM*(TEMP1/(VSO+DELV)-1.)
      IF (P .LT. 0.) GO TO 130
C        CRACK OPENING BASIS FOR PRESSURE ESTIMATE
      PG=PSO+(DELV-TEMP1*(1.-PSO/EQSTCM)+VSO)/(VOPO-1./EQSTCM*TEMP1)
      IF (PG .GT. 0.) PG=PSO-(DELV-TEMP1*(1.-PSO/EQSTCM)+VSO)/(TEMP1/
     1   EQSTCM)
      GO TO 150
C        NUCLEATION BASIS FOR PRESSURE ESTIMATE
130   IF (DELV .GT. 0.) PN=-PSO+2.*TSR(M,5)+2.*TSR(M,6)*ALOG(ABS(DELV/
     1   VCR(M)/VCN(M)/DT/PSO))
C        GROWTH, EXPANSION, AND STRAIN BASIS FOR PRESSURE ESTIMATE
      XP=EXP(A1*AMIN1(0.,PSO+SDH-TSR(M,2)))
      PG=PSO+(DELV-VVO*XP+VVO-VSO+TEMPO/TEMP1*VSO)/(VVO*XP*(1./(PSO+SDH)
     1   +A1/2.)-VSO*VSO/EQSTCM/TEMP1)
150   PJ=AMAX1(PS,PG,PN)
      DVS=TEMP1/(1.+PJ/EQSTCM)-VSO
      COSR=COS(2.*R)
      SINR=SIN(2.*R)
C ***    COMPUTE STRESSES AT TIME(N-1) FOR EACH ANGULAR GROUP
      STH(NANG)=STTEN+PSO
      DO 170 NA=1,NANG1
170   STH(NA)=(SXXEN+SYYEN)/2.+PSO+(SXXEN-SYYEN)/2.*(COS2TH(NA)*COSR-
     1   SIN2TH(NA)*SINR)+TXYEN*(SIN2TH(NA)*COSR+COS2TH(NA)*SINR)
      SINR=SIN(2.*(R+DR))
      COSR=COS(2.*(R+DR))
      NC=0
      ETAU=0.
      IF (TAUO .GT. 0.) ETAU=EXP(A1*AMIN1(0.,PSO+SDH-TSR(M,2)))*TAUO
1220  CONTINUE
C*********                                                  ********
C                   BEGIN ITERATION LOOP
C*********                                                  ********
200   CONTINUE
      NC=NC+1
```

```
C ***    COMPUTE PRESSURE                                                BFRACT2   151
      PA=EQSTCM*(TEMP1/(VSO+DVS)-1.)                                     BFRACT2   152
      TAU=ETAU*EXP(AMIN1(2.,A1/2.*(PA-PSO)))                             BFRACT2   153
      FU1=AMAX1(0.,AMIN1(1.,(1.-TAU)/(1.-TSR(M,12))))                    BFRACT2   154
      VV=VH-FU1*(VSO+DVS)                                                BFRACT2   155
C ***    COMPUTE DEVIATOR STRESS                                         BFRACT2   156
      RED=AMAX1(0.,1.-4.*VV*DH)                                          BFRACT2   157
      RED1=AMAX1(1.-SMF*VV*DH,0.)                                        BFRACT2   158
      WS1=-.66667*(DOLD-DH)/(DOLD+DH)                                    BFRACT2   159
      BETA=2.*TXYEN*DROT/NLOOP*DV/DVO                                    BFRACT2   160
      ELMUF=RED1*2.*ELMU                                                 BFRACT2   161
      TXYE=TXYEN+ELMUF*EXY+(SYYEN-SXXEN)*DROT*DELV/DVO                   10/8/79    59
      SXXE=SXXEN+ELMUF*(EXX-WS1)+BETA                                    BFRACT2   163
      SYYE=SYYEN+ELMUF*(EYY-WS1)-BETA                                    BFRACT2   164
      STTE=STTEN+ELMUF*(ETT-WS1)                                         BFRACT2   165
      WS4=SXXE**2+SYYE**2+STTE**2+2.*TXYE**2                             BFRACT2   166
      YE=YT*F*RED                                                        BFRACT2   167
      IF (WS4 .LE. YE**2/1.5) GO TO 230                                  BFRACT2   168
      WS3=YE/SQRT(1.5*WS4)                                               BFRACT2   169
      SXXE=SXXE*WS3                                                      BFRACT2   170
      SYYE=SYYE*WS3                                                      BFRACT2   171
      TXYE=TXYE*WS3                                                      BFRACT2   172
      STTE=STTE*WS3                                                      BFRACT2   173
230   CONTINUE                                                           BFRACT2   174
C ***    COMPUTATION OF CRACK VOLUME FROM ELASTIC OPENING, GROWTH,       BFRACT2   175
C                NUCLEATION AND FRAGMENTATION                            BFRACT2   176
      VVA=0.                                                             BFRACT2   177
      TAU=0.                                                             BFRACT2   178
      DO 250 NA=1,NANG                                                   BFRACT2   179
      IF (NA .LT. NANG) GO TO 237                                        BFRACT2   180
      STHW=STTE+PA                                                       BFRACT2   181
      GO TO 240                                                          BFRACT2   182
237   STHW=PA+(SXXE+SYYE)/2.+(SXXE-SYYE)/2.*(COS2TH(NA)*COSR-SIN2TH(NA)* BFRACT2   183
     1   SINR)+TXYE*(SIN2TH(NA)*COSR+COS2TH(NA)*SINR)                    BFRACT2   184
240   SAVG=(STH(NA)+STHW)/2.                                             BFRACT2   185
      DTC=CN(NA)*DH/DOLD*CL(NA)                                          BFRACT2   186
      IF (SAVG .LT. TSR(M,2)) DTC=DTC*EXP(A1*(SAVG-TSR(M,2)))            BFRACT2   187
      SCN=SAVG-TSR(M,9)*(PSO+PA)/2.-TSR(M,5)                             BFRACT2   188
      DTN=0.                                                             BFRACT2   189
      IF (SCN .LT. 0.) DTN=TSR(M,4)*EXP(SCN/TSR(M,6))*DT*FNUC(NA)        BFRACT2   190
     1   *TSR(M,3)                                                       BFRACT2   191
      IF (STHW .LT. 0.) VVA=VVA-VCR(M)*STHW*(DTC+DTN)                    BFRACT2   192
250   TAU=TAU+DTC+DTN                                                    BFRACT2   193
      VVA=VVA/DH                                                         BFRACT2   194
      TAU=VFR(M)*TAU                                                     BFRACT2   195
      FU1=AMAX1(0.,AMIN1(1.,(1.-TAU)/(1.-TSR(M,12))))                    BFRACT2   196
C ***    COMPUTE CHANGES IN V AND IN V SUB S                             BFRACT2   197
      SDH=AMIN1(SXXE,SYYE,STTE)                                          BFRACT2   198
      DVSA=DVS                                                           BFRACT2   199
      DELVA=DVS+VVA-VVO                                                  BFRACT2   200
      PJ=PA                                                              BFRACT2   201
C                                                                        BFRACT2   206
C ***    TEST FOR COMPLETION OF ITERATIONS                               BFRACT2   207
      IF (ABS(DELVA-DELV)/VSO .LT. 2.E-5) GO TO 300                      BFRACT2   208
      IF (NC .GE. 30) GO TO 450                                          BFRACT2   209
C        DELVA IS RECENT VALUE, DELVB IS LARGER STORED VALUE, AND        BFRACT2   210
C        DELVC IS SMALLER STORED VALUE.                                  BFRACT2   211
      IF (NC .EQ. 1) GO TO 270                                           BFRACT2   212
      IF(NC.EQ.2)GO TO 260                                               BFRACT2   213
      IF (DELVC .GT. DELV) GO TO 265                                     BFRACT2   214
      IF (DELVB .LT. DELV) GO TO 260                                     BFRACT2   215
      IF (DELVA .GT. DELV) GO TO 265                                     BFRACT2   216
C        INTERPOLATION TO FIND DVS                                       BFRACT2   217
260   DVS=DVSA+(DVSB-DVSA)/(DELVB-DELVA)*(DELV-DELVA)                    BFRACT2   218
      IF (MOD(NC+2,3) .EQ. 0) DVS=0.5*(DVSA+DVSB)                        BFRACT2   219
      GO TO 280                                                          BFRACT2   220
265   DVS=DVSA+(DVSC-DVSA)/(DELVC-DELVA)*(DELV-DELVA)                    BFRACT2   221
      IF (MOD(NC+2,3) .EQ. 0) DVS=0.5*(DVSA+DVSC)                        BFRACT2   222
      GO TO 280                                                          BFRACT2   223
270   PJ=PA+(DELV-DELVA)/(VVA*(1./(PA+SDH/2.)+A1/2.)-TEMP1/(EQSTCM+PA))  BFRACT2   224
      IF (PJ .LT. 0. .OR. PA .GE. 0.) GO TO 279                          BFRACT2   225
```

```
      PJ=PA+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/EQSTCM)-VSO
 280  IF (NC .GT. 2) GO TO 285
      IF(NC.EQ.1)GO TO 290
      IF (DELVA .LT. DELVB) 293,289
285   IF (DELVA .GT. DELVB .OR. DELVA .LT. DELVC) GO TO 287
      IF (DELVA .LT. DELV) 293,290
287   IF (DELVB .LT. DELV .AND. DELVA .GT. DELVB) GO TO 289
      IF (DELVC .GT. DELV .AND. DELVA .GT. DELVC) 292,200
  289 DELVC=DELVB
      DVSC=DVSB
  290 DELVB=DELVA
      DVSB=DVSA
      GO TO 200
  292 DELVB=DELVC
      DVSB=DVSC
  293 DELVC=DELVA
      DVSC=DVSA
      GO TO 200
C
C        ENDING ROUTINE
300   CONTINUE
      NT=0.
      R=R+DR
      DO 320 NA=1,NANG
      IF (NA .LT. NANG) GO TO 307
      STHW=STTE+PJ  $  GO TO 310
307   STHW=PJ+(SXXE+SYYE)/2.+(SXXE-SYYE)/2.*(COS2TH(NA)*COSR-SIN2TH(NA)*
     1   SINR)+TXYE*(SIN2TH(NA)*COSR+COS2TH(NA)*SINR)
310   SAVG=(STH(NA)+STHW)/2.
      STH(NA)=STHW
      SCN=SAVG-TSR(M,9)*(PSO+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 320
      CL(NA)=(CNO*CL(NA)*EXP(A1*AMIN1(SAVG-TSR(M,2),0.))+
     1   DN*TSR(M,3))/CN(NA)
      NT=NT+CN(NA)
320   CONTINUE
350   NM=(VVA+(1.-FU1)*(VSO+DVS))*DH
      FU2D=FU1
      PSO=PJ
      IF (FU1 .LT. 0.01) GO TO 400
      PJ=PJ*FU1*(VSO+DVS)*DH
      SXXEN=SXXE
      SYYEN=SYYE
      STTEN=STTE
      TXYEN=TXYE
      P=PJ
      Y=YT
C        **********
C        END OF SUBCYCLING LOOP
C        **********
      VVO=VVA
      VSO=VSO+DVS
      FUO=FU1
380   DOLD=DH
      ROT=R
      IF(LS .EQ. 2)GO TO 500
      RETURN
C
C        END WITH SEPARATION
 400  CONTINUE
      SXXEN=0.
      SYYEN=0.
      STTEN=0.
      TXYEN=0.
      P=0.
      Y=YT
      NM=-ABS(NM)
      RETURN
```

```
  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.0.) P=0.
      IF(LS.EQ.2)GO TO 500
      RETURN
C
C ***    PROVISION FOR ABORT IN CASE OF ITERATION FAILURE
 450  NTRY=NTRY+1
      IF (NTRY .GE. 5) GO TO 460
      DV=VO/RHOS-1./DOLD
      NLOLD=NLOOP
      NLOOP=MAX1(3.,-4.*2.**NTRY*DV*EQSTCM/VSO/TSR(M,5)+0.5,2.*NLOOP)
      IF(TSR(M,6).GT.0.)NLOOP=MIN1(AMAX1(3.,4.*2**NTRY*DV*EQSTCM/VSO
     1    /3.E9+0.5,2.*NLOLD),10.*NLOLD)
      GO TO 100
 460  PRINT 1600,NN,KS,JS,SDH,P,DV,DELVA,DELVB,DELV,DVO,VO
      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.*(COS2TH(NA)*COSR-SIN2TH(NA)*
     1    SINR)+TXYE*(SIN2TH(NA)*COSR+COS2TH(NA)*SINR)
 610  SAVG=(STH(NA)+STHW)/2.
      SCN=SAVG-TSR(M,9)*(PSO+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=(VVA+(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=YT
      VVO=VVA
      VSO=VSO+DVS
      DOLD=DH
      ROT=R
      IF(LS .EQ. 2)GO TO 500
      RETURN
C
C        FINAL PRINTOUT
C
 500  IZERO=1
      IF (NT .EQ. 0.) GO TO 520
      IZERO=2
      CNSUM=0.
      CRIT2=0.
      CRIT3=0.
      DO 510 NA=1,NANG
      CL3(NA)=CL(NA)**(.3333333333)
      CRIT2=CRIT2+CN(NA)*CL3(NA)**2
      CNSUM=CNSUM+CN(NA)
  510 CRIT3=CRIT3+CN(NA)*CL(NA)
      IF(CNSUM .EQ. 0.) GO TO 520
      CRIT2=3.1416*CRIT2
```

SUBROUTINE BFRACT (Concluded)

```
      RAD=(CRIT3/CNSUM)**(1./3.)                                        BFRACT2   376
      FRAGRAD=0.                                                        BFRACT2   377
      FRAGNUM=0.                                                        BFRACT2   378
      IF(FU2D .EQ. 1.)GO TO 515                                         BFRACT2   379
      FRAGRAD=RAD*TSR(M,11)                                             BFRACT2   380
      FRAGNUM=CNSUM*TSR(M,10)*(1.-FU2D)                                 BFRACT2   381
  515 CONTINUE                                                          BFRACT2   382
      PRINT 1510,(CL3(I),I=1,5),RAD,CRIT2,ROT,FU2D,KS,JS,(CN(I),I=1,5), BFRACT2   383
     1  CNSUM,FRAGRAD,FRAGNUM                                           BFRACT2   384
 520  CONTINUE                                                          BFRACT2   385
      RETURN                                                            BFRACT2   386
 1510 FORMAT(13HOCELL    CL = 1P4E10.3,2X,E10.3,11H  CL-AVG = E10.3,    BFRACT2   387
     1 12H  PI*N*R**2=0PF6.0,6H  ROT=F6.0,5H  FU=F6.4/2I3,7H  CN =      BFRACT2   388
     2 1P4E10.3,2X,E10.3,10H  CN-TOT =E10.3,16H  FRAGMENT RAD.=E10.3,   BFRACT2   389
     3 6H  NO.=E10.3)                                                   BFRACT2   390
 1600 FORMAT(32H ITERATION FAILURE IN BFRACT, N=I5,4H, K=I3,4H, J=I3,   BFRACT2   391
     1  1P5E12.3/5X,1P3E12.3)                                           BFRACT2   392
      END                                                               BFRACT2   393
```

SUBROUTINE CAP1

```
      SUBROUTINE CAP1(LS,IN,M,N,IH,DH,DORG,E,EX,EY,EZ,EXY,SX,SY,
     1  SZ,SXY,ZEVP,K,J,TEVP)
C
C CAP1 - WRITTEN BY L. SEAMAN - INSERTED INTO THE COPS CODE 10 - 78
C
C ********                                                   ***********
C       DEFINITION OF INDICATOR -IH-              5 - ELASTIC
C                  6 - MOHR-COULOMB SURFACE       7 - CAP SURFACE
C                  8 - CAP AND MOHR-COULOMB       9 - CONSOLIDATED
C                 10 - SEPARATION
C
      INTEGER DBUG1,DBUG2
      REAL MUP,MUP2
      DIMENSION INIT(4),AMC1(4),AMC2(4),AMC3(4),AMC4(4),AMC5(4),
     1           AMC6(4),AK(4),AK2(4),MUP(4),MUP2(4),NREG(4),DAMG(4),
     2  SCRIT(4),W2(4),AKSOL(4)
      DIMENSION PA(5),AJ(5),DL(2)
      COMMON /EQS/ EQSTC(6),EQSTD(6),EQSTE(6),EQSTG(6),EQSTH(6),EQSTN(6)
     1 ,EQSTS(6),RHO(6),RHOS(6),YC(6),YAD(6),MU(6),ESC(6,20),CLIN,CQSQ,
     2  TRIQ,AMAT(6,4),SP(6),G2(6),PMIN(6)
      COMMON /POR/ PORA(6,5),PORB(6,5),PORC(6,5),EVP(6,5)
      DATA INIT/4*0/
      GH(X)=MUP(M)+MUP2(M)*AMIN1(AMC1(M),X)
      GG(X,Y)=MUP(M)+MUP2(M)*AMIN1(AMC1(M),0.5*(X+Y))
      BKK(X,Y)=AMIN1(AKSOL(M),AK(M)+AK2(M)*AMIN1(0.,.5*(X+Y)))
      BKH(X)=AMIN1(AKSOL(M),AK(M)+AK2(M)*AMIN1(0.,X))
C
      IF (LS) 30,30,50
C ********                                                   ***********
C       READ AND INITIALIZE MATERIAL ARRAYS.
C ********                                                   ***********
   30 IF(INIT(M) .EQ. M) GO TO 50
      INIT(M)=M
      READ (IN,1020)A1,A2,AK(M),A3,A4,AK2(M),A5,A6,MUP(M),A7,A8,MUP2(M)
      PRINT 1040,    A1,A2,AK(M),A3,A4,AK2(M),A5,A6,MUP(M),A7,A8,MUP2(M)
      PRINT 1021,IN
      READ (IN,1022)A1,A2,AMC1(M),AMC2(M),AMC3(M),AJ10,EN
      PRINT 1042,    A1,A2,AMC1(M),AMC2(M),AMC3(M),AJ10,EN
      AMC4(M)=-(AMC1(M)+AMC2(M)*EXP(AJ10/AMC3(M)))*EXP(-EN)
      AMC5(M)=AJ10/EN
      AMC6(M)=AJ10
      PRINT 1021,IN
      READ (IN,1020)A1,A2,SCRIT(M),A3,A4,DAMG(M),A5,A6,AKSOL(M)
      IF (AKSOL(M) .LT. AK(M)) AKSOL(M)=2.*AK(M)
      PRINT 1040,    A1,A2,SCRIT(M),A3,A4,DAMG(M),A5,A6,AKSOL(M)
      PRINT 1021,IN
      READ (IN,1022)A1,A2,(EVP(M,I),I=1,5)
      PRINT 1042,    A1,A2,(EVP(M,I),I=1,5)
      PRINT 1021,IN
      READ (IN,1024)A1,A2,NREG(M),A3,A4,NPRCAP,A5,A6,P1,A7,A8,W2(M)
      PRINT 1044,    A1,A2,NREG(M),A3,A4,NPRCAP,A5,A6,P1,A7,A8,W2(M)
      PRINT 1021,IN
C        COMPUTATION OF PARAMETERS ON HYDROSTAT
       PORA(M,1)=P1
      PORB(M,1)=0.
      PORC(M,1)=0.
      NP=MIN0(NREG(M),4)
      DO 15 NQ=1,NP
      READ (IN,1020)A1,A2,P2,A3,A4,DELP
      PRINT 1040,    A1,A2,P2,A3,A4,DELP
      PRINT 1021,IN
      DE=EVP(M,NQ+1)-EVP(M,NQ)
      DP=4.*DELP/DE
      PORA(M,NQ+1)=P1-EVP(M,NQ)/DE*(P2-P1+DP*EVP(M,NQ+1))
      PORB(M,NQ+1)=(P2-P1+DP*(EVP(M,NQ)+EVP(M,NQ+1)))/DE
      PORC(M,NQ+1)=-DP/DE
   15 P1=P2
       EVP(M,5)= EVP(M,NP+1)
C        SET ACCURACY CRITERIA
      NMAX=30
      FCR1=60.
      FCR2=300.
      DFCR1=1.E5
C          SET LPATH=0 FOR CONSTANT VOLUME ON M-C, =1 FOR NORMALITY
      LPATH=1
```

```
      PRINT 1004,FCR1,FCR2,DFCR1,NMAX,LPATH
1004  FORMAT (* ACCURACY AND SUBCYCLING CRITERIA FCR1,FCR2 =*1P2E10.3,
     1  * DFCR1,NMAX=* E10.3,I4,/* MOHR-COULOMB PATH, LPATH=*I2,
     2  * - 0 - FOR CONSTANT VOLUME, - 1 - FOR NORMALITY*)
C
      DBUG1=0
      DBUG2=1
      IF(DBUG2.EQ.0) GO TO 41
      DO 31 I=1,5
      PA(I)=(I+1)*PORA(M,1)/2.
31    AJ(I)=0.
      PRINT 3000,(PA(I),I=1,5)
3000  FORMAT (*1MOHR-COULOMB AND CAP COORDINATES*/20X,*J1*,9X,*M-C*,
     1  9X,*J2 ON CAPS CORRESPONDING TO PR VALUES*/32X,*J2*,1P5E12.3)
      II=0
      AJ2=0.
      PRINT 3001,II,AJ10,AJ2
3001  FORMAT (I10,1P7E12.3)
      DP=0.2*PORA(M,1)
      DO 32 I=1,5
32    PA(I)=PA(I)**2
      DO 40 II=1,50
      AJ1=AJ10+II*DP
      AJ2=AMC1(M)+AMC2(M)*EXP(AJ1/AMC3(M))+AMC4(M)*EXP(AJ1/AMC5(M))
      IF (AJ1 .GT. 0.) GO TO 38
      PP=AJ1**2/9.
      DO 35 JJ=1,5
      AJ(JJ)=0.
      IF (PP .GE. PA(JJ)) GO TO 35
      AJ(JJ)=SQRT(W2(M)*(PA(JJ)-PP))
35    CONTINUE
38    CONTINUE
      PRINT 3001,II,AJ1,AJ2,(AJ(I),I=1,5)
40    CONTINUE
   41 CONTINUE
      RETURN
C
C ********                                                   ***********
C           COMPUTATION OF STRESS
C ********                                                   ***********
 50   AJ10=SX+SY+SZ
      P0=AJ10/3.
      NRE=0
      DOLD=DORG
      EPRAT1=EPRAT2=0.25
      RR=1.
      RSUM=0.
      IF (DBUG1.EQ.1) PRINT 1052,N,K,J,IH,SX,SY,SZ,SXY,EX,EY,EXY,TEVP,
     1  ZEVP,DH,DORG
1052  FORMAT (*0BEGIN CAP N,K,J,IH=*4I4,* SX,SY,SZ,SXY=*1P4E10.3,
     1  * EX,EY,EXY=*3E10.3/ 10X,* TEVP,ZEVP=*2E10.3,* DH,DORG=*0P2F10.6
     2 )
      ZEVT=ALOG(DORG/DH)
      EZ=ZEVT-EX-EY
      DEVT=ZEVT
C           RECOMBINATION OF SPALLED MATERIAL.
      IF (IH .NE. 10) GO TO 80
      TEVP=TEVP+ZEVT
      IF(TEVP.GT.0.) GO TO 580
      DE3=(ZEVT-TEVP)/3.
      EX=EX-DE3
      EY=EY-DE3
      EZ=EZ-DE3
      ZEVT=TEVP
      TEVP=-1.
      IH=5
80    CONTINUE
      AJ20=SQRT(((SX-P0)**2+(SY-P0)**2+(SZ-P0)**2)/2.+SXY**2)
      EV=DEVT/3.
C ********                                                   ***********
C           COMPUTE STRESSES ON ELASTIC BASIS AND TEST FOR YIELDING.
      BG=GH(AJ20)
      SXT=(SX-P0)+2.*BG*(EX-EV)
      SYT=(SY-P0)+2.*BG*(EY-EV)
      SZT=(SZ-P0)+2.*BG*(EZ-EV)
```

SUBROUTINE CAP1 (Continued)

```
      SXYT=SXY+2.*BG*EXY
      AJ2T=SQRT((SXT**2+SYT**2+SZT**2)/2.+SXYT**2)
      ZEIT=(AJ2T-AJ20)/(2.*BG)
      IF (MUP2(M) .EQ. 0.) GO TO 95
      AJ2T=(MUP(M)/MUP2(M)+AJ20)*EXP(2.*MUP2(M)*ZEIT)-MUP(M)/MUP2(M)
95    BG=GG(AJ2T,AJ20)
      AJ2T=AMIN1(AJ2T,AJ20+2.*BG*ZEIT)
      BK=(AK(M)+AK2(M)*AJ10)*(1.+1.5*AK2(M)*ZEVT*(1.+AK2(M)*ZEVT))
      BK= AMIN1(AKSOL(M),AMAX1(BK,AK(M)))
      AJ1T=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/
     1  AMC5(M)))
      AMC=AMC1(M)
      IF(AJ1T.LT.-5.*(AMC3(M)+AMC5(M)))GOTO110
      AMC=-AJ1T
      IF(AJ1T.GT.AMC6(M))GOTO110
      AMC=       AMC1(M)+AMC2(M)*EXP(AJ1T/AMC3(M))+AMC4(M)*EXP(AJ1T/
     1  AMC5(M))
  110 F1T=AJ2T-AMC
C
C        COMPUTATION OF F2T.
      CALL CAPPR(PT,ZEVP,M,DOLD,IHH,YM)
      F2T=AJ2T**2/W2(M)+AJ1T**2/9.-PT**2
      F20=AJ20**2/W2(M)+AJ10**2/9.-PT**2
      IF (DBUG1 .EQ. 1)
     1PRINT 1145,AJ1T,AJ2T,AJ10,AJ20,BK,BG,ZEVT,ZEIT,F1T,F2T,F10,F20,PT
1145  FORMAT (* -145- J1T,J2T,J10,J20=*1P4E10.3,* BK,BG,ZEVT,ZEIT=*
     1  4E10.3,/ 10X,*F1T,F2T,F10,F20=*4E10.3,* PT=*E10.3)
C
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.OR.DBUG1.EQ.1) PRINT 1195,K,J,N,NINC,AJ1T,AJ2T,AJ10,
     1  AJ20
1195  FORMAT ( 30H CAP SUBCYCLING - K,J,N,NINC =3I4,I10,* J1T,J2T,J10,J2
     10=*1P4E10.3)
      RR=1./AMINO(600,NINC)
      NINC=0
C
C        SET INITIAL STRAIN INCREMENT.
200   NINC=NINC+1
      D=DOLD*EXP(-ZEVT*RR)
      DEVT=ZEVT*RR
      AJ10=SX+SY+SZ
      PO=AJ10/3.
      AJ20=SQRT(((SX-PO)**2+(SY-PO)**2+(SZ-PO)**2)/2.+SXY**2)
C        COMPUTE STRESS INVARIANTS.
      BK=(AK(M)+AK2(M)*AJ10)*(1.+1.5*AK2(M)*DEVT*(1.+AK2(M)*DEVT))
      BK= AMIN1(AKSOL(M),AMAX1(BK,AK(M)))
      AJ1T=AJ10+3.*BK*DEVT
      PAT=AJ1T/3.
      EV=DEVT/3.
      BG=GH(AJ20)
      SXT=(SX-PO)+2.*BG*(EX*RR-EV)
      SYT=(SY-PO)+2.*BG*(EY*RR-EV)
      SZT=(SZ-PO)+2.*BG*(EZ*RR-EV)
      SXYT=SXY+2.*BG*EXY*RR
      AJ2T=SQRT((SXT**2+SYT**2+SZT**2)/2.+SXYT**2)
      IF (MUP2(M) .EQ. 0) GO TO 205
      DEIT=(AJ2T-AJ20)/(2.*BG)
      AJ2T=(MUP(M)/MUP2(M)+AJ20)*EXP(2.*MUP2(M)*DEIT)
     1     -MUP(M)/MUP2(M)
205   BG=GG(AJ2T,AJ20)
      AJ2T=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
1205  FORMAT (* LOC=205 J10,J1T,J20,J2T=*1P4E10.3,* BK,BG=*2E10.3,
     1  /10X,* DEVT,DEIT=*2E10.3,* RR,RSUM=*2E10.3,* D,DOLD=*0P2F10.6)
C
```

```
C        EVALUATE F1 AND F2 FROM ELASTIC STRESSES.
      AMC=AMC1(M)
      IF(AJ1T.LT.-5.*(AMC3(M)+AMC5(M)))GOTO215
      AMC=-AJ1T
      IF(AJ1T.GT.AMC6(M))GOTO215
      AMC=     AMC1(M)+AMC2(M)*EXP(AJ1T/AMC3(M))+AMC4(M)*EXP(AJ1T/
     1 AMC5(M))
  215 F1T=AJ2T-AMC
      CALL CAPPR(PT,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-5,DEVT)
      ZEP=ZEVP+DZEP
      CALL CAPPR(PZ,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,
     1  /10X,* RR,EPRAT1,EPRAT2=*0P3F10.6)
      IF (F1T .LE. 0. .AND. F2T .LE. 0.) GO TO 500
      AJ11=AJ10+EPRAT1*(AJ1T-AJ10)
      AJ21=AJ20+EPRAT2*(AJ2T-AJ20)
C ********                                                 ***********
C        COMPUTATION OF YIELDING PROCESS
      NQ=1
      BK=BKH(AJ11)
      BG=GH(AJ21)
C
C         YIELD ON MOHR-COULOMB SURFACE.
C
      IF(F1T.LT.0.) GO TO 350
      IF(ZEVT.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 .EQ. 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
      XI1=TAU2/AMC3(M)+TAU3/AMC5(M)
      BK=BKK(AJ1,AJ10)
      BG=GG(AJ2,AJ20)
      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))/(XI1+BG/(9.*BK*XIB))
      IF (DBUG1 .EQ. 1) PRINT 1320,NC,AJ1,AJ2,DJ2,DJ,TAU2,TAU3,XI1,BK,BG
1320  FORMAT (* M-C NC=*I2,* J1,J2,DJ2,DJ=*1P4E12.5,* TAU,XI,K,G=*
     1  5E10.3)
      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,AJ1,AJ2,AJ2A,F21,F1T
1330  FORMAT (* M-C END, J1,J2,J2A=*1P3E12.5,* F21,F1T=*2E10.3)
      IF (F21 .GT. 0.) GO TO 410
      IH=6
      GO TO 600
C
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=SQRT(W2(M)*AMAX1(0.,PT**2-AJ10P**2/9.))
      ZEP=ZEVP+AMIN1(0.,DEVT-(AJ10P-AJ10)/BK)
      CALL CAPPR(PZ,ZEP,M,D,IHH,YM)
353   IF (ABS(AJ1T-AJ10P) .GE. 1.) BB=(PZ-PT)/(AJ1T-AJ10P)
```

```
      IH=7
      ILO=1
      IHI=2
C      FIRST ESTIMATE OF J1
      NCAP=1
      INT=ILO
      AJ1=AJ1OP*(1.+BB/(PT+BB*AJ1OP)*(3.*DEVT*BKH(AJ1OP)))
C           COMPUTATION OF J2 AND ERROR DLA
  355 BK=BKK(AJ1,AJ10)
      DZEP=AMIN1(0.,DEVT-(AJ1-AJ10)/(3.*BK))
      CALL CAPPR(PR,ZEVP+DZEP,M,D,IH,YM)
      AJ1=AMAX1(AJ1,3.3*PR)
      PJ=AMIN1(ABS(AJ1),ABS(AJ1-6.*PR))/3.
      AJ2=0.
      IF (PR**2-PJ**2 .GT. 1.)
     1AJ2=SIGN(SQRT(W2(M)*(PR**2-PJ**2)),AJ1-3.*PR)
      BG=GG(ABS(AJ2),AJ20)
      IF (IH .EQ. 9 .AND. NCAP .GE. 3) GO TO 480
      DEIP=DEIT-(AJ2-AJ20P)/(2.*BG)
      DLA=DEIP-1.5*(AJ2+AJ20P)*DZEP/(W2(M)*(AJ1+AJ1OP))
      AJA=AJ1
      IF (DBUG1 .EQ. 1)
     1PRINT 1365,NCAP,AJ1,AJ2,DLA,BG,DEIP,PJ,PR
 1365 FORMAT (* 365 NCAP=*I3,* J1,J2=*1P2E13.6,* DLA,BG,DEIP,PJ,PR=*
     1  5E10.3)
      IF (2.*ABS(DLA)*BG .LT. DFCR1) GO TO 390
      IF (NCAP .GE. 30) GO TO 700
      IF (NCAP-2) 358,360,370
C         SECOND ESTIMATE OF J1
  358 IF (ABS(AJ1) .LE. 1.E4) AJ1=-3.*SQRT(AMAX1(0.,PT**2-AJ2T**2/
     1  W2(M)))
      DJ2=2.*DLA*BG
      IF (ABS(AJ1) .GT. 1.E4) AJ1=-9./W2(M)*ABS(AJ2)/AJ1*DJ2 + AJ1
      GO TO 382
  360 INT=ILO
      IF (DL(ILO)*DLA .LT. 0.) GO TO 375
      INT=IHI
      IF (AMAX1(DL(ILO),DLA) .LT. 0.) GO TO 366
C         MOST TENSILE ESTIMATE OF J1
      AJ1=-3.*SQRT(AMAX1(0.,PT**2-AJ2T**2/W2(M)))
      AJ1=0.5*(AJ1+AMAX1(AJ1T,AJ10))
      GO TO 382
C           MOST COMPRESSIVE ESTIMATE OF J1
  366 AJ1=3.*PT
      IF (ABS(DEVT) .LT. 1.E-7) GO TO 370
      DEVZP=DEVT-(PZ-AJ10/3.)/BKK(AJ10,3.*PZ)
      DEVTP=DEVT-(PT-AJ10/3.)/BKK(AJ10,3.*PT)
      DEVP=DEVTP*DEVT/(DEVT+DEVTP-DEVZP)
      AJ1=3.*(PT+(PZ-PT)*DEVP/DEVT)
      IF (DBUG1 .EQ. 1)
     1PRINT 1357,NCAP,AJ1OP,AJ2OP,AJ10,AJ2,DEVZP,DEVTP,DEVP,AJ1
 1357 FORMAT (* 357 NCAP=*I3,* AJ1OP,AJ2OP,AJ10,AJ2=*1P4E10.3/
     1  * DEVZP,DEVTP,DEVP,AJ1=*4E10.3)
      GO TO 382
  370 INT=IHI
C           REGULA FALSI CALCULATION OF J1
      IF (DL(ILO) .GT. 0. .OR. (DL(IHI) .GT. 0. .AND. DLA .GT. 0.))
     1  INT=ILO
  375 AJ1=AJA-(AJ(INT)-AJA)/(DL(INT)-DLA)*DLA
      IF (MOD(NCAP,5) .EQ. 0) AJ1=(AJ1+AJA+AJ(INT))/3.
C           STORAGE OF RESULTS OF PREVIOUS ITERATIONS
      INT=IHI
      IF (NCAP .EQ. 2) GO TO 382
      IF ((DL(IHI) .GT. 0. .AND. DLA .GT. DL(IHI)) .OR. (DLA .LT.
     1  DL(ILO) .AND. DL(ILO) .LT. 0.)) GO TO 385
      IF (DL(ILO) .GT. 0. .OR. (DL(IHI) .GT. 0. .AND. DLA .GT. 0.))
     1  GO TO 382
      INT=ILO
  382 DL(INT)=DLA
      AJ(INT)=AJA
      IF (NCAP .EQ. 1) GO TO 385
      IF (DL(IHI) .GT. DL(ILO)) GO TO 385
      INT=IHI
      IHI=ILO
      ILO=INT
```

SUBROUTINE CAP1 (Continued)

```
385   CONTINUE
      NCAP=NCAP+1
      IF (DBUG1 .EQ. 1)
     1PRINT 1385,NCAP,IHI,ILO,AJ(1),AJ(2),DL(1),DL(2)
1385  FORMAT (* 385 NCAP=*I3,* HI,LO=*2I2,* AJ,DL=*1P4E10.3)
      GO TO 355
C     CHECK FOR CONVERGENCE TO POINT ABOVE THE M-C CURVE
390   F11=AJ2-(AMC1(M)+AMC2(M)*EXP(AJ1/AMC3(M))+AMC4(M)*EXP(AJ1/
     1  AMC5(M)))
      DJ2=2.*DLA*BG
      IF (ABS(AJ1) .GE. 1.E4) AJ1=-9./W2(M)*ABS(AJ2)/AJ1*DJ2 +AJ1
      AJ2=AMAX1(0.,AJ2+DJ2)
      IF (F11 .LT. DFCR1) GO TO 600
      IF (DBUG1 .EQ. 1)
     1PRINT 1390,NCAP,AJ1,AJ2,F11
1390  FORMAT (* 390 SKIP TO JOINT, NCAP=*I3,* J1,J2,F11=*1P3E10.3)
      GO TO 418
C
C           YIELD AT JOINT OF CAP AND MOHR-COULOMB
C
400   IF (IH .NE. 8) GO TO 350
      CRIT=(AJ2T-AJ20)*W2(M)*BK*AJ10
      CRIT2=(AJ1T-AJ10)*BG*AJ20
      IF (DBUG1 .EQ. 1)
     1PRINT 1405,CRIT,CRIT2,AJ10,AJ20
1405  FORMAT (* 405 CRIT,CRIT2=*1P2E10.3,* AJ10,AJ20=*2E10.3)
      IF (CRIT .GT. CRIT2) GO TO 350
410   DF11=-AMC2(M)/AMC3(M)*EXP(AJ11/AMC3(M))-AMC4(M)/AMC5(M)*EXP(AJ11/
     1  AMC5(M))
      DF21=2./9.*AJ11
      DF22=2.*AJ21/W2(M)
      DET=9.*BK*(DF11*DF22-DF21)-DPDE
      AJ1=-DPDE*(AJ1T-AJ10)/DET + AJ10
      AJ2=DPDE*DF11*(AJ1T-AJ10)/DET + AJ20
      AJ2=AMAX1(0.,AJ2)
      IF (DBUG1 .EQ. 1)
     1PRINT 1908,AJ1,AJ2,AJ10,AJ20,DF11,DF21,DF22,DPDE
 1908 FORMAT(*    JOINT 418      - AJ1,AJ2,AJ10,AJ20 =*,1P4E10.3/
     1 *    DF11,DF21,DF22,DPDE =*,4E10.3)
      DEP=DEVT-(AJ1-AJ10)/(3.*BK)
418   NMC=0
420   NMC=NMC+1
      IH=8
      TAU2=TAU3=0.
      IF (AJ1  .LT.  -10.*AMAX1(AMC3(M),AMC5(M))) GO TO 430
      AJJ=AJ1
      IF (AJ1  .GT.  AMC6(M)) AJJ=AMC6(M)
      TAU2=AMC2(M)*EXP(AJJ/AMC3(M))
      TAU3=AMC4(M)*EXP(AJJ/AMC5(M))
430   AJ2 =AMC1(M)+TAU2+TAU3
      AJ2=AMAX1(0.,AJ2)
      XI1=TAU2/AMC3(M)+TAU3/AMC5(M)
      XI2=TAU2/AMC3(M)**2+TAU3/AMC5(M)**2
      DZEP=DEVT-(AJ1-AJ10)/(3.*BK)
      ZEP=ZEVP+AMIN1(0.,DZEP)
      CALL CAPPR(PR,ZEP,M,D,IH,YM)
      IF (IH .EQ. 9) GO TO 480
      YM=YM/(3.*BK)
      IF (ABS(SQRT(AJ1**2/9.+AJ2**2/W2(M))+PR)  .LT.  DFCR1) GO TO 600
475   DJ=0.5*(AJ2 **2/W2(M)+AJ1**2/9.-PR**2)/(-AJ2 *XI1/W2(M)-AJ1/9.+
     1  PR*YM)
      DJA=DJ
      AAA=(XI2*DJ)**2/W2(M)+2.*XI1*XI2*DJ/W2(M)+XI1**2/W2(M)+2.*AJ2*XI2
     1  /W2(M)+0.11111-YM**2
      BBB=2.*(AJ2 *XI1/W2(M)+AJ1/9.-PR*YM)
      CCC=AJ2 **2/W2(M)+AJ1**2/9.-PR**2
      IF (BBB**2-4.*AAA*CCC .GT. 0.)
     1DJ=0.5*BBB/AAA*(SQRT(1.-4.*AAA*CCC/BBB**2)-1.)
      AJ1=AMIN1(AJ10,AJ1+DJ)
      CRIT=SQRT(AJ2**2/W2(M)+AJ1**2/9.)
      IF (DBUG1 .EQ. 1)
     1PRINT 1480,AJ1,DJ,DJA,AJ2,CRIT,PR,AAA,BBB,CCC
1480  FORMAT (* 480 AJ1,DJ,DJA=*1P3E10.3,* AJ2,CRIT,PR=*3E10.3,
     1  * AAA,BBB,CCC=*3E10.3)
      IF(NMC.GE. 5) GO TO 700
```

```
      IF(NMC.GT.1) GO TO 420
      BK=BKK(AJ1,AJ10)
      BG=GG(AJ2,AJ20)
      GO TO 420
480   AJ1=3.*PR
      AJ2=AMIN1(AJ2,AMC1(M))
      GO TO 600
C ********                                                   ***********
C        COMPLETION OF STRESS CALCULATION FOR ELASTIC CASE
  500 AJ1=AJ1T
      P=AJ1/3.
      SX=SX-P0+2.*BG*(EX*RR-DEVT/3.)+P
      SY=SY-P0+2.*BG*(EY*RR-DEVT/3.)+P
      SZ=AJ1-SX-SY
      SXY=SXY+2.*BG*EXY*RR
      AJ2=AJ2T
      IH=5
      GO TO 630
C     **********
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.
      SXD=SX-P0+2.*BG*(EX-ZEVT/3.)*RR
      SYD=SY-P0+2.*BG*(EY-ZEVT/3.)*RR
      SZD=SZ-P0+2.*BG*(EZ-ZEVT/3.)*RR
      SXYD=SXY+2.*BG*EXY*RR
      AJ2T=SQRT(0.5*(SXD**2+SYD**2+SZD**2)+SXYD**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. DAMG(M) .AND. SMAX .GE. SCRIT(M) .AND. AJ1 .GE. 0.)
     1 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.*BG/BK)
      EMOD=2.*(1.+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=I3,4H, J=I3,9H ON CYCLEI4)
      IH=10
      GO TO 630
C ********                                                   ***********
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.
      SXD=SX-P0+2.*BG*(EX-EV)*RR
```

SUBROUTINE CAP1 (Concluded)

```
      SYD=SY-P0+2.*BG*(EY-EV)*RR
      SZD=SZ-P0+2.*BG*(EZ-EV)*RR
      SXYD=SXY+2.*BG*EXY*RR
      AJ2T=SQRT(0.5*(SXD**2+SYD**2+SZD**2)+SXYD**2)
      FAC=AJ2/AMAX1(1.,AJ2T)
      SX=SXD*FAC+P
      SY=SYD*FAC+P
      SZ=SZD*FAC+P
      SXY=SXYD*FAC
C ********                                                    ***********
C         PREPARE FOR NEXT SUBCYCLE
630   CONTINUE
      RSUM=RSUM+RR
      IF (DBUG1.EQ.1) PRINT 1630,K,J,RR,RSUM,SX,SY,SZ,SXY,AJ1,AJ2,D,
     1  DOLD,IH,ZEVT,ZEIT,ZEVP,TEVP
1630  FORMAT (* --FINAL--K,J=*2I3,* RR,RSUM=*2F8.5,* SX,SY,SZ,SXY=*
     1  1P4E10.3,/4X,* AJ1,AJ2=*2E13.6,* D,DOLD=*0P2F10.6,* IH=*I2,
     2  * ZEVT,ZEIT,ZEVP,TEVP=*1P4E10.3)
      IF(1.-RSUM.LT.1.E-10) RETURN
      IF (ABS(AJ1T-AJ10) .GT. 1.) EPRAT1=AMAX1(-1.,AMIN1(1.,
     1  (AJ1-AJ10)/(AJ1T-AJ10)))
      IF (ABS(AJ2T-AJ20) .GT. 1.) EPRAT2=AMAX1(-1.,AMIN1(1.,
     1  (AJ2-AJ20)/(AJ2T-AJ20)))
      RR=AMIN1(1.-RSUM,1.3*RR)
      DOLD=D
      IF (NINC .LT. NMAX) GO TO 200
      PRINT 1630,K,J,RR,RSUM,SX,SY,SZ,SXY,AJ1,AJ2,D,DOLD,IH,ZEVT,
     1  ZEIT,ZEVP,TEVP
      PRINT 1650,NINC
1650  FORMAT (* STOP CALLED FOR NMAX=NINC=*I4)
      STOP 3121
C ********                                                       ********
C              CUT STRAIN INCREMENT AND RESTART
C
  700 RR=0.5*RR
      NRE=NRE+1
      IF(NRE.GE.1) PRINT 1700,NRE,N,K,J,DH,DOLD,DORG,RR,SX,SY,SZ,SXY,
     2 EX,EY,EZ,EXY,ZEVP,TEVP,IH,AJ1,AJ2,F11,F21,AJ10,AJ20,F10,F20,
     2 F1T,F2T,DPDE
      EPRAT1=EPRAT2=0.1
      IF(NRE.GE.10) STOP 3120
      GO TO 200
C ********                                                    ***********
1020  FORMAT (4(2A5,E10.3))
1021  FORMAT (1H+,80X,3HIN=I2,4H CAP)
1022  FORMAT (2A5,7E10.3)
1024  FORMAT (2A5,I10,2A5,I10,2A5,E10.3,2A5,E10.3)
1040  FORMAT (4(2A5,1PE10.3))
1042  FORMAT (2A5,1P7E10.3)
1044  FORMAT (2A5,I10,2A5,I10,2A5,1PE10.3,2A5,E10.3)
1700  FORMAT (* RESTART WITH NRE,N,K,J=* I2,I5,2I3,* DH,DOLD,DORG,RR=*
     1  4F15.10/*  SX,SY,SZ,SXY=*1P4E10.3,* EX,EY,EZ,EXY=*4E10.3/
     2  *ZEVP,TEVP,IH=*2E10.3,I5,* AJ1,AJ2=*2E10.3,* F11,F21=*2E10.3
     3  /* AJ10,AJ20,F10,F20=*4E10.3,* F1T,F2T,DPDE=*3E10.3)
      END
```

SUBROUTINE CAPPR

```
      SUBROUTINE CAPPR(P,EP,M,D,IH,YM)
C
      COMMON /EQS/EQSTC(6),EQSTD(6),EQSTE(6),EQSTG(6),EQSTH(6),EQSTN(6),
     1  EQSTS(6),RHO(6),RHOS(6),YC(6),YAD(6),MU(6),ESC(6,20),CLIN,CQSQ,
     2  TRIQ,AMAT(6,4),SP(6),G2(6),PMIN(6)
      COMMON /POR/ PORA(6,5),PORB(6,5),PORC(6,5),EVP(6,5)
C
      P=PORA(M,1)
      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=(PORA(M,NC)+(PORB(M,NC)+PORC(M,NC)*EP)*EP)
      YM=-(PORB(M,NC)+2.*EP*PORC(M,NC))
      IF (D .LT. RHOS(M)) GO TO 145
      EMU=D/RHOS(M)-1.
      PS=EMU*(EQSTC(M)+EMU*(EQSTD(M)+EMU*EQSTS(M)))
      IF (PS .LT. -P) GO TO 145
      YM=-D*EQSTC(M)/RHOS(M)
      P=-PS
      IH=9
145   RETURN
      END
```

SUBROUTINE DEPOS

```
      SUBROUTINE DEPOS(NPART,IN)
C
C     THIS ROUTINE USED WITH SRI GENRAT.
C        CALLED BY GENRAT FOR RADIATION DEPOSITION CALCULATIONS.
C        ROUTINE IS SEPARATED INTO 3 PARTS BY INDICATOR, NPART, TO
C          1 READ DATA ON MATERIAL ABSORPTION PROPERTIES
C          2 READ SPECTRUM AND DEPOSIT ENERGY INTO  SS  ARRAY.
C          3 PRINT OUT COORDINATE ARRAYS IN DEPOSITION EDIT
C     INPUT -
C      *   TWO   FORMAL PARAMETERS
C      *   READS ABSORPTION SPECTRA, RADIATED SPECTRA FROM CARDS.
C     OUTPUT -
C      *   FILLS SS ARRAYS.
C      *   SETS *SSTOPM*, *JSTAR*, *NSPEC*.
C      *   WRITES DEPOSITION EDIT.
C     THIS IS A VERSION MODIFIED TO ACCEPT THE ABSORPTION COEFFICIENT
C     DATA DIRECTLY FROM FSCATT.
C
      INTEGER H,POROUS,PRESS,RINTER,SOLID,SPALL
      REAL MATL,NEM,NET,NEMH,NETH
C                MISCELLANEOUS
      COMMON AZERO(1),CEF,CKS,DAVG,DELTIM,DISCPT(10),DOLD,DRHO,DTMAX,
     1  DTMIN,DTN,DTNH,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),NSEPRAT,NSPALL,NTEDT,
     4  NTEX,NTR(15),POLD,P6(20),R(30),RLAST,SLAST,SMAX,TEDIT(50),
     5  TF,TIME,TJ,TREZON,TS,T6(20),ULAST,UOLD,UZERO,XLAST,XNOW,XOLD
     1  ,XJDIT(20),MS
C                HALFSTEP VALUES
      COMMON DH,DHLAST,DUH,EH,PH,RH,RHLAST,SH,SHLAST,UH,UHLAST,XH,XHLAST
     1  ,NEMH,NETH
C                CONDITION INDICATORS
      COMMON INF,LINTER,MIRROR,NORMAL,POROUS,PRESS,RINTER,SOLID,SPALL
C                CELL LAYOUT
      COMMON DXX(30),JBND(30),JMAT(30),NAUTO,MATL(6,2),NLAYER,NMTRLS,
     1  THK(30)
C
C                COORDINATE ARRAYS
      COMMON/COORD/X(200),XD(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/NSC/A(5000)
C                NAMED COMMON
      REAL MU,MUM
      COMMON /EQS/  EQSTA(6),EQSTC(6),EQSTD(6),EQSTE(6),EQSTG(6),
     1  EQSTH(6),EQSTN(6),EQSTS(6),EQSTV(6),CZQ(6),CWQ(6),C2(6)
      COMMON /MELT/ EMELT(6,8),GMELT(6,8),SPH(6),THERM(6,8)
      COMMON /RHO/ RHO(6),RHOS(6)
      COMMON /TSR/ TSR(6,30),EXMAT(6,20),TENS(6,3)
      COMMON /Y/ YD(6),YADD(6),MU(6),MUM,YADDM
      COMMON /IND/ IEOS(6),INDK(20),NALPHA,NCMP(6),NFR(6),NPOR(6),
     1  NDS(6),NPR(6),NCON(6),NVAR(6)
      COMMON /RAD/ SSTOP(9),START(9),SDURM,SSTOPM,NSPEC,SSJ,JSS,IPLOT(4)
     1  ,XMAX(4),XMIN(4),YMAX(4),YMIN(4),IA(7),ITITLE(24),NARZ,TARZ
      COMMON/SS/SS(500)
      COMMON /PES/ LVMAX,LVTOT,LVAR(200),COM(4000)
C
      DIMENSION AC(109),AAD(6,6,10),AA1(6,6,10),AA2(6,6,10),AA3(6,6,10),
     1  EDGE(6,6,10),EI(109),RHOC(6,6),TBL(109),NOE(6,6),IVAR(8),
     2  ATWT(6,6),BBDY(100),PBW(6),NAME(6),EIBB(100)
      DIMENSION DELX(200),EPGJ(200),PCT(200),CPG(200),TC(200),P(200),
     1  DIMPMCC(200),FRONT(5,30),XPL(200),YPL(200),EABS(200)
      DIMENSION ANGLE(30)
C
      EQUIVALENCE(A(201),AC),(A(310),AAD),(A(670),AA1),(A(1030),AA2),
     1  (A(1390),AA3),(A(1750),EDGE),(A(2110),EI),(A(2219),RHOC),
     2  (A(2255),TBL),(A(2364),NOE),(A(2400),ATWT)
      EQUIVALENCE(DELX,A),(EPGJ,A(201)),(PCT,A(401)),(CPG,A(601)),
     1  (TC,A(801)),(P,A(1001)),(DIMPMCC,A(1201)),(XPL,A(1401)),
     2  (YPL,A(1601)),(EABS,A(1801))
C
      DATA BBDY/.01,.03,.05,.07,.09,.15,.25,.35,.45,.55,.65,.75,
     1.85,.95,1.05,1.15,1.25,1.35,1.45,1.55,1.65,1.75,1.85,1.95,2.05,
     22.15,2.25,2.35,2.45,2.55,2.65,2.75,2.85,2.95,3.05,3.15,3.25,
     33.35,3.45,3.55,3.65,3.75,3.85,3.95,4.05,4.15,4.25,4.35,4.45,
```

```
     44.55,4.65,4.75,4.85,4.95,5.05,5.15,5.25,5.35,5.45,5.55,5.65,
     55.75,5.85,5.95,6.10,6.3,6.5,6.7,6.9,7.1,7.3,7.5,7.7,7.9,8.1,
     68.3,8.5,8.7,8.9,9.1,9.3,9.5,9.7,9.9,10.5,11.5,12.5,13.5,14.5,15.5,
     716.5,17.5,18.5,19.5,20.5,5*0.0/
      DATA EIBB/4.076E-7,2.829E-6,7.604E-6,1.466E-5,2.393E-5,
     13.312E-4,8.555E-4,1.582E-3,2.475E-3,3.498E-3,4.622E-3,5.818E-3,
     27.057E-3,8.330E-3,9.595E-3,1.085E-2,1.208E-2,1.326E-2,1.438E-2,
     31.544E-2,1.644E-2,1.736E-2,1.819E-2,1.893E-2,1.960E-2,2.017E-2,
     42.067E-2,2.106E-2,2.138E-2,2.163E-2,2.178E-2,2.187E-2,2.188E-2,
     52.183E-2,2.172E-2,2.155E-2,2.132E-2,2.105E-2,2.073E-2,2.037E-2,
     61.998E-2,1.956E-2,1.910E-2,1.863E-2,1.814E-2,1.763E-2,1.711E-2,
     71.657E-2,1.603E-2,1.549E-2,1.495E-2,1.440E-2,1.387E-2,1.332E-2,
     81.279E-2,1.227E-2,1.176E-2,1.125E-2,1.076E-2,1.027E-2,9.800E-3,
     99.350E-3,8.910E-3,8.470E-3,1.572E-2,1.417E-2,1.274E-2,1.142E-2,
     .1.021E-2,9.110E-3,8.100E-3,7.200E-3,6.370E-3,5.640E-3,4.970E-3,
     .4.380E-3,3.850E-3,3.380E-3,2.960E-3,2.510E-3,2.350E-3,1.980E-3,
     .1.730E-3,1.500E-3,5.008E-3,2.425E-3,1.147E-3,5.322E-4,2.429E-4,
     .1.092E-4,4.852E-5,2.131E-5,9.269E-6,3.996E-6,2.960E-6,5*0./
C
2     FORMAT(1H+,79X,5H IND=A2,5H, IN=I2,9H -DEPOS- )
3     FORMAT(1H+,103X,* ANGLE FROM NORMAL(DEG)*)
4     FORMAT(1H+,103X,*, , ,CAL/CM2,SEC,SEC*)
5     FORMAT(1H+,103X,*,CM,ERG/G*)
 10   FORMAT(A1,A9,A5,A2,I3,3(A10,E10.3))
 11   FORMAT(4(A10,1PE10.3))
14    FORMAT (*1*,10A10//3X,*J*,9X,*X*,9X,*X*,2(5X,*DEPOS*),
     1 3X,*PCT TR.*,5X,*TEMP.*,4X,*PRESS.*,3X,*IMPULSE*,5X,*MATERIAL*4X,
     2*COND*6X,*J*,8X,*DX*2X,*ABSORBED*/10X,*INCH*8X,*CM*3X,*ERGS/GM*,4X
     3 ,*CAL/GM*,14X,*DEG. C*,6X,*KBAR*,5X,*KTAPS*,36X,*CM*3X,*CAL/CM2*)
 15   FORMAT(I4,2F10.6,1P2E10.3,F10.3,1P3E10.3,3X,A9,3X,3R2,2X,I5,1PE10.
     13,F10.3)
 54   FORMAT(A10,I10,A7,A3,1P5E10.3)
 55   FORMAT(A1,A9,A8,I2,A10,F10.7)
 56   FORMAT(A1,A9,2A10,I10,A10,F10.3)
 57   FORMAT(1P8E10.3)
 70   FORMAT(*0*8X,*YIELD=*1PE10.3,* SOUND SP=*1PE10.3,* DENSITY=*
     1  1PE10.3,* TENS STR=*1PE10.3,* INTERFACE STRENGTH=*1PE10.3/)
 72   FORMAT(10X,*DEPOS - CONST. DENSITIES (G/CM3), RHOC =*1P6E10.3)
 73   FORMAT(10X,*DEPOS - ESUM =*1PE10.3,* CAL/CM2*)
 74   FORMAT(*    TOTAL ENERGY ABSORBED IS*1PE12.3,* CAL/CM2*)
75    FORMAT (A1,A9,2A10,I10,4A10)
 76   FORMAT(* TOTAL ENERGY ABSORBED IS*1PE12.3,* CAL/CM*)
 77   FORMAT(* TOTAL ENERGY ABSORBED IS*1PE12.3,* CAL*)
 80   FORMAT(A7,1P4E13.5)
 81   FORMAT(7X,1P4E13.5)
82    FORMAT (* ERROR IN MCCLOSKEY INTEGRAL FOR LAYER*I3,*, X(J+1)=*
     1  1PE10.3,* DID NOT LIE BETWEEN *1PE10.3,* AND *1PE10.3)
 89   FORMAT(13X,E12.5,1X,E12.5,1X,E12.5,1X,E12.5)
 92   FORMAT(4(2A5,1PE10.3))
 93   FORMAT(A1,A9,2A10,1PE10.3,A10,1PE10.3)
C
240   GO TO (250,400,700) NPART
C *************                                                ***********
C                ENTRY FOR READING MATERIAL ABSORPTION PARAMETERS
250   CONTINUE
      NCONST=NCON(M)
      NNOE=10H X-RAY ABS
      IDD=1H    $   IN5=5
      DO 260 NC=1,NCONST
      READ (IN,55)A1,NAME(NC),A2,ITAPE,A3,PBW(NC)
      WRITE (6,55)A1,NAME(NC),A2,ITAPE,A3,PBW(NC)
      WRITE (6,2) IDD,IN
      INL=IN
      IF (ITAPE .EQ. 0) GO TO 255
      INL=ITAPE
      CALL REDR(NAME(NC),NNOE,INL,2)
255   READ (INL,56)A1,NAME(NC),A2,A3,NOE(M,NC),A4,ATWT(M,NC)
      WRITE (6,56)A1,NAME(NC),A2,A3,NOE(M,NC),A4,ATWT(M,NC)
      WRITE (6,2) IDD,INL
      NOED=NOE(M,NC)  $  NOE1=NOED+1
      READ  (INL,89)   (EDGE(M,NC,ND),ND=1,NOED)
      FN=7H EDGE
      WRITE ( 6,80)FN,(EDGE(M,NC,ND),ND=1,NOED)
      WRITE (6,2) IDD,INL
      READ  (INL,89)   (AA0(M,NC,ND),AA1(M,NC,ND),AA2(M,NC,ND),
```

SUBROUTINE DEPOS (Continued)

```
     1  AA3(M,NC,ND),ND=1,NOED)
      FN=7H COEFS
      WRITE(6,80) FN,AA0(M,NC,1),AA1(M,NC,1),AA2(M,NC,1),AA3(M,NC,1)
      IF (NOED .EQ. 1) GO TO 258
      WRITE(6,81)(AA0(M,NC,ND),AA1(M,NC,ND),AA2(M,NC,ND),AA3(M,NC,ND),
     1  ND=2,NOED)
258   WRITE(6,2)IDD,INL
260   CONTINUE
      WTOT=1.
      IF (NCONST .EQ. 1 .OR. PBW(1) .LT. 1.) GO TO 280
      WTOT=0.
      DO 270 NC=1,NCONST
      PBW(NC)=PBW(NC)*ATWT(M,NC)
270   WTOT=WTOT+PBW(NC)
280   DO 290 NC=1,NCONST
290   RHOC(M,NC)=RHO(M)*PBW(NC)/WTOT
      IF (NCONST .EQ. 1) RHOC(M,1)=RHO(M)
      WRITE (6,72) (RHOC(M,NC),NC=1,NCONST)
      RETURN
C ************                                              ***********
C                 ENTRY FOR DEPOSITING RADIATION IN THE  SS  ARRAY
400   READ  (5,54) A1,NSPEC,N2,A2,(ANGLE(NL),NL=1,5)
      WRITE (6,54) A1,NSPEC,N2,A2,(ANGLE(NL),NL=1,5)
      WRITE (6,2) IND,IN5  $   WRITE (6,3)
      IF (N2 .EQ. 7H ANGLES ) GO TO 402
      DO 401 NL=2,NLAYER
401   ANGLE(NL)=ANGLE(1)
      GO TO 403
402   IF (NLAYER .LE. 5) GO TO 403
      READ  (5,57) (ANGLE(NL),NL=6,NLAYER)
      WRITE (6,57) (ANGLE(NL),NL=6,NLAYER)
      WRITE (6,2) IND,IN5   $   WRITE (6,3)
403   DO 404 NL=1,NLAYER
      FRONT(1,NL)=FRONT(2,NL)=FRONT(3,NL)=FRONT(4,NL)=FRONT(5,NL)=0.
404   ANGLE(NL)=COS(ANGLE(NL)/57.2957795)
C                    BEGIN LOOP OVER EACH SPECTRUM
      TOTCAL=0.
      DO 4040 I=1,500
4040  SS(I)=0.
      DO 485 NS=1,NSPEC
      JFINNS=JFIN*(NS-1)
      IN=5
      IDD=5H
C        INDICATOR IN COLUMNS 11 THROUGH 15 SHOWS SPECTRUM TYPE
C              5H NHNU  = ARBITRARY SPECTRUM
C              5H NBB   = SERIES OF BLACK BODIES (NBB OF THEM)
C              5H NARB  = DEPOSITION FROM SCATT PROGRAM
      READ (5,10) IND,SPECNAM,A1,A2,NHNU,A3,ECAL,A4,START(NS),A5,
     1  SSTOP(NS)
      WRITE (6,10)IDD,SPECNAM,A1,A2,NHNU,A3,ECAL,A4,START(NS),A5,
     1  SSTOP(NS)
      WRITE (6,2) IND,IN5   $   WRITE (6,4)
      NARB=NBB=NHNU
      IF (IND .EQ. IDD) GO TO 405
      IN=4
      CALL REDR(SPECNAM,IDD,IN,1)
405   CONTINUE
      SSTOPM=AMAX1(SSTOPM,SSTOP(NS) )
      IF (A1 .EQ. 5H NARB) GO TO 465
      IF (A1 .EQ. 5H NBB ) GO TO 420
      NRAD=1  $  TEMP=1.
C               ARBITRARY SPECTRUM INPUT
      READ (IN,75)A1,SPECNAM,A2,A3,NHNU,(IVAR(I),I=1,4)
      WRITE (6,75)A1,SPECNAM,A2,A3,NHNU,(IVAR(I),I=1,4)
      WRITE (6,2) IDD,IN
      IF (IVAR(1) .NE. IDD) GO TO 412
      DO 410 NH=1,NHNU
      READ (IN,11) A1,TBL(NH),A2,EI(NH),A3
410   WRITE (6,11) A1,TBL(NH),A2,EI(NH),A3
      WRITE (6,2) IDD,IN
      GO TO 415
412   READ (IN,IVAR)(TBL(NH),EI(NH),NH=1,NHNU)
      WRITE (6,IVAR)(TBL(NH),EI(NH),NH=1,NHNU)
      WRITE (6,2) IDD,IN
      ESUM = 0.
```

```
      DO 413 NH = 1,NHNU
413   ESUM = ESUM+EI(NH)
415   DO 417 NH=1,NHNU
417   EI(NH) = EI(NH)*ECAL/ESUM
      NR=1  $  GO TO 430
C                 BLACK BODY INPUT
420   NRAD=NBB  $     NHNU=95
      NR=1
424   READ (IN,93) A1,SPECNAM,A2,A3,TEMP,A4,ECAL
      WRITE (6,93) A1,SPECNAM,A2,A3,TEMP,A4,ECAL
      WRITE (6,2) IDD,IN
      DO 428 NH=1,NHNU
      TBL(NH)=BBDY(NH)
428   EI(NH)=ECAL*EIBB(NH)
430   ESUM=0.
      DO 431 NH=1,NHNU
431   ESUM=ESUM+EI(NH)
      WRITE (6,73) ESUM
C
C                 COMPUTATION OF ABSORPTION COEFFICIENT - AC
      PERCNT=0.005*ECAL
      X(1)=0.
      XBNDM=0.
      DX2=50.
      JBEG=1
      DO 460 L=1,NLAYER
      M=JMAT(L)
      DO 432 I=1,4
432   FRONT(I,L)=0.
      DO 433 NH=1,109
433   AC(NH)=0.
      NCONST=NCON(M)
      DO 445 NC=1,NCONST
      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,NC)*EXP(AA0(M,NC,1)+ALNE*AA1(M,NC,1))
     1  *(C.602252/ATWT(M,NC))/ANGLE(L)
      GO TO 444
438   IF (NEDG .GE. NOE(M,NC)) GO TO 440
      IF (EDGE(M,NC,NEDG+1) .GT. TBL(NH)*TEMP) GO TO 440
      NEDG=NEDG+1  $  GO TO 438
440   AC(NH)=AC(NH)+RHOC(M,NC)*EXP(AA0(M,NC,NEDG)+ALNE*(AA1(M,NC,NEDG)
     1  +ALNE*(AA2(M,NC,NEDG)+ALNE*AA3(M,NC,NEDG))))*(0.602252/ATWT(M,NC
     2  ))/ANGLE(L)
444   CONTINUE
445   CONTINUE
C
C                 DISTRIBUTE ENERGY INTO CELLS
      XBNDM=XBNDM+THK(L)
      JBNDM=JBND(L)-1
      J=JBEG
446   IF (J .GT. JBEG+1 .AND. XBNDM .EQ. 0.) GO TO 447
      DEP=0.
      DO 4461 NH=1,NHNU
4461  DEP=DEP+AC(NH)*EI(NH)
      IF (J .EQ. JBEG)  GO TO 4462
      FRONT(4,L)=DEP/RHO(M)+FRONT(4,L)
      GO TO 447
4462  FRONT(1,L)=DEP/RHO(M)+FRONT(1,L)
      FRONT(3,L)=FRONT(1,L)*RHO(M)*EQSTG(M)*4.186E-2
      IF (SPH(M) .GT. 0.) FRONT(2,L)=FRONT(1,L)/SPH(M)+22.2
      DX1=DX2
447   IF (XBNDM .GT. 0.) GO TO 4481
      DX=X(J+1)-X(J)
      GO TO 449
4481  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.*DX
      DX=XBNDM-X(J)
      X(J+2) = XBNDM
      JBND(L)=J+1
      JBNDM=J
```

SUBROUTINE DEPOS (Continued)

```
448   X(J+1)=X(J)+DX                                                    DEPOS   272
      DX1=DX                                                            DEPOS   273
449   ESUM=0.                                                           DEPOS   274
      DO 450 NH=1,NHNU                                                  DEPOS   275
      IF (EI(NH) .LT. 1.E-20) GO TO 450                                 DEPOS   276
      EIZ=EI(NH)*(1.-EXP(-1.*AC(NH)*DX))                                DEPOS   277
      EI(NH)=EI(NH)-EIZ                                                 DEPOS   278
      ESUM=EIZ+ESUM                                                     DEPOS   279
450   CONTINUE                                                          DEPOS   280
      SS(JFINNS+J)=ESUM*4.186E7/RHO(M)/DX/(SSTOP(NS)-START(NS))*ANGLE(L) DEPOS   281
     1  +SS(JFINNS+J)                                                   DEPOS   282
      TOTCAL=ESUM+TOTCAL                                                DEPOS   283
      IF (J .EQ. JBNDM) GO TO 460                                       DEPOS   284
      J=J+1                                                             DEPOS   285
      GO TO 446                                                         DEPOS   286
460   JBEG=JBND(L)+1                                                    DEPOS   287
      IF (JFIN .GT. 0) GO TO 462                                        DEPOS   288
      JFIN=JBEG                                                         DEPOS   289
      X(JFIN)=X(J+1)                                                    DEPOS   290
462   JINIT=1                                                           DEPOS   291
      NR=NR+1                                                           DEPOS   292
      IF (NR-NRAD) 424,424,485                                          DEPOS   293
C              DEPOSITION FROM SCATT PROGRAM                            DEPOS   294
465   ETOT=0.                                                           DEPOS   295
      DO 483 L=1,NLAYER                                                 DEPOS   296
      M=JMAT(L)                                                         DEPOS   297
      RATIO = 1.                                                        DEPOS   298
      IF (NARB .GE. 0) GO TO 466                                        DEPOS   299
      READ (5,11) A1,RHOOLD                                             DEPOS   300
      WRITE (6,11) A1,RHOOLD                                            DEPOS   301
      RATIO = RHOOLD/RHO(M)                                             DEPOS   302
466   CONTINUE                                                          DEPOS   303
      READ (IN,75) A1,SPECNAM,A2,A3,NPOINT,      (IVAR(I),I=1,4)        DEPOS   304
      WRITE (6,75) A1,SPECNAM,A2,A3,NPOINT,      (IVAR(I),I=1,4)        DEPOS   305
      WRITE (6,2) IDD,IN                                                DEPOS   306
      IF (NPOINT .EQ. 0) GO TO 483                                      DEPOS   307
      IF (IVAR(1) .NE. IDD) GO TO 475                                   DEPOS   308
      DO 470 NP=1,NPOINT                                                DEPOS   309
      READ (IN,92) A1,A2,TBL(NP),A3,A4,EI(NP)                           DEPOS   310
470   WRITE (6,92) A1,A2,TBL(NP),A3,A4,EI(NP)                           DEPOS   311
      WRITE (6,2) IDD,IN   $   WRITE (6,5)                              DEPOS   312
      GO TO 476                                                         DEPOS   313
475   READ (IN,IVAR)(TBL(NP),EI(NP),NP=1,NPOINT)                        DEPOS   314
      WRITE (6,IVAR)(TBL(NP),EI(NP),NP=1,NPOINT)                        DEPOS   315
      WRITE (6,2) IDD,IN   $   WRITE (6,5)                              DEPOS   316
476   CONTINUE                                                          DEPOS   317
      FRONT(1,L) = EI(1) * ECAL + FRONT(1,L)                            DEPOS   318
      IF (SPH(M) .GT. 0.) FRONT(2,L)=FRONT(1,L)/SPH(M)+22.2             DEPOS   319
      FRONT(3,L)=FRONT(1,L)*RHO(M)*EQSTG(M)*4.186E-2                    DEPOS   320
      J=1                                                               DEPOS   321
      IF (L .GT. 1) J=JBND(L-1)+1                                       DEPOS   322
      IF (ABS(TBL(1)-X(J)) .LT. 1.E-10 .AND. RATIO .EQ. 1.) GO TO 478   DEPOS   323
      DX = X(J)-TBL(1)                                                  DEPOS   324
      DO 477 I = 1, NPOINT                                              DEPOS   325
477   TBL(I)=(TBL(I)+DX-X(J))*RATIO+X(J)                                DEPOS   326
478   CONTINUE                                                          DEPOS   327
      XJP1=X(J+1)                                                       DEPOS   328
      DO 479 I=1,NPOINT                                                 DEPOS   329
      IF (TBL(I) .GT. XJP1-1.E-8) GO TO 480                             9/12/79   1
479   CONTINUE                                                          DEPOS   331
      PRINT 82,L,XJP1,TBL(1),TBL(NPOINT)                                DEPOS   332
      GO TO 481                                                         DEPOS   333
480   I=MIN0(I,NPOINT-1)                                                DEPOS   334
      X1=TBL(I-1)  $   X2=TBL(I)   $   X3=TBL(I+1)                      DEPOS   335
      Z1=(XJP1-X3)/(X2-X1)*(XJP1-X2)/(X3-X1)                            DEPOS   336
      Z2=(XJP1-X1)/(X3-X2)*(XJP1-X3)/(X1-X2)                            DEPOS   337
      Z3=(XJP1-X2)/(X3-X1)*(XJP1-X1)/(X3-X2)                            DEPOS   338
      FRONT(4,L)=ECAL*EI(1)**Z1*EI(2)**Z2*EI(3)**Z3 + FRONT(4,L)        DEPOS   339
481   CONTINUE                                                          DEPOS   340
      CALL SCATTO(TBL,EI,ECAL,NPOINT,NS,L,ESUM)                         DEPOS   341
      ETOT=ESUM*RHO(M)+ETOT                                             DEPOS   342
483   CONTINUE                                                          DEPOS   343
      RATIO = ECAL                                                      DEPOS   344
      IF (IABS(NARB) .EQ. 1)  RATIO = ECAL/ETOT                         DEPOS   345
      DO 484 J=1,JFIN                                                   DEPOS   346
```

```
  484 SS(JFINNS+J)=SS(JFINNS+J)*RATIO
      TOTCAL=TOTCAL+RATIO*ETOT
C               END OF NSPEC LOOP
 485  CONTINUE
 500  RETURN
C *************                                              ***********
C               ENTRY FOR PRINTING DEPOSITION EDIT
 700  WRITE (6,14)(DISCPT(I),I=1,10)
      JBEG=1
      SUMCAL=0.
      DO 708 L=1,NLAYER
C**FIND IMPULSE IN EACH LAYER
      ZLAGR=0.
      M=JMAT(L)
      JBNDM=JBND(L)
      EQE=EQSTE(M)  $  EQM=EMELT(M,1)
      DZLAST=0.
      DO 707 J=JBEG,JBNDM
      DELX(J)=X(J+1)-X(J)
      DZ=ZHL(J)   $   ZLAGR=(DZLAST+DZ)/2.+ZLAGR
          EPG=0.
      IF (J .LT. JBNDM) GO TO 701
      XRAT=(X(J)-X(J-1))/(X(J)-X(J-2))
      CPG(J)=CPG(J-1)+(CPG(J-1)-CPG(J-2))*XRAT
      PCT(J)=PCT(J-1)
       TC(J)= TC(J-1)+( TC(J-1)- TC(J-2))*XRAT
        P(J)=  P(J-1)+(  P(J-1)-  P(J-2))*XRAT
      EABS(J)=EABS(J-1)+(EABS(J-1)-EABS(J-2))*XRAT
      EPGJ(J)=0.  $   DIMPMCC(J)=DIMPMCC(J-1)   $   GO TO 707
 701  CONTINUE
          DO 702 NS=1,NSPEC
      JF=JFIN*(NS-1)+J
  702 EPG=SS(JF)*(SSTOP(NS)-START(NS))+EPG
C               TEST FOR SETTING JSTAR
          IF (EPG*EQSTG(M) .GT. 1.E7) JSTAR=J
C               STORE ENERGY (ERGS/GM), CALORIES AND SUM OF CALORIES IN -
      EPGJ(J)=EPG
      CPG(J)=EPG/4.186E7
      SUMCAL=SUMCAL+CPG(J)*ZHL(J)/ANGLE(L)
      PCT(J)=100.*(1.-SUMCAL/TOTCAL)
      EABS(J)=SUMCAL
      TC(J)=0. $   IF(SPH(M) .GT. 0.) TC(J)=CPG(J)/SPH(M)+22.2
      IH1=H(J,1)  $  IH3=H(J,3)
      DH=DOLD=DHL(J)  $  EH=EHL(J)+EPG  $  EOLD=0.
      CALL HSTRESS
      P(J)=PHL(J)*1.E-9
      H(J,1)=IH1  $  H(J,3)=IH3
      PHL(J)=SHL(J)=RHL(J)=0.
 7031 IF (EQM .EQ. 0.) GO TO 707
      DIMPMCC(J)=DIMPMCC(J-1)
      IF (EPG .LT. EQM) GO TO 707
      IF (J .GT. JBEG .OR. J .EQ. JBNDM-1) GO TO 706
      IF (FRONT(1,L)*FRONT(4,L) .EQ. 0.) GO TO 706
C          SPECIAL INTEGRATION FOR FIRST CELL OF A LAYER TO OBTAIN
C              MCCLOSKEY INTEGRAL
      IF (FRONT(1,L) .LT. CPG(J) .OR. CPG(J) .LT. FRONT(4,L)) GO TO 706
      EO=FRONT(1,L)*4.186E7   $   EA=FRONT(4,L)*4.186E7
      ENN=(EO-EA)/(EPG-EA)-1.   $   NAB=MIN1(100.,2.*EO/EA+1.)
      H1=ZLAGR+DZ/2.   $   ZL=ZLAGR-DZ/2.
      VOLD=ZL*(EO-EQM*(1.+ALOG(EO/EQM)))
      DZL=DZ/NAB   $   EMSUM=0.
      DO 704 I=1,NAB
      ZL=ZL+DZL   $   EHH=EA+(EO-EA)*((H1-ZL)/DZ)**ENN
      IF (EHH .LT. EQM) GO TO 705
      VNEW=ZL*(EHH-EQM*(1.+ALOG(EHH/EQM)))
      EMSUM=0.5*DZL*(VOLD+VNEW) + EMSUM
 704  VOLD=VNEW
 705  DIMPMCC(J)=EMSUM + DIMPMCC(J)
      GO TO 707
 706  DIMPMCC(J)=DIMPMCC(J)+ZLAGR*(EPG-EQM*(1.+ALOG(EPG/EQM)))*DZ
 707  DZLAST=DZ
      IF (NPOR(M) .NE. 0) FRONT(3,L)=0.
      JBEG=JBNDM+1
 708  CONTINUE
      COEF=1.2*SQRT(2.)
```

```
      DO 709 J=1,JFIN
  709 DIMPMCC(J)=COEF*SQRT(DIMPMCC(J))*1.0E-3
      FPCT=100.
      JJ=0
      FEPG=FRONT(1,1)*4.186E7
      WRITE (6,15)  JJ,X(1),X(1),FEPG ,FRONT(1,1),FPCT,FRONT(2,1),
     1    FRONT(3,1),X(1),MATL(1,1)
      L=K=J1=1   $    M=JMAT(L)
  710 J2=MIN0(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,I),I=1,3),J,DELX(J),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),I=1,10)
  718 IF (J2 .NE. JBND(L)) GO TO 710
      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),DIMPMCC(J2),MATL(M,1)
      GO TO 710
  740 WRITE(6,70) YHL(J2),CHL(J2),DHL(J2),T(J2-1),T(J2)
      GO TO (742,743,744) NALPHA
  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
  746 CONTINUE
      IF (IPLOT(1)+IPLOT(2)+IPLOT(3)+IPLOT(4) .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 $    XPL(JJ)=X(J)
      ITITLE(9)=10HDEPTH - CM
      DO 753 NN=10,24
  753 ITITLE(NN)=10H
      L=L+1
  754 CONTINUE
      JMAX=JJ
      DO 776 I=1,4
      IF (IPLOT(I) .EQ. 0)  GO TO 776
      GO TO (756,758,760)I
  756 ITITLE(17)=10HABSORBED E  $  ITITLE(18)=10HNERGY - CA
      ITITLE(19)=10HL/G            $  GO TO 762
  758 ITITLE(17)=10HTEMP. FROM  $  ITITLE(18)=10H ABS. ENER
      ITITLE(19)=10HGY - DEG C
      GO TO 762
  760 ITITLE(17)=10HPSEUDO PRE  $  ITITLE(18)=10HSSURE AT D
      ITITLE(19)=10HEP. - KBAR
  762 CONTINUE
      L=1 $    JJ=1 $    YPL(1)=FRONT(I,1)
      DO 774 J=1,JEND
      JJ=JJ+1
      GO TO (769,770,771)  I
  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 .LT. JBND(L)) GO TO 774
      IF (J .EQ. JEND) GO TO 774
      JJ=JJ+1 $    L=L+1
```

SUBROUTINE DEPOS (Concluded)

```
      YPL(JJ)=FRONT(I,L)                                                DEPOS     497
  774 CONTINUE                                                          DEPOS     498
      CALL GRAPH4(XPL,YPL,JMAX,1,XMAX(I),XMIN(I),YMAX(I),YMIN(I),ITITLE, DEPOS     499
     1     IA)                                                          DEPOS     500
  776 CONTINUE                                                          DEPOS     501
  780 RETURN                                                            DEPOS     502
      END                                                               DEPOS     503
```

SUBROUTINE DFRACT

```
      SUBROUTINE DFRACT(SXX,SYY,STT,TXY,EXX1,EYY1,ETT1,EXY1,P,NM,NT,DHO,
     1  DOLDO,DTO,EOLD,EH,EQSTCM,EQSTGM,ELMU,RHOS,TSR,Y,YD,F,M,J,K,ALFA)
C
C         PJ          ESTIMATE OF PRESSURE
C         PA          COMPUTED PRESSURE BASED ON PJ
C         PN,PG       PRESSURES ASSOCIATED WITH NUCLEATION AND GROWTH
C         NM          RELATIVE VOID VOLUME
C         NT          VOID DENSITY, NUMBER/CM3
C         TSR(1)      GROWTH CONSTANT = 3/(4*ETA)
C         TSR(2)      GROWTH THRESHOLD, DYN/CM2
C         TSR(3)      NUCLEATION RADIUS PARAMETER, CM
C         TSR(4),     PARAMETERS IN THE NUCLEATION FUNCTION :
C           TSR(6)         NDOT = T4*EXP((P-TSR(5))/TSR(6))
C         TSR(5)      NUCLEATION THRESHOLD, DYN/CM2
C         VVO, VVA    VOID VOLUME, CM3/G
C         VGA         VOID VOLUME ASSOCIATED WITH GROWTH, CM3/G
C         VNA         VOID VOLUME ASSOCIATED WITH NUCLEATION, CM3/G
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
      VSO=1./DOLD-VVO
      PSO=P/(VSO*DOLD)
      DVO=1./DHO-1./DOLD
      IF (ABS(DVO) .LT. 1.E-9) DVO=1.E-9
      DV=DVO
      IF (TSR(M,7) .EQ. 0.) TSR(M,7)=8.*3.1416*TSR(M,3)**3*TSR(M,4)
C ********
C               BEGIN SUBCYCLING LOOP FOR CASE OF LARGE STRAIN
C ********
      NLOOP=MAX1(1.,-2.*DV*EQSTCM/VSO/TSR(M,5)+0.5,2.5*TSR(M,1)*DTO*
     1  AMIN1(P-TSR(M,2),TSR(M,2)))
100   DELV=DV/NLOOP
      EXX=EXX1*DELV/DVO
      EYY=EYY1*DELV/DVO
      ETT=ETT1*DELV/DVO
      EXY=EXY1*DELV/DVO
      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./DOLDO)/DVO+EOLD
      TEMP1=1.-RHOS*EQSTGM*E/EQSTCM
C
C               ESTIMATE OF PRESSURE BASED ON STRAIN, GROWTH, NUCLEATION
      PN=0.
      YS=VSO**2*RHOS/EQSTCM
      YSC=YS*(PSO+RHOS*EQSTGM*DE)
      DVS=DELV
      PG=AMAX1((YSC-DELV)/YS,EQSTCM*(1./RHOS/(VSO+DELV)-TEMP1))
      PS=PG
      PJ=PG
      IF(0.5*(PJ+PSO) .GT. AMAX1(TSR(M,2),TSR(M,5)))GO TO 300
      IF (DELV .GT. 0.) PN=2.*TSR(M,6)*ALOG(DELV*DH/TSR(M,7)/DT)+
     1  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.
      YGC= VVO*(XP-1.)-YG*PSO
      YN=XN/(2.*TSR(M,6))
      YNC=XN-YN*PSO
      PG=(DELV-YSC-YGC-YNC)/(-YS+YG+YN)
```

SUBROUTINE DFRACT (Continued)

```
      CH=1.                                                             DFRACT2   77
      IF (0.5*(PG+PSO).GT. TSR(M,2))YG=YGC=CH=0.                        DFRACT2   78
      IF (0.5*(PG+PSO).GT. TSR(M,5))YN=YNC=CH=0.                        DFRACT2   79
      IF(CH.EQ.0.)PG=(DELV-YSC-YGC-YNC)/(-YS+YG+YN)                     DFRACT2   80
      IF(DELV.GT.0. .AND. PSO .LT. TSR(M,2))PG=AMIN1(PG,TSR(M,2))       DFRACT2   81
      PJ=AMAX1(PS,PG,PN)                                                10/8/79   34
  150 DVS=1./RHOS/(PJ/EQSTCM+TEMP1)-VSO                                 DFRACT2   83
      VVA=VVO+DELV-DVS                                                  DFRACT2   84
      NC=0.                                                             DFRACT2   85
C ********                                                              DFRACT2   86
C                BEGIN ITERATION LOOP                                   DFRACT2   87
C ********                                                              DFRACT2   88
200   NC=NC+1                                                           DFRACT2   89
      VV=VVO+DELV-DVS                                                   DFRACT2   90
      PJ=PA=EQSTCM*(1./RHOS/(VSO+DVS)-TEMP1)                            DFRACT2   91
      PN=AMIN1(0.5*(PA+PSO)-TSR(M,5),0.)                                10/8/79   35
      IF (PN .LT. 0.) PN=EXP(PN/TSR(M,6))                               DFRACT2   94
      VNA=TSR(M,7)*PN*DT/DH                                             DFRACT2   95
      VGA=VVO                                                           DFRACT2   96
      PG=AMIN1(0.5*(PA+PSO)-TSR(M,2),0.)                                10/8/79   36
      IF (PG .LT. 0.) VGA=VVO*EXP(A1*PG)                                DFRACT2   98
      VVA=VGA+VNA                                                       DFRACT2   99
      DVSA=1./RHOS/(PJ/EQSTCM+TEMP1)-VSO                                DFRACT2  100
      DELVA=DVSA+VVA-VVO                                                DFRACT2  101
C         TEST FOR COMPLETION OF ITERATIONS                             DFRACT2  103
      IF (ABS(DELVA-DELV)/VSO .LT. 2.E-5 .AND. ABS(DVS-DVSA)/VSO .LT.   DFRACT2  104
     1  1.E-5) GO TO 300                                                DFRACT2  105
      IF(NC.LT.30) GO TO 250                                            DFRACT2  106
      IF(NTRY.LT.5)GO TO 450                                            DFRACT2  107
      PRINT 1250,J,K,M,PJ,DELV,DELVA,DELVB,DELVC                        DFRACT2  108
      GO TO 300                                                         DFRACT2  109
C         DELVA IS RECENT VALUE, DELVB IS LARGER STORED VALUE, AND      DFRACT2  110
C         DELVC IS SMALLER STORED VALUE.                                DFRACT2  111
  250 IF(NC.EQ.1) GO TO 270                                             DFRACT2  112
      IF(NC .EQ. 2) GO TO 260                                           DFRACT2  113
      IF (DELVC .GT. DELV) GO TO 265                                    DFRACT2  114
      IF (DELVB .LT. DELV) GO TO 260                                    DFRACT2  115
      IF (DELVA .GT. DELV) GO TO 265                                    DFRACT2  116
  260 DVS=DVSA+(DVSB-DVSA)/(DELVB-DELVA)*(DELV-DELVA)                   DFRACT2  117
      IF(MOD(NC+2,3).EQ.0) DVS=.5*(DVSA+DVSB)                           DFRACT2  118
      GO TO 275                                                         10/8/79   37
  265 DVS=DVSA+(DVSC-DVSA)/(DELVC-DELVA)*(DELV-DELVA)                   DFRACT2  120
      IF(MOD(NC+2,3).EQ.0) DVS=.5*(DVSA+DVSC)                           DFRACT2  121
      GO TO 275                                                         10/8/79   38
  270 PJ=PA+(DELV-DELVA)/(VGA*A1/2.-YS +VNA/2./TSR(M,6))                DFRACT2  123
      PN=PJ                                                             DFRACT2  124
      IF (VNA+DELV-DELVA .GT. 0. .AND. VNA .GT. 0.) PN=2.*TSR(M,6)*     DFRACT2  125
     1  ALOG((VNA+DELV-DELVA)/VNA) + PA                                 DFRACT2  126
      PJ=AMAX1(PJ,0.5*(PN+PJ))                                          DFRACT2  127
      PJ=PA+SIGN(AMIN1(ABS(PJ-PA),DPJ),DELVA-DELV)                      DFRACT2  128
      DVS=VSO*(1./(1.+(PJ*YS-YSC)/VSO)-1.)                              DFRACT2  129
  275 IF(NC-2) 290,285,280                                              10/8/79   39
  280 IF((DELVB.GT.DELV.AND.DELVA.GT.DELVB).OR.                         10/8/79   40
     C (DELVA.LT.DELVC.AND.DELVC.LT.DELV)) GO TO 200                    10/8/79   41
      IF(DELVC.GT.DELV.OR.(DELVB.GT.DELV.AND.DELVA.GT.DELV))            10/8/79   42
     C GO TO 290                                                        10/8/79   43
  285 DELVC = DELVA                                                     10/8/79   44
      DVSC = DVSA                                                       10/8/79   45
      GO TO 292                                                         10/8/79   46
  290 DELVB = DELVA                                                     10/8/79   47
      DVSB = DVSA                                                       10/8/79   48
      IF(NC.EQ.1) GO TO 200                                             10/8/79   49
  292 IF(DELVB.GT.DELVC) GO TO 200                                      10/8/79   50
      DELVA = DELVB                                                     10/8/79   51
      DVSA = DVSB                                                       10/8/79   52
      DELVB = DELVC                                                     10/8/79   53
      DVSB = DVSC                                                       10/8/79   54
      DELVC = DELVA                                                     10/8/79   55
      DVSC = DVSA                                                       10/8/79   56
      GO TO 200                                                         DFRACT2  147
C         ENDING ROUTINE                                                DFRACT2  148
C                                                                       DFRACT2  149
300   NM=VVA*DH                                                         DFRACT2  150
      NT=NT*DH/DOLD+TSR(M,4)*PN*DT                                      DFRACT2  151
      IF(NM .GT. 0.6) GO TO 400                                         DFRACT2  152
```

SUBROUTINE DFRACT (Concluded)

```
      BETA=2.*TXY*ALFA/NLOOP                                            DFRACT2   153
      ELMUF=2.*ELMU*AMAX1(1.-SMF*VVA*DH,0.)                             DFRACT2   154
      WS1=0.6667*(DOLD-DH)/(DOLD+DH)                                    DFRACT2   155
      TXY=TXY+ELMUF*EXY+(SYY-SXX)*ALFA/NLOOP                            10/8/79    57
      SXX=SXX+ELMUF*(EXX-WS1)+BETA                                      DFRACT2   157
      SYY=SYY+ELMUF*(EYY-WS1)-BETA                                      DFRACT2   158
      STT=STT+ELMUF*(ETT-WS1)                                           DFRACT2   159
      WS4=SXX**2+SYY**2+STT**2+2.*TXY**2                                DFRACT2   160
      YE=Y*F*AMAX1(1.-4.*VVA*DH,0.)                                     DFRACT2   161
      IF (WS4 .LT. YE**2/1.5) GO TO 340                                 DFRACT2   162
      WS3=YE/SQRT(1.5*WS4)                                              DFRACT2   163
      PTERM=(DOLD-DH)/(DOLD+DH)/DT/TSR(M,1)                             DFRACT2   164
      WS5=1.5/TSR(M,1)/DT                                               DFRACT2   165
      SXX=SXX*WS3+EXX*WS5-PTERM                                         DFRACT2   166
      SYY=SYY*WS3+EYY*WS5-PTERM                                         DFRACT2   167
      STT=STT*WS3+ETT*WS5-PTERM                                         DFRACT2   168
      TXY=TXY*WS3+EXY*WS5                                               10/8/79    58
340   CONTINUE                                                          DFRACT2   170
      PSO=PJ                                                            DFRACT2   171
      P=PJ*(VSO+DVS)*DH                                                 DFRACT2   172
      Y=YT                                                              DFRACT2   173
      VVO=VVA                                                           DFRACT2   174
      VSO=VH-VVA                                                        DFRACT2   175
380   DOLD=DH                                                           DFRACT2   176
      RETURN                                                            DFRACT2   177
C                                                                       DFRACT2   178
C        END WITH SEPARATION                                            DFRACT2   179
 400  P=0.                                                              DFRACT2   180
      Y=0.                                                              DFRACT2   181
      SXX=0.                                                            DFRACT2   182
      SYY=0.                                                            DFRACT2   183
      STT=0.                                                            DFRACT2   184
      TXY=0.                                                            DFRACT2   185
      NM=-ABS(NM)                                                       DFRACT2   186
      RETURN                                                            DFRACT2   187
C                                                                       DFRACT2   188
C              PROVISION FOR ABORT IN CASE OF ITERATION FAILURE         DFRACT2   189
450   NTRY=NTRY+1                                                       DFRACT2   190
      DV=1./DHO-1./DOLD                                                 DFRACT2   191
      NLOOP=MAX1(3.,-4.*2.**NTRY*DV*EQSTCM/VSO/TSR(M,5)+0.5)            DFRACT2   192
      GO TO 100                                                         DFRACT2   193
C        FORMATS                                                        DFRACT2   194
1250  FORMAT (30H   ITERATION FAILURE IN DFRACT/5H J,K=2I2,3H M=I2,     DFRACT2   195
     1  4H PJ=1PE10.3,6H DELV=E10.3,7H DELVA=E10.3,7H DELVB=E10.3,      DFRACT2   196
     2  7H DELVC=E10.3)                                                 DFRACT2   197
      END                                                               DFRACT2   198
```

SUBROUTINE EDIT

```
      SUBROUTINE EDIT                                                   EDIT   2
C                                                                       EDIT   3
C         *  EDIT LISTS COOROINATE QUANTITIES FOR TIME OF TEOIT          EDIT   4
C     INPUT - JSTAR.                                                    EDIT   5
C     OUTPUT - NTEX.                                                    EDIT   6
C                                                                       EOIT   7
      INTEGER H,POROUS,PRESS,RINTER,SOLID,SPALL                         PUFCOM 2
      REAL MATL,NEM,NET,NEMH,NETH                                       PUFCOM 3
C             MISCELLANEOUS                                             PUFCOM 4
      COMMON AZERO(1),CEF,CKS,OAVG,OELTIM,DISCPT(10),DOLO,ORHO,OTMAX,   PUFCOM 5
     1  DTMIN,OTN,OTNH,DU,OX,EOLD,F,FAC,FIRST,J,JCYCS,JINIT,            PUFCOM 6
     2  JFIN,JREZON(15),JSMAX,JSTAR,JTS,LSUB(30),M,MAXPR(30),N,NCYCS,   PUFCOM 7
     3  NEDIT,NPERN,NR,NREZON,NSCRB(6),NSEPRAT,NSPALL,NTEDT,            PUFCOM 8
     4  NTEX,NTR(15),POLD,P6(20),R(30),RLAST,SLAST,SMAX,TEOIT(50),      PUFCOM 9
     5  TF,TIME,TJ,TREZON,TS,T6(20),ULAST,UOLD,UZERO,XLAST,XNOW,XOLD    PUFCOM10
     1  ,XJDIT(20)                                                      PUFCOM11
C             HALFSTEP VALUES                                           PUFCOM12
      COMMON DH,OHLAST,DUH,EH,PH,RH,RHLAST,SH,SHLAST,UH,UHLAST,XH,XHLASTPUFCOM13
     1  ,NEMH,NETH                                                      PUFCOM14
C             CONOITION INOICATORS                                      PUFCOM15
      COMMON INF,LINTER,MIRROR,NORMAL,POROUS,PRESS,RINTER,SOLIO,SPALL   PUFCOM16
C             CELL LAYOUT                                               PUFCOM17
      COMMON DXX(30),JBND(30),JMAT(30),NAUTO,MATL(6,2),NLAYER,NMTRLS,   PUFCOM18
     1  THK(30)                                                         PUFCOM19
C                                                                       PUFCOM20
C             COORDINATE ARRAYS                                         COORDC02
      COMMON/COORD/X(200),X0(200),CHL(200),DHL(200),DPDD(200),DPDE(200),COOROC03
     1  EHL(200),H(200,3),NEM(200),NET(200),PHL(200),RHL(200),SDT(200), COORDC04
     2  SHL(200),T(200),U(200),YHL(200),ZHL(200)                         COORDC05
      COMMON /IND/ IEOS(6),INDK(20),NALPHA,NCMP(6),NFR(6),NPOR(6),      INDCOM 2
     1  NDS(6),NPR(6),NCON(6),NVAR(6)                                   INDCOM 3
      COMMON /PES/ LVMAX,LVTOT,LVAR(200),COM(4000)                      EDIT  11
      DIMENSION P1(300),P2(300),EMOM(300)                               EDIT  12
C                                                                       EOIT  13
C                                                                       EDIT  14
C                                                                       EDIT  15
C             PRINTOUT FOR EACH EOIT                                    EDIT  16
      NTEX=NTEX+1                                                       EDIT  17
      CALL SECOND(CHANGE)  $  OUR=CHANGE-FIRST                          EDIT  18
      JSTARD=MIN0(JSTAR+1,JFIN-1)  $  NPTS=JSTARD-JINIT+1               EOIT  19
      WRITE (6,1025)(DISCPT(I),I=1,10)                                  EDIT  20
      WRITE(6,1026)NTEX,N,TIME,JSTAR,DUR,DTNH                           EDIT  21
      EMSUM = EMOM(JINIT) = 0.                                          EOIT  22
      J1=JINIT   $   L=1   $   M=JMAT(L)                                EDIT  23
    4 J2=MIN0(JSTARO,JBND(L))                                           EOIT  24
      NJ=J2-J1+1                                                        EDIT  25
      IF (NPR(M) .EQ. 1) GO TO 7                                        EDIT  26
      IF (NFR(M) .EQ. 1 .OR. NFR(M) .EQ. 2) GO TO 7                     EDIT  27
      IF (NFR(M) .EQ. 3) GO TO 5                                        EDIT  28
      IF (NPOR(M) .EQ. 3) GO TO 10                                      EDIT  29
      IF (NDS(M).EQ.0 .OR. NDS(M).EQ.1 .OR. NDS(M).EQ.4) GO TO 5        EDIT  30
      IF (NPOR(M) .EQ. 4) GO TO 9                                       EDIT  31
      GO TO 7                                                           EDIT  32
    5 DO 6 J=J1,J2                                                      EDIT  33
      P1(J)=YHL(J)                                                      EDIT  34
    6 P2(J)=SHL(J)-PHL(J)                                               EDIT  35
      IF (NFR(M) .GE. 3) GO TO 10                                       EDIT  36
      N1 = 10H  YIELO     $  N2 = 10H DEVIATOR                          EDIT  37
      GO TO 13                                                          EDIT  38
    7 DO 8 J=J1,J2                                                      EDIT  39
      P1(J)=NEM(J)                                                      EOIT  40
    8 P2(J)=NET(J)                                                      EDIT  41
      N1 = 10H   NEM      $  N2 = 10H   NET                             EOIT  42
      IF (NPR(M) .EQ. 1) N1 = 10H  FBURN                                EDIT  43
      IF (NFR(M) .NE. 0) N1=10H   RVV                                   EOIT  44
      GO TO 13                                                          EDIT  45
    9 N1=10H   EVP      $  N2=10H DEVIATOR                              EDIT  46
      DO 91 J=J1,J2                                                     EDIT  47
```

SUBROUTINE EDIT (Concluded)

```
      P1(J)=NEM(J)
 91   P2(J)=SHL(J)-PHL(J)
      GO TO 13
10    DO 11 J=J1,J2
      IF (H(J,3)-2) 101,11,102
 101  P1(J)=NEM(J)
      P2(J)=NET(J)
      GO TO 11
 102  LV=LVAR(J)
      IF (NPOR(M) .EQ. 3 .AND. H(J,3) .EQ. 5R    M) LV=LVAR(J)+3
      P1(J)=COM(LV)
      P2(J)=COM(LV+1)
 11   CONTINUE
      N1=10H Y/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
      IF(DHL(J2) .GT. 1.) WRITE(6,1028)(J,X(J),U(J),RHL(J),PHL(J),SHL(J)
     1,EHL(J),DHL(J),CHL(J),(H(J,I),I=1,3),MATL(M,1),EMOM(J),P1(J),P2(J)
     2,J=J1,J2)
      IF(DHL(J2) .LE. 1.) WRITE(6,1027)(J,X(J),U(J),RHL(J),PHL(J),SHL(J)
     1,EHL(J),DHL(J),CHL(J),(H(J,I),I=1,3),MATL(M,1),EMOM(J),P1(J),P2(J)
     2,J=J1,J2)
      IF (J2 .EQ. JSTARO) GO TO 20
      J1=J2+1
      L = L+1  $  M = JMAT(L)  $  GO TO 4
20    CONTINUE
      RETURN
1025  FORMAT (1H0,10A10)
 1026 FORMAT(18H0     TIME EDIT NO.I3,7H AT N =I5,8H, TIME =1PE12.5,
     1  14H SECS, JSTAR =I5,14H, CALC TIME IS 0PF10.3,13H SECS, DTNH =
     2  1PE10.3,5H SECS/)
 1027 FORMAT(I5,0PF9.6,F9.0,1P4E10.3,0PF8.6,1PE11.4,3R2,1X,A9,1PE10.3,
     1  1P2E11.3)
 1028 FORMAT(I5,0PF9.6,F9.0,1P4E10.3,0PF8.4,1PE11.4,3R2,1X,A9,1PE10.3,
     1  1P2E11.3)
1029  FORMAT (4X,1HJ,8X,1HX,8X,1HU,7X,3HRHL,7X,3HPHL,7X,3HSHL,7X,3HEHL,
     1  5X,3HDHL,8X,3HCHL,2X,4HCOND,17X,3HMOM,1X,A10,1X,A10/
     2 5H CELL,7X,2HCM,3X,6HCM/SEC,3(10H  DYN/CM2 ),6X,4HERGS,2X,
     3 6HGM/CM3,5X,6HCM/SEC,22X,4HTAPS)
      END
```

SUBROUTINE EOSTAB

```
      SUBROUTINE EOSTAB(NCALL,IN,XN,YN,ZN)                              EOSTAB     2
      DIMENSION X(30),Z(30),EX(30)                                      EOSTAB     3
      IF (NCALL .GT. 0) GO TO 100                                       EOSTAB     4
C                                                                       EOSTAB     5
C          INITIALIZE AND READ DATA                                     EOSTAB     6
C                                                                       EOSTAB     7
      READ (IN,1001) A1,IMAX,I2,I3                                      10/8/79    1
      IF (I2 .NE. 10H VOLUME    ) I2=10H DENSITY                        EOSTAB     9
      IF (I3 .NE. 10H LOG       ) I3=10H LINEAR                         EOSTAB    10
      PRINT 1001,A1,IMAX,I2,I3                                          EOSTAB    11
      PRINT 1003,IN                                                     EOSTAB    12
      READ (IN,1002) A1,(X(I),Z(I), I=1,IMAX)                           EOSTAB    13
      PRINT 1012,A1,(X(I),Z(I), I=1,IMAX)                               EOSTAB    14
C         VOLUME TRANSFORMATION                                         EOSTAB    15
      IF (I2 .NE. 10H VOLUME    ) GO TO 45                              EOSTAB    16
      DO 30 I=1,IMAX                                                    EOSTAB    17
30    X(I)=1./X(I)                                                      EOSTAB    18
45    IM1=IMAX-1                                                        EOSTAB    19
      DO 50 I=1,IM1                                                     EOSTAB    20
50    EX(I)=(Z(I+1)-Z(I))/(X(I+1)-X(I))                                 EOSTAB    21
      IF (I3 .NE. 10H LOG       ) GO TO 80                              EOSTAB    22
      DO 65 I=1,IM1                                                     EOSTAB    23
      IF (Z(I) .LE. 0. .OR. Z(I+1) .LE. 0.) GO TO 65                    EOSTAB    24
      EX(I)=ALOG(Z(I+1)/Z(I))/ALOG(X(I+1)/X(I))                         EOSTAB    25
65    CONTINUE                                                          EOSTAB    26
80    IM2=IMAX-2                                                        EOSTAB    27
      N1=2                                                              EOSTAB    28
      NM=IMAX-1                                                         EOSTAB    29
      NORDER=1                                                          EOSTAB    30
      IF (X(1) .LT. X(2)) RETURN                                        EOSTAB    31
      NORDER=0                                                          EOSTAB    32
      N1=IMAX-1                                                         EOSTAB    33
      NM=2                                                              EOSTAB    34
      IM2=IMAX-1                                                        10/8/79    2
      RETURN                                                            EOSTAB    35
C                                                                       EOSTAB    36
C         CALCULATE PRESSURE                                            EOSTAB    37
100   IT=N1-NORDER                                                      EOSTAB    38
      IF (XN .LT. X(N1)) GO TO 175                                      EOSTAB    39
      IT=NM-1+NORDER                                                    EOSTAB    40
      IF (XN .GT. X(NM)) GO TO 175                                      EOSTAB    41
      DO 140 I=2,IM2                                                    EOSTAB    42
      N4=I+1                                                            EOSTAB    43
      IF (NORDER .EQ. 0) N4=IMAX-I+1                                    10/8/79    3
      IT=N4-NORDER                                                      EOSTAB    45
      IF (XN .LT. X(N4)) GO TO 175                                      EOSTAB    46
140   CONTINUE                                                          EOSTAB    47
175   IF (I3 .EQ. 10H LOG       ) GO TO 190                             EOSTAB    48
180   ZN=Z(IT)+(XN-X(IT))*EX(IT)                                        EOSTAB    49
      RETURN                                                            EOSTAB    50
190   IF (Z(IT) .LE. 0. .OR. Z(IT+1) .LE. 0.) GO TO 180                 EOSTAB    51
      ZN=Z(IT)*(XN/X(IT))**EX(IT)                                       EOSTAB    52
      RETURN                                                            EOSTAB    53
1001  FORMAT(A10,I10,2A10)                                              EOSTAB    54
1002  FORMAT(A10,6E10.3/(10X,6E10.3))                                   EOSTAB    55
1003  FORMAT(1H+,79X,4H IN=,I2,11H EOSTAB P-V  )                        EOSTAB    56
1012  FORMAT(A10,1P6E10.3/(10X,6E10.3))                                 EOSTAB    57
      END                                                               EOSTAB    58
```

SUBROUTINE EQST

```
      SUBROUTINE EQST(EJ,DJ,PJ,MJ,CJ,DPDDJ,DPDEJ)
C
C     COMPUTES PRESSURE AND SOUND SPEED FOR SOLIDS AND EXPLOSIVES
C        *  MIE-GRUNEISEN FOR COMPRESSION
C             PUFF HUGONIOT IN P - MU FORM
C             MURNAGHAN HUGONIOT FORM (FOR EQSTS=1.0)
C             LINEAR US-UP HUGONIOT FORM (FOR EQSTS=2.0)
C        *  EXPANSION EQUATION OF STATE FOR DENSITIES LESS THAN RHOS
C        *  POLYTROPIC GAS EQUATION FOR EXPLOSIVES (NPR=1)
C     INPUT - FORMAL PARAMETERS EJ, DJ, MJ, CJ.
C     OUTPUT - PJ, CJ.
C
C              NAMED COMMON
      REAL MU,MUM
      COMMON /EQS/  EQSTA(6),EQSTC(6),EQSTD(6),EQSTE(6),EQSTG(6),
     1  EQSTH(6),EQSTN(6),EQSTS(6),EQSTV(6),CZQ(6),CWQ(6),C2(6)
      COMMON /MELT/ EMELT(6,5),SPH(6),THERM(6,8)
      COMMON /RHO/ RHO(6),RHOS(6)
      COMMON /TSR/ TSR(6,30),EXMAT(6,20),TENS(6,3)
      COMMON /Y/ Y0(6),YADD(6),MU(6),MUM,YADDM
      COMMON /IND/ IEOS(6),INDK(20),NALPHA,NCMP(6),NFR(6),NPOR(6),
     1  NDS(6),NPR(6),NCON(6),NVAR(6)
      DIMENSION NHUG(6),AMURN(6),BPMURN(6),S1(6)
C
C        EXPLANATION OF SOME MODIFIED PUFF EXPANSION MODEL PARAMETERS
C
C       * ABSOLUTE VALUE OF EQSTV IS EXPONENT IN GRUNEISEN EXPRESSION,
C --          DEFAULT VALUE OF EXPONENT FOR EQSTV=0. IS 0.5
C       * IF EQSTV .GT. 0., THEN MCCLOSKEY-THOMPSON LOG VARIATION IS
C --          USED FOR EJ .GT. SUBL ENERGY
C       * EQSTA IS THE COEFFICIENT OF THE SECOND TERM ASSUMED IN THE
C --          GRUNEISEN SERIES AND A NONZERO VALUE INDICATES THAT THE
C --          PRESSURE-DENSITY SLOPES OF MIE-GRUNEISEN EOS AND EXPANSION
C --          EOS HAVE BEEN MATCHED AT THE INITIAL SOLID DENSITY.
C --          IF EQSTA=0. OR IS UNSPECIFIED, THEN THE PRESSURE-VOLUME
C             SLOPE IS NOT MATCHED.
C
C
C ****          INITIALIZATION PORTION                                *****
C
      IF (EQSTN(MJ) .GT. 0.) GO TO 200
      EQSTN(MJ) = 1.
      IF (EQSTG(MJ)*EQSTE(MJ)*RHOS(MJ) .NE. 0.) EQSTN(MJ) = EQSTC(MJ)/
     1 (EQSTG(MJ)*EQSTE(MJ)*RHOS(MJ))
      IF (EQSTV(MJ) .GT. 0.) PRINT 1005
      ENN=ABS(EQSTV(MJ))
      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*(EQSTH(MJ)-EQSTG(MJ)))/EQSTG(MJ)
      IF (AMURN(MJ)+EQSTN(MJ) .GT. 0.) GO TO 30
      PRINT 1055,EQSTN(MJ),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
C        P = A*((D/RHOS)**B0P-1.)
C        -A-=B0/B0P IS READ AS -C-. -B0P- IS READ IN AS -D-
40    AMURN(MJ) = EQSTC(MJ)
      BPMURN(MJ) = EQSTD(MJ)
      EQSTC(MJ) = EQSTC(MJ)*EQSTD(MJ)
      EQSTD(MJ) = 0.5*EQSTC(MJ)*(EQSTD(MJ)-1.)
```

```
      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-
60    S1(MJ) = EQSTD(MJ)
      EQSTC(MJ) = RHOS(MJ)*EQSTC(MJ)**2
      EQSTD(MJ) = EQSTC(MJ)*(2.*EQSTD(MJ)-1.)
      PRINT 1020,EQSTC(MJ),EQSTD(MJ)
      GO TO 180
180   IF (EQSTN(MJ) .EQ. 1.) PRINT 1050
      RETURN
C
C ****   COMPUTATION PORTION                                        ****
C
200   IF (NPR(MJ) .EQ. 1) GO TO 400
      AMU=1.333*MUM    $    IF (EJ .GT. EMELT(MJ,1))  AMU=0.
      VJ=RHOS(MJ)/DJ    $    EMU=(1.-VJ)/VJ
      IF (EMU .GE. 0.) GO TO 300
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.+ALOG(EJ/
     1  EQSTE(MJ))
      ERAT=EJ/EQSTE(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=0.  $  IF (ENU2 .GT. -10.) EX2=EXP(ENU2)
      TS1=TS1*(1.-EX2)
      TS2=EQSTH(MJ)+GHNU  $  PJ=(EJ-TS1)*DJ*TS2
      IF (EJ .GT. EMELT(MJ,1)) PJ=AMAX1(0.,PJ)
      IF (CJ .EQ. 1.) GO TO 500
      DPDDJ=(EJ-TS1)*(TS2+ENN*GHNU)+TS2*(EQSTE(MJ)*ESUBC-TS1)
     1  *(EQSTN(MJ)+ERAT*EQSTA(MJ))/ESUBC*(2.*VJ-1.)*VJ
      DPDEJ=DJ*TS2*(1.+EQSTA(MJ)*EX2*VJ*(1.-VJ))
      IF (EQSTV(MJ) .LE. 0. .AND. EJ .GT. EQSTE(MJ)) DPDEJ=DJ*TS2
      EX1=EX2*(1.-ENU2)
      IF (EQSTV(MJ) .GT. 0. .AND. EJ .GT. EQSTE(MJ)) DPDEJ=DJ*TS2*(1.-
     1  EQSTE(MJ)/EJ*(1.-EX1))
      CSQ=DPDDJ+(EJ-TS1)*TS2/DJ*DPDEJ+AMU/DJ
      GO TO 450
C                EQST FOR COMPRESSED ZONES
300   IF (NHUG(MJ)-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 = AMURN(MJ)*((DJ/RHOS(MJ))**BPMURN(MJ)-1.)
      DPHDD = (EQSTC(MJ)+BPMURN(MJ)*PH)/DJ
      GO TO 370
C        LINEAR US-UP HUGONIOT
330   PH = EQSTC(MJ)*(1.-VJ)/(1.-S1(MJ)*(1.-VJ))**2
      DPHDD = VJ/DJ*(1.+S1(MJ)*(1.-VJ))/(1.-S1(MJ)*(1.-VJ))**3
C        COMPUTE PRESSURE DERIVATIVES AND SOUND SPEED
370   GF = 1.-0.5*EQSTG(MJ)*(1.-VJ)
      PJ = PH*GF+EQSTG(MJ)*RHOS(MJ)*EJ
      IF (CJ .EQ. 1.) GO TO 500
      DPDDJ = DPHDD*GF-0.5*PH*EQSTG(MJ)/RHOS(MJ)/(1.+EMU)**2
      DPDEJ = EQSTG(MJ)*RHOS(MJ)
      CSQ = DPDDJ+PJ*DPDEJ/DJ**2+AMU/DJ
      GO TO 450
C                 EQST FOR EXPLOSIVE (NPR = 1)
400   PJ = EQSTG(MJ)*DJ*EJ
      DPDEJ=EQSTG(MJ)*DJ
      DPDDJ=EQSTG(MJ)*EJ
      CSQ=EQSTG(MJ)*(EJ+PJ/DJ)
C                SOUND SPEED COMPUTATION
```

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-THOMPSO
     1N LOG VARIATION ABOVE EQSTE*)
 1007 FORMAT(* EQST: EXPONENT IN GRUNEISEN EXPRESSION =*1PE10.3)
 1009 FORMAT(* EQSTA=*1PE10.3,* IS COEFFICIENT OF SECOND TERM ASSUMED IN
     1 GRUNEISEN SERIES USED FOR IMPROVING EXPANSION EOS MODEL*)
 1010 FORMAT(* MURNAGHAN HUGONIOT, CONSTANTS CHANGED TO EQSTC=*1PE10.3,*
     1, EQSTD=*1PE10.3)
 1020 FORMAT(* LINEAR US-UP HUGONIOT, CONSTANTS CHANGED TO EQSTC=*1PE10.
     13,*, EQSTD=*1PE10.3)
1050  FORMAT(* EXPANSION PORTION OF EQUATION OF STATE IS INCOMPLETE*)
 1055 FORMAT(* EXPANSION EOS WILL BE UNSTABLE ABOVE SUBLIMATION
     1  ENERGY FOR CHOSEN VALUE OF EQSTA*/*----------
     2  EQSTN=*1PE10.3,*ADDITIONAL EXPONENT=*E10.3)
500   RETURN
      END
```

SUBROUTINE EQSTPF

```
      SUBROUTINE EQSTPF(NCALL,IN,M,CJ,D,E,P)                            EQSTPF 2
C                                                                       EQSTPF 3
C         EQSTPF COMPUTES PRESSURE FROM A THREE-PHASE EQUATION OF STATE EQSTPF 4
C         DEVELOPED BY PHILCO-FORD.  ROUTINE HAS TWO PARTS, ONE FOR     EQSTPF 5
C         READING AND INITIALIZING AND THE OTHER FOR COMPUTING PRESSURE. EQSTPF 6
C                                                                       EQSTPF 7
C     READ INPUT (NCALL=0).  CALL IS FROM GENRAT.                       EQSTPF 8
C        INPUT  - NCALL, IN, M, AND MATERIAL PROPERTY CARDS             EQSTPF 9
C        OUTPUT - PRINTS CARD IMAGES, ORGANIZES DATA INTO ARRAYS        EQSTPF10
C     COMPUTE PRESSURE (NCALL=1)  CALL IS FROM HSTRESS USUALLY          EQSTPF11
C        INPUT  - NCALL, M, CJ, D, E                                    EQSTPF12
C        OUTPUT - P  (CURRENT PHASE OR STATE OF MATERIAL IS AVAILABLE)  EQSTPF13
C                                                                       EQSTPF14
C               NAMED COMMON                                            EQSTCOM2
      REAL MU,MUM                                                       EQSTCOM3
      COMMON /EQS/  EQSTA(6),EQSTC(6),EQSTD(6),EQSTE(6),EQSTG(6),       EQSTCOM4
     1  EQSTH(6),EQSTN(6),EQSTS(6),EQSTV(6),CZQ(6),CWQ(6),C2(6)         EQSTCOM5
      COMMON /MELT/ EMELT(6,5),SPH(6),THERM(6,8)                        EQSTCOM6
      COMMON /RHO/ RHO(6),RHOS(6)                                       EQSTCOM7
      COMMON /TSR/ TSR(6,30),EXMAT(6,20),TENS(6,3)                      EQSTCOM8
      COMMON /Y/ Y0(6),YAOD(6),MU(6),MUM,YADOM                          EQSTCOM9
      DIMENSION A1(6),A2(6),B(6),BP(6),C1(6),CBT(6),CC(6),CV(6),D1(6),  EQSTPF16
     1  DEDV(6),EBL(6),EBS(6),EC(6),EES(6),ELO(6),EO(6),EOVO(6),EPS1(6),EQSTPF17
     2  EPS2(6),ESO(6),EVO(6),HDCT(6),PC(6),PVO(6),TM(6),VC(6),VLO(6),  EQSTPF18
     3  VO(6),VSO(6),VVO(6),WT(6),Y1(6),Y3(6),ZC(6),ZKO(6),ZK1(6),ZK2(6)EQSTPF19
     4  ,ZN(6),ZM(6)                                                    EQSTPF20
      DATA ACC, RI /1.E-4, 8.3144E7/                                    EQSTPF21
C                                                                       EQSTPF22
C ******        BRANCH TO INITIALIZATION OR COMPUTATION PORTIONS        EQSTPF23
      IF(NCALL .EQ. 1) GO TO 200                                        EQSTPF24
C ******                                                         ****** EQSTPF25
C               READ INPUT DATA AND INITIALIZE CONSTANTS                EQSTPF26
C ******                                                         ****** EQSTPF27
      IND = 5H                                                          EQSTPF28
      READ(IN,1101) Z1,C1(M),DLM,DSM,D1(M),HLB,HLM,HSM                  EQSTPF29
      WRITE(6,1101) Z1,C1(M),DLM,DSM,D1(M),HLB,HLM,HSM                  EQSTPF30
      WRITE(6,1102) IND,IN                                              EQSTPF31
      READ(IN,1101) Z1,HVB,HVM,TBK,TCK,TMK,WT(M),ZKO(M)                 EQSTPF32
      WRITE(6,1101) Z1,HVB,HVM,TBK,TCK,TMK,WT(M),ZKO(M)                 EQSTPF33
      WRITE(6,1102) IND,IN                                              EQSTPF34
      VO(M) = 1./RHOS(M)                                                EQSTPF35
      ESO(M)=HSM                                                        EQSTPF36
      IF(DSM .GT. 0.) GO TO 50                                          EQSTPF37
C          COMPUTE -DSM- IF UNSPECIFIED                                 EQSTPF38
      ERG = EQSTG(M)*RHOS(M)*ESO(M)                                     EQSTPF39
      EMU = -ERG/(EQSTC(M)+ERG)                                         EQSTPF40
      EMU = -ERG/(EQSTC(M)+(EQSTD(M)+EQSTS(M)*EMU)*EMU+ERG)             EQSTPF41
      NC2=0                                                             EQSTPF42
40    EMUO = EMU                                                        EQSTPF43
      NC2=NC2+1                                                         EQSTPF44
      IF(NC2 .GT. 20) GO TO 42                                          EQSTPF45
      P = EMU*(EQSTC(M)+EMU*(EQSTD(M)+EMU*EQSTS(M))+ERG)+ERG            EQSTPF46
      PP = EQSTC(M)+ERG+EMU*(2.*EQSTD(M)+3.*EMU*EQSTS(M))               EQSTPF47
      EMU = EMU-P/PP                                                    EQSTPF48
      IF(ABS(EMU-EMUO) .GT. ACC) GO TO 40                               EQSTPF49
      GO TO 44                                                          EQSTPF50
42    PRINT 1103,EMUO,P,PP,EMU,M                                        EQSTPF51
      STOP 42                                                           EQSTPF52
44    CONTINUE                                                          EQSTPF53
      VSO(M) = VO(M)/(EMU+1.)                                           EQSTPF54
      GO TO 60                                                          EQSTPF55
C          ADJUST -ESO- , -VSO- TO AGREE WITH -DSM-                     EQSTPF56
50    VSO(M) = 1./DSM                                                   EQSTPF57
      EMU = DSM/RHOS(M)-1.                                              EQSTPF58
      ESO(M) = -EMU*(EQSTC(M)+EMU*(EQSTD(M)+EMU*EQSTS(M)))/(EQSTG(M)*   EQSTPF59
     1  RHOS(M)*(1.+EMU))                                               EQSTPF60
60    ELO(M) = ESO(M)+HLM-HSM                                           EQSTPF61
C          COMPUTE -DLM- IF UNSPECIFIED                                 EQSTPF62
      IF(DLM .LE. 0.)DLM = 0.935/VSO(M)                                 EQSTPF63
```

```
      VLO(M) = 1./DLM                                                   EQSTPF64
      TM(M) = TMK/TCK                                                   EQSTPF65
      TB = TBK/TCK                                                      EQSTPF66
      ELB = HVB-HLB                                                     EQSTPF67
C         SOLVE FOR -CL- FROM EQ. 3.21                                  EQSTPF68
      CL = (HLB-HLM)/(TBK-TMK)                                          EQSTPF69
      CV(M) = (HVB-HVM)/(TBK-TMK)                                       EQSTPF70
      DLTC = CV(M)-CL                                                   EQSTPF71
C         SOLVE FOR -A1- , -A2- AND -ALPHA- FROM EQS. 3.24              EQSTPF72
      A1(M) = DLTC/R1*WT(M)                                             EQSTPF73
      A2(M) = (ELB-DLTC*TBK)/(R1*TCK)*WT(M)                             EQSTPF74
C         SOLVE FOR -AR- FROM EQ. 3.25                                  EQSTPF75
      X = 36./TB+6.*TB**6-42.                                           EQSTPF76
      AR = (A2(M)/TB+A1(M)+0.31425*X)/(1.+0.0838*X)                     EQSTPF77
      A2(M) = A2(M)-A1(M)                                               EQSTPF78
C         SOLVE FOR -ZC- FROM EQ. 3.27                                  EQSTPF79
      ZC(M) = 1./(3.72+0.26*(AR-7.))                                    EQSTPF80
C         SOLVE FOR -VC- FROM EQ. 3.33                                  EQSTPF81
      VC(M) = (1.+C1(M)*(1.-TM(M))**(1./3.)+D1(M)*(1.-TM(M)))/OLM       EQSTPF82
C         SOLVE FOR CRITICAL PRESSURE -PC- FROM EQ. 3.34                EQSTPF83
      PC(M) = ZC(M)*R1*TCK/VC(M)/WT(M)                                  EQSTPF84
C         SOLVE EQ. 3.6B FOR B1 = BETA, COMPUTE B,BP                    EQSTPF85
      B1 = 3.                                                           EQSTPF86
      B2 = 1.5*(1./ZC(M)-1.)                                            EQSTPF87
      B3 = 2.25/ZC(M)**2-5.5/ZC(M)-0.75                                 EQSTPF88
70    B0 = B1                                                           EQSTPF89
      B1 = B2+SQRT(B3-1./B1)                                            EQSTPF90
      IF (ABS((B1-B0)/B1) .GT. ACC) GO TO 70                            EQSTPF91
      B(M) = ((3.*B1-6)*B1-1.)/(B1*(3.*B1-1.))                          EQSTPF92
      BP(M) = (B1-3.)/(3.*B1-1.)                                        EQSTPF93
C         COMPUTE -K0- , -K1- , AND -K2- (EQS. 3.7)                     EQSTPF94
      IF(ZK0(M) .EQ. 0.) ZK0(M) = B1                                    EQSTPF95
      ZK1(M) = B1-ZK0(M)                                                EQSTPF96
      ZK2(M) = (1.+ZK1(M)+B1-A1(M)-A2(M))/2.                            EQSTPF97
      EPS1(M) = ZC(M)*TCK*R1/WT(M)                                      EQSTPF98
      EPS2(M) = TCK*(CV(M)-R1/WT(M))                                    EQSTPF99
      E0(M) = HVB-CV(M)*TBK                                             EQSTP100
C         SOLVE EQ. 3.28 FOR RV TO FIND EV0, PV0, DV0, VV0              EQSTP101
      T = TM(M)                                                         EQSTP102
      PV = EXP(A2(M)*(1.-1./T)+A1(M)*ALOG(T))                           EQSTP103
      X1 = T/ZC(M)                                                      EQSTP104
      A = ZK0(M)+ZK1(M)/T                                               EQSTP105
      AP = ZK2(M)*(T-1./T)                                              EQSTP106
C         SOLVE EQ. 4.5 FOR RV                                          EQSTP107
      RV = PV/X1                                                        EQSTP108
      NC3=0                                                             EQSTP109
80    RV1 = RV                                                          EQSTP110
      NC3=NC3+1                                                         EQSTP111
      IF (NC3 .GT. 20) GO TO 82                                         EQSTP112
      X2 = 1.-(B(M)-BP(M)*RV1)*RV1                                      EQSTP113
      P0 = X1*RV1/X2-(A+AP*RV1)*RV1**2                                  EQSTP114
      POP = X1/X2+(X1*RV1*(B(M)-2.*BP(M)*RV1))/(X2*X2)-(2.*A+3.*AP*RV1) EQSTP115
     1*RV1                                                              EQSTP116
      RV=AMAX1(RV1+(PV-P0)/POP,1.E-12)                                  EQSTP117
      IF (ABS(RV-RV1).GT.ACC*RV .AND. ABS(RV-RV1).GT.1.E-12) GO TO 80   EQSTP118
      GO TO 83                                                          EQSTP119
82    PRINT 1104,RV1,P0,POP,RV,M                                        EQSTP120
      STOP 72                                                           EQSTP121
83    CONTINUE                                                          EQSTP122
C         SOLVE EQS. 4.4C, D, AND E FOR EV, RL, EL                      EQSTP123
      EV = E0(M)+EPS2(M)*T-EPS1(M)*((ZK0(M)+2.*ZK1(M)/T)-ZK2(M)*RV/T)*RVEQSTP124
      RL = 1.+C1(M)*(1.-T)**(1./3.)+O1(M)*(1.-T)                        EQSTP125
      EL = EV-EPS1(M)*PV*(1./RV-1./RL)*(A2(M)/T+A1(M)-1.)               EQSTP126
      E1 = E0(M)+EL0(M)-EL                                              EQSTP127
      EV0(M) = EV+E1-E0(M)                                              EQSTP128
      PV0(M) = PV                                                       EQSTP129
      VV0(M) = VC(M)/RV                                                 EQSTP130
      E0(M) = E1                                                        EQSTP131
C         SOLVE EQ. 4.4D FOR -EC- WITH T = 1, RV = 1                    EQSTP132
      EC(M) = E0(M)+EPS2(M)-EPS1(M)*(ZK0(M)+2.*ZK1(M)-ZK2(M))           EQSTP133
```

```
      DEDV(M) = (ELO(M)-ESO(M))/(VLO(M)-VSO(M))
      EOVO(M)=DEDV(M)
      EES(M) = VO(M)*DEDV(M)
      CC(M) = (C1(M)/D1(M))**3/27.
      CSO = ESO(M)/(TMK-298.)
      HDCT(M) = 0.5*(CL-CSO)*TMK
      CBT(M) = 0.5*(CL+CSO)*TMK
      EBL(M) = ELO(M)-CSO*TMK
      EBS(M) = ESO(M)-CL*TMK
      Y1(M) = 2.*CBT(M)
      Y3(M) = Y1(M)*(CL-CSO)*TMK
C     CONSTRUCT A FIT TO APPROXIMATE RV-T RELATION ON LV-V BOUNDRY
      T1=T=0.95
      NPART=5
      GO TO 650
100   R1=RV
      T2=T=0.9
      NPART =6
      GO TO 650
105   R2=RV
      ZN(M)= ALOG((1.-R1)/(1.-R2))/ALOG((1.-T1)/(1.-T2))
      ZM(M) = (1.-R1)/(1.-T1)**ZN(M)
      RETURN
C  ******                                                          ******
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 = I  $  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+SQRT(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 = I
C         SEPARATE SOLID-LIQUID AND LIQUID
      IF (V-VLM) 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)-EPSI(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/(ZC(M)*(I.-(B(M)-BP(M)*RV)*RV))-ZK2(M)*RV**3
      X2 = -ZKO(M)*RV*RV
```

```
      X3 = (ZK2(M)*RV-ZK1(M))*RV*RV                                     EQSTP202
      TM1N = 0.0                                                        EQSTP203
      1F (X1 .GT. 0.0 .AND. X3 .GT. 0.0) TM1N=SQRT(X3/X1)               EQSTP204
      FMAX = (E-ELO(M))/(EVO(M)-ELO(M))                                 EQSTP205
      1F(V .GT. FMAX*VVO(M)+(1.-FMAX)*VLO(M)) GO TO 990                 EQSTP206
      T = 1.0                                                           EQSTP207
      PV = EXP(A2(M)*(1.-1./T)+A1(M)*ALOG(T))                           EQSTP208
      NC4=0                                                             EQSTP209
310   PVT = PV                                                          EQSTP210
      NC4=NC4+1                                                         EQSTP211
      IF (NC4 .GT. 20) GO TO 312                                        EQSTP212
      TA = T                                                            EQSTP213
      PG = X1*T+X2+X3/T                                                 EQSTP214
      PVP = PV*(A2(M)/T+A1(M))/T                                        EQSTP215
      PGP = AMAX1(0.,X1-X3/(T*T))                                       EQSTP216
      T = AMAX1(TA+(PG-PV)/(PVP-PGP),TM1N+ACC)                          EQSTP217
      IF (PVP-PGP .LT.0. ) T=TA+0.05                                    EQSTP218
      T = AM1N1(1.,0.8*TA+0.199)                                        EQSTP219
      PV = EXP(A2(M)*(1.-1./T)+A1(M)*ALOG(T))                           EQSTP220
      IF (ABS((PV-PVT)/PV) .GT. ACC) GO TO 310                          EQSTP221
      EV = EO(M)+EPS2(M)*T-EPS1(M)*(ZKO(M)+2.*ZK1(M)/T-ZK2(M)*RV/T)*RV  EQSTP222
C         BRANCH TO E1THER V OR LV REG1ONS                              EQSTP223
      IF (T .LE. TM(M)) GO TO 985                                       EQSTP224
      IF (E-EV) 850,900,900                                             EQSTP225
312   PR1NT 1105,TA,PG,PVP,PGP,T,PV,M                                   EQSTP226
      STOP 312                                                          EQSTP227
C                                                                       EQSTP228
C  ***          TEST TO SEPARATE L AND LV REG1ONS                       EQSTP229
C         F1RST COMPUTE T ON L-LV L1NE, THEN EL                         EQSTP230
350   NL = 2                                                            EQSTP231
      1F (E .GT. EC(M)) GO TO 800                                       EQSTP232
      RL = VC(M)/V                                                      EQSTP233
      X1 = (1.-RL)/D1(M)/2.                                             EQSTP234
      X = SQRT(X1*X1+CC(M))                                             EQSTP235
      T = 1.-((X-X1)**(1./3.)-(X+X1)**(1./3.))**3                       EQSTP236
C         GO TO 650 TO OBTA1N EL                                        EQSTP237
      NPART = 1                                                         EQSTP238
      GO TO 650                                                         EQSTP239
C         BRANCH TO E1THER L OF LV REG1ONS                              EQSTP240
375   NL = 3                                                            EQSTP241
      1F (E-EL)855,855,800                                              EQSTP242
C  ******                                                        ****** EQSTP243
C               BU1LT-1N SU8ROUT1NES                                    EQSTP244
C  ******                                                        ****** EQSTP245
C                                                                       EQSTP246
C  ***          SOLVE FOR VS ON S-SL L1NE,G1VEN ES-EZ                   EQSTP247
600   TF = (Y2+SQRT(Y2*Y2-Y3(M)))/Y1(M)                                 EQSTP248
602   RGE = RHOS(M)*EQSTG(M)*EZ                                         EQSTP249
      DEN = EQSTC(M)+RGE                                                EQSTP250
      ENUM = EOVO(M)*(TF-1.)-RGE                                        EQSTP251
      EMU1A = 0.                                                        EQSTP252
      EMU1B = EMUJA = ENUM/DEN                                          EQSTP253
      NC1=0                                                             EQSTP254
605   EMUJ8 = ENUM/(DEN+EMU1B*(EQSTD(M)+EMU18*EQSTS(M)))                EQSTP255
      NC1=NC1+1                                                         EQSTP256
      1F(NC1 .GT. 20) GO TO 620                                         EQSTP257
      EMU = (EMU1A*EMUJ8-EMU1B*EMUJA)/(EMUJB-EMUJA+EMUIA-EMUIB)         EQSTP258
      IF ( ABS(EMU-EMUJB) .LE. ACC) GO TO 610                           EQSTP259
      EMU1A = EMU1B                                                     EQSTP260
      EMUJA = EMUJB                                                     EQSTP261
      EMU1B = EMU                                                       EQSTP262
      GO TO 605                                                         EQSTP263
610   VS = 1./(RHOS(M)*(EMU+1.))                                        EQSTP264
      GO TO (220,250,805)NPART                                          EQSTP265
620   PRINT 1106,TF,EZ,M,EMU1A,EMU18                                    EQSTP266
      STOP 620                                                          EQSTP267
C                                                                       EQSTP268
```

```
C ***     SOLVE FOR P,RL,EL,RV,EV, ON LV-V BOUNDRY
C
650   PV=EXP(A2(M)*(1.-1./T)+A1(M)*ALOG(T))
      X1=T/ZC(M)
      A=ZKO(M)+ZK1(M)/T
      AP=ZK2(M)*(T-1./T)
      PX=PV/X1
      8AX=B(M)-A/X1
      IF (PX*BAX .LT. -0.25 .AND. NPART .LT. 5) GO TO 653
      RV=PX*(1.-PX*BAX)
      IF (PX*8AX .LT. -0.05) RV=PV/(X1/(1.+(-B(M)+8P(M)*RV)*RV)-(A+AP*RV
     1)*RV)
      GO TO 654
653   RV=1.-ZM(M)*(1.-T)**ZN(M)
654   NC7=0
655   RV1 = RV
      NC7=NC7+1
      IF (NC7 .GT. 20) GO TO 670
      X2 = 1.-(B(M)-8P(M)*RV)*RV
      PO = X1*RV/X2-(A+AP*RV)*RV**2
      POP = X1/X2+(X1*RV*(B(M)-2.*BP(M)*RV))/X2**2-(2.*A+3.*AP*RV)*RV
      RV = AMAX1(RV+(PV-PO)/POP,1.E-12)
      IF (ABS(RV-RV1).GT.ACC*RV .AND. A8S(RV-RV1).GT.1.E-12) GO TO 655
      EV = EO(M)+EPS2(M)*T-EPS1(M)*((ZKO(M)+2.*ZK1(M)/T)-ZK2(M)*RV/T)*RV
      IF (NPART .GT. 1) RL = 1.+C1(M)*(1.-T)**(1./3.)+D1(M)*(1.-T)
      EL = EV-EPS1(M)*PV*(1./RV-1./RL)*(A2(M)/T+A1(M)-1.)
      GO TO (375,815,875,817,100,105) NPART
670   PRINT1109,RV,RV1,PV,PO,POP,EV,RL,EL,T,M
      STOP 670
C  ******                                                         ******
C                CALCULATIONS FOR EACH PHASE
C  ******                                                         ******
C
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
C  ***           SOLID - LIQUID MIXED PHASE
750   FMAX = (E-ESO(M))/(ELO(M)-ESO(M))
      IF (V .GT. FMAX*VLO(M)+(1.-FMAX)*VSO(M)) GO TO 990
C         FIND T FOR V, E IN SL REGION
755   EPS = E-DEDV(M)*V
      ES = EPS+DEDV(M)*VS
      Y2 = ES-EBS(M)
      TF = (Y2+SQRT(Y2*Y2-Y3(M)))/Y1(M)
      NC5=0
760   TFO = TF
      NC5=NC5+1
      IF(NC5 .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)*ES
      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 -EQSTG(M)*RHOS(M)*ETA*ESP
      TF = TF-H/HP
      ES = EBS(M)+CBT(M)*TF+HDCT(M)/TF
      VS = (ES-EPS)/DEDV(M)
      IF (ABS(TF-TFO)/TF .GT. ACC) GO TO 760
      P = EOVO(M)*(TF-1.)
      GO TO 1000
780   PRINT 1107,TF,TFO,T,M
      STOP 770
```

```
C                                                                       EQSTP337
C  ***         LIQUID PHASE                                             EQSTP338
C         SOLVE FOR PLM, VLM, ON SL-L LINE                              EQSTP339
800   Y2 = E-EBL(M)                                                     EQSTP340
      TF = (Y2+SQRT(Y2*Y2-Y3(M)))/Y1(M)                                 EQSTP341
      EZ = ES = EBS(M)+CBT(M)*TF+HDCT(M)/TF                             EQSTP342
C         GO TO 600 TO GET VS ON S-SL LINE                              EQSTP343
      NPART = 3                                                         EQSTP344
      GO TO 602                                                         EQSTP345
805   VLM= VS+(E-ES)/DEDV(M)                                            EQSTP346
810   PLM = EOVO(M)*(TF-1.)                                             EQSTP347
C         SOLVE FOR PLB, VLB ON L-LV LINE                               EQSTP348
      IF (NL .EQ. 3) GO TO 815                                          EQSTP349
      IF (E .GE. EC(M)) GO TO 820                                       EQSTP350
      IF (NL .EQ. 1) GO TO 812                                          EQSTP351
      RL = VC(M)/V                                                      EQSTP352
      X1 = (1.-RL)/D1(M)/2.                                             EQSTP353
      X = SQRT(X1*X1+CC(M))                                             EQSTP354
      T = 1.-((X-X1)**(1./3.)-(X+X1)**(1./3.))**3                       EQSTP355
C         GO TO 650 TO OBTAIN EL                                        EQSTP356
      NPART = 2                                                         EQSTP357
      GO TO 650                                                         EQSTP358
812   T = TM(M)                                                         EQSTP359
      EL = ELO(M)                                                       EQSTP360
C                                                                       EQSTP361
C      BEGIN ITERATION LOOP TO FIND VLB ON L-LV BOUNDRY, GIVEN E        EQSTP362
C                                                                       EQSTP363
815   TL=T $ ETL=EL $ TU=1.0 $ ETU=EC(M)                                EQSTP364
      TLAST = 0.5*(TU+TL)                                               EQSTP365
C      USE PARABOLIC ESTIMATE OF SLOPES TO OBTAIN T FOR E               EQSTP366
      S2=S23=(TU-TL)/(ETU-ETL)                                          EQSTP367
      IF(ETL .NE. ELO(M))                                               EQSTP368
     1S2 = (TL-TM(M))/(ETL-ELO(M))+S23-(TU-TM(M))/(ETU-ELO(M))          EQSTP369
      T = TL+(S2+(S23-S2)*(E-ETL)/(ETU-ETL))*(E-ETL)                    EQSTP370
      TLAST = 0.5*(TU+TL)                                               EQSTP371
      NC8=0 $ NPART=4                                                   EQSTP372
816   NC8=NC8+1                                                         EQSTP373
      IF(T .GT. TU) T=0.1*TLAST+0.9*TU                                  EQSTP374
      IF(T .LT. TL) T=0.1*TLAST+0.9*TL                                  EQSTP375
      IF(NC8 .GT. 20) GO TO 827                                         EQSTP376
C                                                                       EQSTP377
C      GO TO 650 TO COMPUTE RL,EL,RV,EV FOR GIVEN T                     EQSTP378
C                                                                       EQSTP379
      GO TO 650                                                         EQSTP380
817   IF (ABS(E-EL) .LE. ACC*AMAX1(ABS(E),ELO(M))) GO TO 819            EQSTP381
      S12 = (T-TL)/(EL-ETL)                                             EQSTP382
      S23 = (TU-T)/(ETU-EL)                                             EQSTP383
      S2 = S12+S23-(TU-TL)/(ETU-ETL)                                    EQSTP384
      TLAST =T                                                          EQSTP385
      IF (EL .LT. E) GO TO 818                                          EQSTP386
      T = T+(S2+(S12-S2)*(E-EL)/(ETL-EL))*(E-EL)                        EQSTP387
      ETU=EL $ TU=TLAST $ GO TO 816                                     EQSTP388
818   T = T+(S2+(S23-S2)*(E-EL)/(ETU-EL))*(E-EL)                        EQSTP389
      ETL=EL $ TL=TLAST $ GO TO 816                                     EQSTP390
819   VLB=VC(M)/RL                                                      EQSTP391
      PLB=PC(M)*PV                                                      EQSTP392
      GO TO 825                                                         EQSTP393
C         SOLVE FOR PLB ABOVE CRITICAL POINT ON V = VC LINE             EQSTP394
820   VLB = VC(M)                                                       EQSTP395
      RV = 1.                                                           EQSTP396
      X1 = E-EO(M)+EPS1(M)*ZKO(M)*RV                                    EQSTP397
      X2 = EPS1(M)*(ZK2(M)*RV-2.*ZK1(M))*RV                             EQSTP398
      T = (X1+SQRT(X1*X1-4.*EPS2(M)*X2))/(2.*EPS2(M))                   EQSTP399
      PG = RV*T/(ZC(M)*(1.-(B(M)-BP(M)*RV)*RV))-(ZKO(M)+ZK1(M)/T+ZK2(M)*EQSTP400
     1(T-1./T)*RV)*RV*RV-PVO(M)                                         EQSTP401
      PLB = PC(M)*PG                                                    EQSTP402
```

```
  825 RM = 1./VLM                                                       EQSTP403
      RB = 1./VLB                                                       EQSTP404
      Z1 = (PLM-PLB)/(RM-RB)                                            EQSTP405
      Z2 = (RB*PLM-RM*PLB)/(RM-RB)                                      EQSTP406
      P1 = Z1/V-Z2                                                      EQSTP407
      Z3 = ALOG(PLM/PLB)/ALOG(RM/RB)                                    EQSTP408
      Z4 = (ALOG(RB)*ALOG(PLM)-ALOG(RM)*ALOG(PLB))/ALOG(RM/RB)          EQSTP409
      ALP2 = Z3*ALOG(1./V)-Z4                                           EQSTP410
      F = (PLM/(RM-1./VLO(M))-Z3*PLM/RM)/(Z1-Z3*PLM/RM)                 EQSTP411
      F = AMIN1(1.,AMAX1(0.,F))                                         EQSTP412
      P = EXP(F*ALOG(P1)+(1.-F)*ALP2)                                   EQSTP413
      GO TO 1000                                                        EQSTP414
  827 PRINT 1110,T,TMIN,TMAX,TU,TL,E,ET,ETL,ETU                         EQSTP415
      STOP 727                                                          EQSTP416
C                                                                       EQSTP417
C  ***           LIQUID-VAPOR MIXED PHASE                               EQSTP418
  850 RL = 1.+C1(M)*(1.-T)**(1./3.)+D1(M)*(1.-T)                        EQSTP419
      EL = EV-EPS1(M)*PV*(1./RV-1./RL)*(A2(M)/T+A1(M)-1.)               EQSTP420
C         CONSTRUCT UPPER AND LOWER BOUNDS ON E, T                      EQSTP421
C         BEGIN ITERATION LOOP FOR E WITH T AS A PARAMETER              EQSTP422
      ETU = EV                                                          EQSTP423
      GO TO 860                                                         EQSTP424
C         ENTER FROM 375 FOR V  LESS THAN VC                            EQSTP425
  855 ETU = EL                                                          EQSTP426
  860 ETL = (V-VLO(M))/(VVO(M)-VLO(M))*(EVO(M)-ELO(M))+ELO(M)           EQSTP427
      FMAX=(E-ELO(M))/(EVO(M)-ELO(M))                                   EQSTP428
      IF(V .GT. FMAX*VVO(M)+(1.-FMAX)*VLO(M)) GO TO 990                 EQSTP429
      TU = T   $   TL = TM(M)                                           EQSTP430
      TLAST=0.5*(TU+TL)                                                 EQSTP431
C         LINEAR INTERPOLATION TO ESTIMATE T                            EQSTP432
      NC6=0                                                             EQSTP433
      NPART=3                                                           EQSTP434
      T = TL+(E-ETL)*(TU-TL)/(ETU-ETL)                                  EQSTP435
  870 NC6=NC6+1                                                         EQSTP436
      IF(NC6 .GT. 20) GO TO 892                                         EQSTP437
      IF (T .GT. TU) T=0.1*TLAST+0.8999*TU                              EQSTP438
      IF(T .LT. TL) T=0.1*TLAST+0.8999*TL                               EQSTP439
C         GO TO 650 TO COMPUTE RL, EL, RV, EV FOR GIVEN T               EQSTP440
      GO TO 650                                                         EQSTP441
  875 ET = (RL*V-1.)/(RL/RV-1.)*(EV-EL)+EL                              EQSTP442
      IF(ABS(E-ET) .LE. ACC*AMAX1(ABS(E),ELO(M))) GO TO 890             EQSTP443
      TLAST=T                                                           EQSTP444
      IF(ABS(ET-ETL) .GT.1.) S12=(T-TL)/(ET-ETL)                        EQSTP445
      IF(ABS(ETU-ET) .GT. 1.)S23=(TU-T)/(ETU-ET)                        EQSTP446
      S2=S12+S23-(TU-TL)/(ETU-ETL)                                      EQSTP447
      IF(ET .LT. E) GO TO 880                                           EQSTP448
      T=T+(S2+(S12-S2)*(E-ET)/(ETL-ET))*(E-ET)                          EQSTP449
      ETU=ET $ TU=TLAST $ GO TO 870                                     EQSTP450
  880 T=T+(S2+(S23-S2)*(E-ET)/(ETU-ET))*(E-ET)                          EQSTP451
      ETL=ET $ TL=TLAST $ GO TO 870                                     EQSTP452
  890 P = PC(M)*(PV-PVO(M))                                             EQSTP453
      GO TO 1000                                                        EQSTP454
  892 PRINT 1108,T,TMIN,TMAX,TU,TL,E,ET,ETL,ETU                         EQSTP455
      STOP 772                                                          EQSTP456
C                                                                       EQSTP457
C  ***           VAPOR PHASE                                            EQSTP458
  900 RV = VC(M)/V                                                      EQSTP459
      X1 = E-EO(M)+EPS1(M)*ZKO(M)*RV                                    EQSTP460
      X2 = EPS1(M)*(ZK2(M)*RV-2.*ZK1(M))*RV                             EQSTP461
      T = (X1+SQRT(X1*X1-4.*EPS2(M)*X2))/(2.*EPS2(M))                   EQSTP462
      P = PC(M)*(RV*T/(ZC(M)*(1.-(B(M)-BP(M)*RV)*RV))-(ZKO(M)+ZK1(M)/T+ EQSTP463
     1  ZK2(M)*(T-1./T)*RV)*RV*RV-PVO(M))                               EQSTP464
      GO TO 1000                                                        EQSTP465
  985 CONTINUE                                                          EQSTP466
```

SUBROUTINE EQSTPF (Concluded)

```
C                                                                       EQSTP467
C  ***          CUTOFF AT ZERO PRESSURE                                 EQSTP468
990    P = 0.                                                           EQSTP469
1000   RETURN                                                           EQSTP470
1100   FORMAT(8A10)                                                     EQSTP471
 1101  FORMAT(A10,1P7E10.3)                                             EQSTP472
1102   FORMAT(1H+,79X,5H IND=A2,5H, IN=I2,* READ IN EQSTPF*)            EQSTP473
 1103  FORMAT(1H-,* LOC=42 IN EQSTPF*5X,* EMUO,P,PP,EMU,M= *1P5E10.3///) EQSTP474
 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,PVP,PGP,T,PV,M= *1P3E10.EQSTP476
      13/1P4E10.3///)                                                   EQSTP477
 1106  FORMAT(1H-,* LOC=620 IN EQSTPF *5X,* T,EZ,M,EMUIA,EMUIB=*1P5E10.3/EQSTP478
      1//)                                                              EQSTP479
 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,EV,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                                                             EQSTP487
```

SUBROUTINE ESA

```
      SUBROUTINE ESA(NCALL,IN,M,C,D,E,P,OPDR,DPDE)
C
C     ROUTINE COMPUTES PRESSURE FROM SIMPLE TWO-PHASE EQUATION OF STATE.
C     ESA HAS TWO PARTS, CORRESPONDING TO REAOING AND COMPUTING
C
C     READ INPUT (NCALL=0). CALL IS FROM GENRAT.
C       INPUT  - NCALL, IN, M, MATERIAL PROPERTY CARDS
C       OUTPUT - PRINTS CARO IMAGES, ORGANIZES OATA INTO ARRAYS
C
C     COMPUTE PRESSURE (NCALL=1)   CALL IS FROM HSTRESS USUALLY.
C       INPUT  - NCALL,M,C,D,E
C       OUTPUT - C,P,OPDE
C
C               NAMED COMMON
      REAL MU,MUM
      COMMON /EQS/  EQSTA(6),EQSTC(6),EQSTD(6),EQSTE(6),EQSTG(6),
     1  EQSTH(6),EQSTN(6),EQSTS(6),EQSTV(6),CZQ(6),CWQ(6),C2(6)
      COMMON /MELT/ EMELT(6,5),SPH(6),THERM(6,8)
      COMMON /RHO/ RHO(6),RHOS(6)
      COMMON /TSR/ TSR(6,30),EXMAT(6,20),TENS(6,3)
      COMMON /Y/ Y0(6),YADD(6),MU(6),MUM,YADDM
C
      OIMENSION B(4,6),F1(6),F2(6),F3(6),F4(6),G1(6),G2(6),G3(6)
      DATA IOO/1H /
C
      IF (NCALL .EQ. 1) GO TO 200
C ********                                                 ********
C               READ INPUT OATA AND INITIALIZE ARRAYS
C ********                                                 ********
      READ (IN,1100) A1,G1(M),F1(M),F2(M),P1,R1,E1
      WRITE (6,1100) A1,G1(M),F1(M),F2(M),P1,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
C         INITIALIZE COEFFICIENTS IN EXPANSION EQUATION
      R0=RHOS(M)
      F3(M)=(2.*F1(M)-F2(M))/R0
      F4(M)=(F2(M)-F1(M))/R0/R0
      G2(M)=EQSTG(M)-G1(M)          $    G3(M)=G1(M)/R0
C         INITIALIZE -B- ARRAY
      A0=EQSTC(M)/R0
      A1=P1-R1*E1*(G2(M)+R1*G3(M))-R1*E1*E1*(F3(M)+R1*F4(M))
      A2=P2-R2*E2*(G2(M)+R2*G3(M))-R2*E2*E2*(F3(M)+R2*F4(M))
      A3=P3-R3*E3*(G2(M)+R3*G3(M))-R3*E3*E3*(F3(M)+R3*F4(M))
C     REDEFINE A TO INCLUDE DENOMINATORS
      R0=RHOS(M)
      D01=R0-R1   $   D02=R0-R2   $   D03=R0-R3   $   D12=R1-R2
      D13=R1-R3   $   D23=R2-R3
      A0=A0/(D01*D02*D03)          $    A1= A1/(D01*D01*D12*D13)
      A2=-A2/(002*D02*D12*023)     $    A3= A3/(D03*D03*013*D23)
      B(1,M)=-A0*R1*R2*R3-R0*A1*R2*R3-R0*R1*A2*R3-R0*R1*R2*A3
      B(2,M)=R0*R1*(A2+A3)+R0*R2*(A1+A3)+R0*R3*(A1+A2)
     1   +R1*R2*(A0+A3)+R1*R3*(A0+A2)+R2*R3*(A0+A1)
      B(3,M)=-R0*(A1+A2+A3)-R1*(A0+A2+A3)-R2*(A0+A1+A3)-R3*(A0+A1+A2)
      B(4,M)=A0+A1+A2+A3
      RETURN
C ********                                                 ********
C               CALCULATION OF PRESSURE AND SOUNO SPEED
C ********                                                 ********
200   IF (D .LT. RHOS(M)) GO TO 300
C
C     ***  COMPRESSION EQUATION OF STATE
      U=(D-RHOS(M))/RHOS(M)
      PH=U*(EQSTC(M)+U*(EQSTD(M)+U*EQSTS(M)))
      GG1=EQSTG(M)+U*G1(M)
      GF=1.-0.5*U*GG1
      FF=F1(M)+U*F2(M)
```

SUBROUTINE ESA (Concluded)

```
      P = PH*GF + (GG1*D + FF*E)*E
      DPDR =((EQSTC(M)+U*(2.*EQSTD(M)+U*3.*EQSTS(M)))*GF
     1  -PH*(0.5*EQSTG(M)+U*G1(M))  +(G1(M)*D + F2(M)*E)*E)/RHOS(M)
     2  +GG1*E
      DPDE = GG1*D + 2.*FF*E
      GO TO 350
C
C    ***  EXPANSION EQUATION OF STATE
300   GG3=D*(G2(M)+D*G3(M))
      FF =D*(F3(M)+D*F4(M))
      BTERMS=B(1,M)+D*(B(2,M)+D*(B(3,M)+D*B(4,M)))
      P = (D-RHOS(M))*BTERMS + (GG3 + FF*E)*E
      DPDR = (G2(M)+2.*D*G3(M)  +  (F3(M)+ 2.*D*F4(M))*E)*E
     1  +BTERMS  +  (D-RHOS(M))*(B(2,M)+D*(2.*B(3,M)+D*3.*B(4,M)))
      DPDE = GG3 + 2.*FF*E
350   CSQ = DPDR + P*DPDE/D**2
      IF (CSQ .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

```
      SUBROUTINE EXPLODE(NCALL,IN,M,EHL,DHL,DOLD,PHL,SHL,FBUR,X,J,QH,   EXPLODE2
     1  TIME,DTNH)                                                      EXPLODE3
C                                                                       EXPLODE4
C        THIS SUBROUTINE FOR DETONATING FLOW HAS THREE FUNCTIONS AND    EXPLODE5
C        IS DIVIDED INTO THREE CORRESPONDING PARTS.                     EXPLODE6
C          1.  INITIALIZE THE MATERIAL VARIABLES AT THE TIME OF READING EXPLODE7
C              MATERIAL PROPERTIES.                                     EXPLODE8
C          2.  INITIALIZE THE COORDINATE ARRAYS TO SIMULATE INITIATION. EXPLODE9
C          3.  COMPUTE PROGRESS OF DETONATION DURING THE CALCULATION.   EXPLOD10
C                                                                       EXPLOD11
C              NAMED COMMON                                             EQSTCOM2
      REAL MU,MUM                                                       EQSTCOM3
      COMMON /EQS/  EQSTA(6),EQSTC(6),EQSTD(6),EQSTE(6),EQSTG(6),       EQSTCOM4
     1  EQSTH(6),EQSTN(6),EQSTS(6),EQSTV(6),CZQ(6),CWQ(6),C2(6)         EQSTCOM5
      COMMON /MELT/ EMELT(6,5),SPH(6),THERM(6,8)                        EQSTCOM6
      COMMON /RHO/ RHO(6),RHOS(6)                                       EQSTCOM7
      COMMON /TSR/ TSR(6,30),EXMAT(6,20),TENS(6,3)                      EQSTCOM8
      COMMON /Y/ Y0(6),YADD(6),MU(6),MUM,YADDM                          EQSTCOM9
C                                                                       EXPLOD13
      DIMENSION BURN(6),DET(6),DIST(6),ECJ(6),PCJ(6),QEXPL(6),VCJ(6)    EXPLOD14
      DIMENSION EHL(1),DHL(1),PHL(1),SHL(1),FBUR(1),X(1)                EXPLOD15
C                                                                       EXPLOD16
      IF (NCALL-2) 100,200,300                                          EXPLOD17
C                                                                       EXPLOD18
C           INITIALIZE MATERIAL VARIABLES                               EXPLOD19
  100 READ(IN,1000)A1,QEXPL(M),BURN(M),DIST(M)                          EXPLOD20
      PRINT 1010,A1,QEXPL(M),BURN(M),DIST(M)                            EXPLOD21
      PRINT 1001, IN                                                    EXPLOD22
      DET(M)=SQRT(2.*QEXPL(M)*EQSTG(M)*(EQSTG(M)+2.))                   EXPLOD23
      EHL(1)=DET(M)                                                     EXPLOD24
      VCJ(M)=(EQSTG(M)+1.)/((EQSTG(M)+2.)*RHO(M))                       EXPLOD25
      ECJ(M)=2.*(EQSTG(M)+1.)*QEXPL(M)/(EQSTG(M)+2.)                    EXPLOD26
      PCJ(M)=2.*RHO(M)*QEXPL(M)*EQSTG(M)                                EXPLOD27
      IF (DIST(M) .EQ.0.) PRINT 1102,QEXPL(M)                           EXPLOD28
      PRINT 1100,DET(M),VCJ(M),ECJ(M),PCJ(M)                            EXPLOD29
 1130 FORMAT(*  AMUR, H1, VOCN =*1P3E10.3)                              EXPLOD30
      RETURN                                                            EXPLOD31
C                                                                       EXPLOD32
C           INITIALIZE CELL VARIABLES                                   EXPLOD33
  200 CONTINUE                                                          EXPLOD34
      IF (DIST(M) .EQ. 0.) GO TO 270                                    EXPLOD35
      DX = X(J+1)-X(J)                                                  EXPLOD36
      IF (DX .LE. 0.) GO TO 250                                         EXPLOD37
      XH=0.5*(X(J)+X(J+1))                                              EXPLOD38
      TBURN = (ABS(XH-BURN(M))-DIST(M)*DX)/DET(M)                       EXPLOD39
      IF (TBURN .GE. 0.) GO TO 250                                      EXPLOD40
      FBURN = AMIN1(1.,-TBURN*DET(M)/(DIST(M)*DX))                      EXPLOD41
      EHL(J)=QEXPL(M)+(ECJ(M)-QEXPL(M))*FBURN                           EXPLOD42
      DHL(J)=RHO(M)/(1.-FBURN*(1.-VCJ(M)*RHO(M)))                       EXPLOD43
      PHL(J)=SHL(J)=PCJ(M)*FBURN                                        EXPLOD44
      FBUR(J)=FBURN                                                     EXPLOD45
  250 IF (FBUR(J) .NE. 0.) PRINT 1300,J,M,EHL(J),DHL(J),PHL(J),FBUR(J)  EXPLOD46
      RETURN                                                            EXPLOD47
  270 EHL(J) = QEXPL(M)                                                 EXPLOD48
      FBUR(J) = 1.0                                                     EXPLOD49
      RETURN                                                            EXPLOD50
C                                                                       EXPLOD51
C           COMPUTE DETONATION PROCESS.                                 EXPLOD52
  300 CONTINUE                                                          EXPLOD53
      DX=X(J+1)-X(J)                                                    EXPLOD54
      XH=0.5*(X(J)+X(J+1))                                              EXPLOD55
      DH=DHL(J)                                                         EXPLOD56
      TBURN = (ABS(XH-BURN(M))-DIST(M)*DX/2.)/DET(M)                    EXPLOD57
      FBURN = AMIN1(1.,AMAX1((TIME-0.5*DTNH-TBURN)*DET(M)/(DIST(M)*DX), EXPLOD58
     1  (1.-RHO(M)/DH)/(1.-VCJ(M)*RHO(M)),FBUR(J)))                     EXPLOD59
      IF (FBURN .LT. 1.E-3) RETURN                                      EXPLOD60
      HDV = 0.5*(1./DOLD-1./DH)                                         EXPLOD61
      POLD=PHL(J)                                                       EXPLOD62
```

SUBROUTINE EXPLODE (Concluded)

```
      PHL(J)=EQSTG(M)*DH*(EHL(J)+POLD*HDV+QEXPL(M)*(FBURN-FBUR(J))+QH*2.EXPLOD63
     1   *HDV)/(1.-EQSTG(M)*HDV*DH)                                     EXPLOD64
      EHL(J)=EHL(J)+(PHL(J)+POLD)*HDV+QEXPL(M)*(FBURN-FBUR(J))+2.*QH*HDVEXPLOD65
      PHL(J)=AMAX1(PHL(J),PCJ(M)*FBURN)                                 EXPLOD66
      EHL(J)=AMAX1(EHL(J),ECJ(M)*FBURN)                                 EXPLOD67
      FBUR(J)=FBURN                                                     EXPLOD68
      IF (FBURN .EQ. 1.) PRINT 1400,J,DH                                EXPLOD69
 1400 FORMAT(* DETONATION COMPLETED FOR J=*I5,*  WITH DENSITY =*1PE12.4)EXPLOD70
      RETURN                                                            EXPLOD71
C                                                                       EXPLOD72
 1000 FORMAT(A10,7E10.3)                                                EXPLOD73
 1010 FORMAT(A10,1P7E10.3)                                              EXPLOD74
1001  FORMAT(1H+,79X,* IND=   , IN=*I2,* -EXPLOOE-*,*,ERG/G,CM,1/CM*)   EXPLOD75
 1100 FORMAT(*    OUTPUT OF EXPLODE, DET=*1PE10.3,*, VCJ=*1PE10.3,*, ECJEXPLOD76
     1=*1PE10.3,*, PCJ=*1PE10.3)                                        EXPLOD77
 1102 FORMAT(10X,*EXPLODE-CONST.VOL.EXPLOSION WITH ENERGY=*1PE10.3,* ERGEXPLOD78
     1/G*)                                                              EXPLOD79
 1300 FORMAT(* EXPLODE, J=*I3,* M=*I3,* E=*1PE10.3,* D=*F10.6,* P=*     EXPLOD80
     1  1PE10.3,* F=*F6.3)                                              EXPLOD81
      END                                                               EXPLOD82
```

SUBROUTINE EXTRA

```
      SUBROUTINE EXTRA                                                  EXTRA  2
C                                                                       EXTRA  3
C        ROUTINE IS CALLEO TO READ IN AUXILIARY INFORMATION FROM CARDS EXTRA  4
C     INPUT - NONE                                                      EXTRA  5
C     OUTPUT - ANY WORDS IN COMMON WHICH ARE READ FROM THE EXTRA CARDS  EXTRA  6
C                                                                       EXTRA  7
      INTEGER H,POROUS,PRESS,RINTER,SOLID,SPALL                         PUFCOM 2
      REAL MATL,NEM,NET,NEMH,NETH                                       PUFCOM 3
C              MISCELLANEOUS                                            PUFCOM 4
      COMMON AZERO(1),CEF,CKS,DAVG,OELTIM,OISCPT(10),OOLO,DRHO,DTMAX,   PUFCOM 5
     1 DTMIN,DTN,OTNH,OU,DX,EOLD,F,FAC,FIRST,J,JCYCS,JINIT,             PUFCOM 6
     2 JFIN,JREZON(15),JSMAX,JSTAR,JTS,LSUB(30),M,MAXPR(30),N,NCYCS,    PUFCOM 7
     3 NEDIT,NPERN,NR,NREZON,NSCRB(6),NSEPRAT,NSPALL,NTEDT,             PUFCOM 8
     4 NTEX,NTR(15),POLD,P6(20),R(30),RLAST,SLAST,SMAX,TEDIT(50),       PUFCOM 9
     5 TF,TIME,TJ,TREZON,TS,T6(20),ULAST,UOLO,UZERO,XLAST,XNOW,XOLD     PUFCOM10
     1 ,XJDIT(20)                                                       PUFCOM11
C              HALFSTEP VALUES                                          PUFCOM12
      COMMON OH,DHLAST,OUH,EH,PH,RH,RHLAST,SH,SHLAST,UH,UHLAST,XH,XHLASTPUFCOM13
     1 ,NEMH,NETH                                                       PUFCOM14
C              CONDITION INOICATORS                                     PUFCOM15
      COMMON INF,LINTER,MIRROR,NORMAL,POROUS,PRESS,RINTER,SOLID,SPALL   PUFCOM16
C              CELL LAYOUT                                              PUFCOM17
      COMMON DXX(30),JBND(30),JMAT(30),NAUTO,MATL(6,2),NLAYER,NMTRLS,   PUFCOM18
     1 THK(30)                                                          PUFCOM19
C                                                                       PUFCOM20
C              COORDINATE ARRAYS                                        COORDCO2
      COMMON/COORD/X(200),X0(200),CHL(200),OHL(200),DPOO(200),OPDE(200),COOROCO3
     1 EHL(200),H(200,3),NEM(200),NET(200),PHL(200),RHL(200),SOT(200),  COORDCO4
     2 SHL(200),T(200),U(200),YHL(200),ZHL(200)                         COOROCO5
      COMMON/NSC/A(5000)                                                NSCCOM 2
C              NAMEO COMMON                                             EQSTCOM2
      REAL MU,MUM                                                       EQSTCOM3
      COMMON /EQS/  EQSTA(6),EQSTC(6),EQSTD(6),EQSTE(6),EQSTG(6),       EQSTCOM4
     1 EQSTH(6),EQSTN(6),EQSTS(6),EQSTV(6),CZQ(6),CWQ(6),C2(6)          EQSTCOM5
      COMMON /MELT/ EMELT(6,5),SPH(6),THERM(6,8)                        EQSTCOM6
      COMMON /RHO/ RHO(6),RHOS(6)                                       EQSTCOM7
      COMMON /TSR/ TSR(6,30),EXMAT(6,20),TENS(6,3)                      EQSTCOM8
      COMMON /Y/ Y0(6),YADD(6),MU(6),MUM,YAODM                          EQSTCOM9
      COMMON /INO/ IEOS(6),INDK(20),NALPHA,NCMP(6),NFR(6),NPOR(6),      INDCOM 2
     1 NDS(6),NPR(6),NCON(6),NVAR(6)                                    INDCOM 3
      COMMON /RAD/ SSTOP(5),START(5),SDURM,SSTOPM,NSPEC,SSJ,JSS,IPLOT(4)RAOCOM 2
     1 ,XMAX(4),XMIN(4),YMAX(4),YMIN(4),IA(7),ITITLE(24),NARZ,TARZ      RAOCOM 3
C                                                                       EXTRA 14
      NAMELIST/NLIST/ DTMAX,JCYCS,JINIT,JFIN,LSUB,MAXPR,NEDIT,NSPALL,   EXTRA 15
     1 P6,TREZON,TS,T6,UZERO,                                           EXTRA 16
     2 X,CHL,DHL,DPDD,DPOE,EHL,H,NEM,NET,PHL,RHL,SHL,T,U,YHL,ZHL,       EXTRA 17
     3 JBND,JMAT,NLAYER,                                                EXTRA 18
     4 EQSTA,EQSTC,EQSTD,EQSTE,EQSTG,EQSTH,EQSTS,EQSTV,CZQ,CWQ,C2,      EXTRA 19
     5 EMELT,  RHO,RHOS,  TSR,EXMAT,TENS,  Y0,YADD,MU,                  EXTRA 20
     6 NCMP,NFR,NPOR,NOS,NPR,NCON,                                      EXTRA 21
     7 SSTOP,START,SOURM                                                EXTRA 22
     8 ,EQSTN                                                           EXTRA 23
C                                                                       EXTRA 24
C                                                                       EXTRA 25
      IN=5  $  JO=6                                                     EXTRA 26
      REWIND 7                                                          EXTRA 27
      NREC=0                                                            EXTRA 28
 10   REAO (IN,902) (A(I),I=1,9)                                        EXTRA 29
      IF (EOF(IN)) 19,I5                                                EXTRA 30
 15   IF (A(I) .EQ. 2H S) GO TO 18                                      EXTRA 31
      WRITE (JO,902) (A(I),I=1,9)                                       EXTRA 32
      WRITE (JO,901)                                                    EXTRA 33
      WRITE (7,902) (A(I),I=I,9)                                        EXTRA 34
      IF (A(1) .EQ. 2H $) NREC=NREC+1                                   EXTRA 35
      GO TO 10                                                          EXTRA 36
 18   WRITE (JO,902)(A(I),I=1,9)                                        EXTRA 37
      GO TO 20                                                          EXTRA 38
```

SUBROUTINE EXTRA (Concluded)

```
 19   WRITE (JO,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           4       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)
      IF (LS.GT. 1) GO TO 200
      IF (LS .GE. 0) GO TO 30
      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.
30    IF (MSAVE .EQ. M) GO TO 150
C     INITIALIZE IN REGION 1
 50   EN = X(1)
      IF (X(1) .GT. 0.  .AND. (X(2) .NE. 0. .OR. X(4) .NE. 0.))
     1  GO TO 60
      E(M,1+LS*3)=X(1)
      NREG(M,LS+1)=-1
      RETURN
 60   NIN = 9 *LS
      IF (X(4) .NE. 0.)  GO TO 100
      NR = 1
      F(M,1+NIN) = 1.
      F(M,2+NIN)=(-1.+4.*X(2))/X(1)
      F(M,3+NIN)=-4.*X(2)/X(1)**2
      E(M,1+3*LS) = X(1)
      NREG(M,LS+1)=1
      IF (ABS(X(2)) .GT. 0.251) GO TO 500
      RETURN
 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
C     INITIALIZE IN REGION 2
      F(M,1+NIN) =X(4)-X(2)/(X(1)-X(2))*(-X(4)+4.*X(5)*X(1)/
     1  (X(1)- X(2)))
      F(M,2+NIN)=(-X(4)+4.*X(5)*(X(2)+X(1))/(X(1)-X(2)))/(X(1)-X(2))
      F(M,3+NIN)=-4.*X(5)/(X(1)-X(2))**2
      E(M,2+LS*3) = X(1)
      NREG(M,LS+1) = 2
      IF (ABS(X(5)) .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
 120  NR=2
      IF (X(5) .LT. 1.)  X(5) = X(5) *X(1)
      F(M,1+NIN)= X(4)-X(2)/(X(5)-X(2))*(X(7)-X(4) +4.*X(6)/(X(5)-X(2))
     1 *X(5))
      F(M,2+NIN)= (X(7)-X(4)+4.*X(6)*(X(5)+X(2))/(X(5)-X(2)))/
     1 (X(5)-X(2))
      F(M,3+NIN)=-4.*X(6)/(X(5)-X(2))**2
      E(M,2+LS*3)=X(5)
      E(M,3+LS*3) = X(1)
      MSAVE = M
      X7 = X(7)
      X5 = X(5)
      IF (ABS(X(6)) .GT. 0.25 *(X(4)-X(7))+1.E-4) GO TO 500
```

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))/(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     ********************
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 .AND. EN .LT. 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 - 3 ) 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 NN=NREG(M,2)
      IF (NN .LE. 0 .AND. 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 = F(M,1+NIN)+(F(M,2+NIN) +F(M,3+NIN) * EN) *EN
  400 RETURN
  500 PRINT 1500, NR
 1500 FORMAT (33HOERROR IN FMELT, SLOPE IN REGION I3,
     1 51H IS POSITIVE BECAUSE CURVE OFFSET EXCEEDS (F1-F2)/4)
      RETURN
      END
```

SUBROUTINE GENRAT

```
      SUBROUTINE GENRAT                                                 GENRAT     2
C                                                                       GENRAT     3
C     READS INPUT DATA AND INITIALIZES ARRAYS                           GENRAT     4
C        *  READS INPUT CARDS, EXCEPT FOR RADIATION INFORMATION         GENRAT     5
C        *  COMPUTES COORDINATE LAYOUT                                  GENRAT     6
C        *  INITIALIZES DENSITY, ENERGY, YIELD, SOUND SPEED, SPALL      GENRAT     7
C             STRENGTH, CONDITION INDICATORS, PARTICLE VELOCIY          GENRAT     8
C        *  PRINTS INITIAL LAYOUT FOR NON-RADIATION PROBLEMS            GENRAT     9
C                                                                       GENRAT    10
      INTEGER H,POROUS,PRESS,RINTER,SOLID,SPALL                         PUFCOM     2
      REAL MATL,NEM,NET,NEMH,NETH                                       PUFCOM     3
C              MISCELLANEOUS                                            PUFCOM     4
      COMMON AZERO(1),CEF,CKS,DAVG,DELTIM,DISCPT(10),DOLD,DRHO,DTMAX,   PUFCOM     5
     1  DTMIN,DTN,DTNH,DU,DX,EOLD,F,FAC,FIRST,J,JCYCS,JINIT,            PUFCOM     6
     2  JFIN,JREZON(15),JSMAX,JSTAR,JTS,LSUB(30),M,MAXPR(30),N,NCYCS,   PUFCOM     7
     3  NEDIT,NPERN,NR,NREZON,NSCRB(6),NSEPRAT,NSPALL,NTEDT,            PUFCOM     8
     4  NTEX,NTR(15),POLD,P6(20),R(30),RLAST,SLAST,SMAX,TEDIT(50),      PUFCOM     9
     5  TF,TIME,TJ,TREZON,TS,T6(20),ULAST,UOLD,UZERO,XLAST,XNOW,XOLD    PUFCOM    10
     1  ,XJDIT(20),MS                                                   PUFCOM    11
C              HALFSTEP VALUES                                          PUFCOM    12
      COMMON DH,DHLAST,DUH,EH,PH,RH,RHLAST,SH,SHLAST,UH,UHLAST,XH,XHLAST PUFCOM    13
     1  ,NEMH,NETH                                                      PUFCOM    14
C              CONDITION INDICATORS                                     PUFCOM    15
      COMMON INF,LINTER,MIRROR,NORMAL,POROUS,PRESS,RINTER,SOLID,SPALL   PUFCOM    16
C              CELL LAYOUT                                              PUFCOM    17
      COMMON DXX(30),JBND(30),JMAT(30),NAUTO,MATL(6,2),NLAYER,NMTRLS,   PUFCOM    18
     1  THK(30)                                                         PUFCOM    19
C                                                                       PUFCOM    20
C              NAMED COMMON                                             EQSTCOM    2
      REAL MU,MUM                                                       EQSTCOM    3
      COMMON /EQS/  EQSTA(6),EQSTC(6),EQSTD(6),EQSTE(6),EQSTG(6),       EQSTCOM    4
     1  EQSTH(6),EQSTN(6),EQSTS(6),EQSTV(6),CZQ(6),CWQ(6),C2(6)         EQSTCOM    5
      COMMON /MELT/ EMELT(6,8),GMELT(6,8),SPH(6),THERM(6,8)             EQSTCOM    6
      COMMON /RHO/ RHO(6),RHOS(6)                                       EQSTCOM    7
      COMMON /TSR/ TSR(6,30),EXMAT(6,20),TENS(6,3)                      EQSTCOM    8
      COMMON /Y/ YO(6),YADD(6),MU(6),MUM,YADDM                          EQSTCOM    9
C              COORDINATE ARRAYS                                        COORDCOM   2
      COMMON/COORD/X(200),X0(200),CHL(200),DHL(200),DPDD(200),DPDE(200), COORDCOM  3
     1  EHL(200),H(200,3),NEM(200),NET(200),PHL(200),RHL(200),SDT(200), COORDCOM   4
     2  SHL(200),T(200),U(200),YHL(200),ZHL(200)                        COORDCOM   5
      COMMON/NSC/A(5000)                                                NSCCOM     2
      COMMON /JED/JEDIT(100),JNUM(100),JTYP(100),NAME2(40),JEDSIZ,      JEDCOM     2
     1  MODLUS,NERR,NJEDIT,NTAPE                                        JEDCOM     3
      COMMON /IND/ IEOS(6),INDK(20),NALPHA,NCMP(6),NFR(6),NPOR(6),      INDCOM     2
     1  NDS(6),NPR(6),NCON(6),NVAR(6)                                   INDCOM     3
      COMMON /RAD/ SSTOP(9),START(9),SDURM,SSTOPM,NSPEC,SSJ,JSS,IPLOT(4) RADCOM    2
     1  ,XMAX(4),XMIN(4),YMAX(4),YMIN(4),IA(7),ITITLE(24),NARZ,TARZ     RADCOM     3
      COMMON /PES/ LVMAX,LVTOT,LVAR(200),COM(4000)                      GENRAT    18
C                                                                       GENRAT    19
      COMMON /ESC/ ESC(6,20)                                            GENRAT    20
      DIMENSION DELFIN(30,5),DELX(30,5),TH(30,5),NCELLS(30,5),NZONES(30) GENRAT   21
      INTEGER HH                                                        GENRAT    22
C                                                                       GENRAT    23
      EQUIVALENCE (DELFIN,H(1)),(DELX,H(151)),(TH,H(301)),              GENRAT    24
     1  (NCELLS,H(451)),(NZONES,H(601))                                 GENRAT    25
C                                                                       GENRAT    26
C                                                                       GENRAT    27
16    FORMAT (1H1,10A10//                                          131H GENRAT    28
     1   J     DX        X(J)       U(J)       YHL(J)     CHL(J)     DHL(J) GENRAT  29
     2 T(J)      ZHL(J)      EHL(J)      MATERIAL    COND     J        , GENRAT    30
     3 /102H          CM         CM        CM/SEC     DYN/CM2    CM/SEC     GM/C GENRAT 31
     4M3     DYN/CM2    GM/CM2     ERG/GM                             )  GENRAT    32
 17   FORMAT(I4,1P9E10.3,2X,A9,3R2,I5)                                  GENRAT    33
18    FORMAT(29H TIME TO COMPLETE  GENRAT IS F10.3,9H SECONDS.)         GENRAT    34
 19   FORMAT(A4,A5,1X,1P7E10.3)                                         GENRAT    35
 1019 FORMAT(A4,A5,1X,7E10.3)                                           GENRAT    36
 1020 FORMAT(A10,7E10.3)                                                GENRAT    37
 1021 FORMAT(2(A10,E10.3),A10,I10,A10,E10.3)                            GENRAT    38
 1025 FORMAT(2(A10,I10),2(A10,E10.3))                                   GENRAT    39
 20   FORMAT(A10,1P7E10.3,A5,A2,A5,I2,3A10,A7)                          GENRAT    40
 21   FORMAT(2(A10,1PE10.3),A10,I10,A10,1PE10.3,A5,A2,A5,I2,3A10,A7)    GENRAT    41
22    FORMAT(A10,14I5,A5,A2,A5,I2,3A10,A7)                              GENRAT    42
23    FORMAT(4(A10,I10),A5,A2,A5,I2,3A10,A7)                            GENRAT    43
24    FORMAT (10A1)                                                     GENRAT    44
 25   FORMAT(2(A10,I10),2(A10,1PE10.3),A5,A2,A5,I2,3A10,A7)             GENRAT    45
```

SUBROUTINE GENRAT (Continued)

```
26    FORMAT(A1,R9,7A10,A5,A2,A5,I2,3A10,A7)
27    FORMAT(1H+,79X,A5,A2,A5,I2,3A10,A7)
30    FORMAT (8A10)
 31   FORMAT (1H1,25X,26H****    SRI PUFF 8    **** /)
41    FORMAT (1H1)
50    FORMAT (A10,* BOUNDARY AT *A5,* SURFACE*)
 1062 FORMAT(2A4,I2,I10,3(2A5,E10.3))
 1064 FORMAT(2A5,I10,3(2A5,E10.3))
 1073 FORMAT(2(A10,I10),A10,E10.3,A10,10I1)
 1075 FORMAT(8E10.3)
 1090 FORMAT(A1,A9,2A10,E10.3,2(A7,3I1),A7,I3,A8,I2)
 62   FORMAT(2A4,I2,I10,3(2A5,1PE10.3),A5,A2,A5,I2,3A10,A7)
 64   FORMAT(2A5,I10,3(2A5,1PE10.3),A5,A2,A5,I2,3A10,A7)
66    FORMAT(A10,I10,A10,10I5,A5,A2,A5,I2,3A10,A7)
 67   FORMAT(86H0 ****  ABORT  FOLLOWING LINE DOES NOT FIT YIELD, MELT,
     1VISCOSITY, OR SPALL CATEGORIES/8A10)
69    FORMAT( )
 73   FORMAT(2(A10,I10),A10,1PE10.3,A10,10I1)
 75   FORMAT(1P8E10.3)
80    FORMAT(10X,14I5)
 90   FORMAT(A1,A9,2A10,1PE10.3,2(A7,3I1),A7,I3,A8,I2,A5,A2,A5,I2,3A10,
     1  A7)
96    FORMAT (31H ERROR  JFIN EXCEEDS 301, JFIN=I4)
C
97    FORMAT (*1         DATA BANK WITH HEADING -- *A10,*-- ON FILE*
     1  I2/)
98    FORMAT (*0   DATA BANK WITH HEADING -- *A10,*-- ON FILE*I2/)
100   DO 101 I=1,456
 101  AZERO(I)=0.
      DO 103 I=1,4000
 103  X(I)=0.
      DO 105 I=1,5000
 105  A(I)=0.
      DO 109 I=1,72
 109  EQSTA(I)=0.
      DO 111 I=1,150
111   EMELT(I)=0.
      DO 113 I=1,318
 113  TSR(I)=0.
      DO 115 I=1,18
 115  YO(I)=0.
      DO 117 I=1,26
 117  IEOS(I)=0
      DO 119 I=1,23
 119  SSTOP(I)=0.
      LVTOT=4000
      LL=LVTOT+200
      DO 121 I=1,LL
121   LVAR(I)=0
      JSMAX=1
      CALL SECOND(FIRST)
      LINTER=5R    L   $   NORMAL=5R    N   $   POROUS=PRESS=5R    P
      MIRROR=5R    M   $   RINTER=5R    R   $   SOLID =SPALL=5R    S
      INF=5R    I
      AHEAD=9HHEADING  $   BHEAD=1H        $   DISCPT(1)=10H   DATE =
      IDD=1H       $   NIND=5H IND=    $   NIN=5H, IN=
      NAT=10H  SEC         $  NBT=10H , , , CM,    $  NCT=10H , , , CM/
      NDT=10HSEC           $  N5T=10H G/CM3        $  NFT=10H DYN/CM2,=
      NGT=10H, ERG/G, ,    $  NHT=10H , DYN/CM2    $  NIT=10H, ERG/G
      NJT=10H DYN/CM2,     $  NKT=10H ERG/G,       $  NLT=10H CM, CM,
      IN = 5  $  OUT = 6
      CALL DATE(DISCPT(2))
C
C        ****  READ AND PRINT DATA  ****
C
152   READ (5,30) (ITITLE(I),I=1,8)
C               CHECK FOR END OF LAST DATA DECK
      IF (EOF(5)) 153,154
153   STOP 70001
 154  IF (ITITLE(1) .NE. 5H DATA .AND. ITITLE(1) .NE. 9H ABS DATA) GO TO
     1 157
      JO=4
      IF (ITITLE(1) .EQ. 9H ABS DATA) JO=2
      IF (ITITLE(2) .EQ. IDD) PRINT 98,ITITLE(1),JO
      IF (ITITLE(2) .NE. IDD) PRINT 97,ITITLE(1),JO
```

```
  155 READ (5,30)(A(I),I=1,8)
      IF (EOF(5)) 152,156
  156 WRITE (JO,30)(A(I),I=1,8)
      IF (ITITLE(2) .NE. IDD) WRITE (6,30) (A(I),I=1,8)
      GO TO 155
  157 DECODE (80,26,ITITLE) IND,(DISCPT(I),I=3,10)
      WRITE (6,31)
      WRITE (6,30) DISCPT(1),DISCPT(2)
      WRITE (6,26) IDD,(DISCPT(I),I=3,10),NIND,IND,NIN,IN
      IN=5
      IF (IND .EQ. IDD) GO TO 158
      IF (IND .EQ. 1HD) GO TO 190
C          ACTIVATE PROCEDURE FOR READING FROM TAPE 4
      IN=4    $    CALL REDR(AHEAD,DISCPT(10),IN,2)
      READ (IN,26) A1
  158 READ (IN,26) INDC,(A(I),I=1,8)
      IF (INDC .NE. 1HC) GO TO 159
      WRITE (6,26) IDD,(A(I),I=1,8),NIND,IDD,NIN,IN
      GO TO 158
  159 DECODE (80,23,A) A1,NTEDT,A2,NJEDIT,A3,NREZON,A4,NALPHA
      IF (NALPHA .EQ. 0) NALPHA=1
      WRITE (6,23) A1,NTEDT,A2,NJEDIT,A3,NREZON,A4,NALPHA,NIND,IDD,
     1 NIN,IN
      IF (NTEDT .EQ. 0) GO TO 170
      DO 165 NT=1,NTEDT,7     $    NZ=NT-1
      READ (IN,1020) A(NT),(TEDIT(I+NZ),I=1,7)
  165 WRITE (6,20) A(NT),(TEDIT(I+NZ),I=1,7),NIND,IDD,NIN,IN,NAT
  170 IF (NJEDIT .EQ. 0) GO TO 175
      NZ=8*NJEDIT
      READ (IN,30) (A(4000+I),I=1,NZ)
      WRITE (6,30) (A(4000+I),I=1,NZ)
      WRITE (6,27) NIND,IDD,NIN,IN
  175 IF (NREZON) 178,180,177
  177 READ (IN,22) A1,(NTR(I),I=1,NREZON)
      WRITE (6,22) A1,(NTR(I),I=1,14),NIND,IDD,NIN,IN
      READ (IN,22) A1,(JREZON(I),I=1,NREZON)
      WRITE (6,22) A1,(JREZON(I),I=1,14),NIND,IDD,NIN,IN
      GO TO 180
  178 READ(IN,1021)A1,DTMAX,A2,TREZON,A3,NARZ,A4,TARZ
      WRITE(6,21)A1,DTMAX,A2,TREZON,A3,NARZ,A4,TARZ,NIND,IDD,NIN,IN,NAT
  180 READ(IN,1025)A1,NEDIT,A2,JCYCS,A3,CKS,A4,TS
      WRITE (6,25) A1,NEDIT,A2,JCYCS,A3,CKS,A4,TS,NIND,IDD,NIN,IN,NBT,
     1  NAT
      NPERN=1
      IF (NEDIT .GT. 0 .OR. JCYCS .EQ. 0) GO TO 190
      NEDIT=MAX0(1,-NEDIT)
      READ (IN,22) A1,(MAXPR(I),I=1,14)
      WRITE (6,22) A1,(MAXPR(I),I=1,14),NIND,IDD,NIN,IN
  190 READ(IN,1073)A1,NMTRLS,A2,MATFL,A3,UZERO,A4,IPLOT,NSCRB
      WRITE (6,73) A1,NMTRLS,A2,MATFL,A3,UZERO,A4,IPLOT,NSCRB
      WRITE(6,27) NIND,IDD,NIN,IN,NCT,NDT
      IIPLOT=0
      DO 191 I=1,4
      IIPLOT=IIPLOT+IPLOT(I)
      IF (IPLOT(I) .EQ. 0) GO TO 191
      READ(IN,1020)A1,XMAX(I),XMIN(I),YMAX(I),YMIN(I)
      WRITE (6,20) A1,XMAX(I),XMIN(I),YMAX(I),YMIN(I)
  191 CONTINUE
      IF (IIPLOT .EQ. 0) GO TO 192
      READ (IN,22) A1,IA
      WRITE (6,22) A1,IA
  192 CONTINUE
      IF (IND .EQ. 1HX) CALL EXTRA
C
C          ****  M-LOOP   ****
C
      DO 290 M=1,NMTRLS
      IN=5
      WRITE (6,69)
      CZQ(M)=4.    $    CWQ(M)=0.15
      TENS(M,1)=TENS(M,2)=-1.E11    $    TENS(M,3)=-1.0
      YOS=0.
  200 READ(IN,1090)IND,MATL(M,1),MATL(M,2),A1,RHOS(M),A2,NCMP(M),NFR(M),
     1  NPOR(M),A3,NDS(M),NPR(M),NYAM,A4,NVAR(M),A5,NCON(M)
      WRITE (6,90) IDD,MATL(M,1),MATL(M,2),A1,RHOS(M),A2,NCMP(M),NFR(M),
```

```
     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
      IF (IN .EQ. 4 .OR. IND .EQ. IDD) GO TO 205                      GENRAT     198
      IN=4                                                            GENRAT     199
      CALL REDR(MATL(M,1),MATL(M,2),IN,2)                             GENRAT     200
      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
      IF (NCMP(M) .NE. 0) GO TO 2055                                  GENRAT     206
C          ****   READ IN EQST VARIABLES    ****                      GENRAT     207
      READ(IN,1020)A1,EQSTC(M),EQSTD(M),EQSTE(M),EQSTG(M),EQSTH(M),   GENRAT     208
     1  EQSTS(M),EQSTV(M)                                             GENRAT     209
      WRITE (6,20) A1,EQSTC(M),EQSTD(M),EQSTE(M),EQSTG(M),EQSTH(M),   GENRAT     210
     1  EQSTS(M),EQSTV(M),NIND,IDD,NIN,IN,NFT,NGT,NHT,NIT             GENRAT     211
      IF (A1 .EQ. 10H EQSTX=    ) READ (IN,1020) A2,EQSTA(M),A3,A4,A5,A6, GENRAT  212
     1  A7,A8                                                         GENRAT     213
      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
      IF (NPR(M) .LE. 1 .OR. NPR(M) .EQ. 7) CALL EQST(A1,A2,A3,M)     GENRAT     216
      LS = -1                                                         GENRAT     217
      EMELT(M)=0.1*EQSTE(M)                                           GENRAT     218
      CALL FMELT(LS,M,EMELT(M),A1,A2,X,MS)                            GENRAT     219
      IF (NCMP(M) .EQ. 0) GO TO 2059                                  GENRAT     220
C                                                                     GENRAT     221
C     ****  READ COMPOSITE DATA   ****                                GENRAT     222
2055  CONTINUE                                                        GENRAT     223
      LS=-1                                                           GENRAT     224
      CALL REBAR(LS,IN,J,I,M,N,H(J,3),RHOS(M),DOLD,EXMAT(M,3),SY,SZ,  GENRAT     225
     1  TXY,EH,PHL(J),EX,EY,EZ,EXY,F,0.,0.,ESC,FS,COM(1),COM(2),COM(6), GENRAT   226
     2  COM(7),YO(M),COM(8),IPRT)                                      GENRAT     227
      NVAR(M)=MAXO(NVAR(M),7)                                         GENRAT     228
       RHO(M)=RHOS(M)                                                 GENRAT     229
      GO TO 245                                                       GENRAT     230
 2059 CONTINUE                                                        GENRAT     231
C                                                                     GENRAT     232
C     ****   READ FRACTURE DATA   ****                                GENRAT     233
      NFRM=NFR(M)+1                                                   GENRAT     234
      GO TO (210,206,207,208,208,207) NFRM                            GENRAT     235
206   CONTINUE                                                        GENRAT     236
207   CONTINUE                                                        GENRAT     237
      READ(IN,1020)A1,(TSR(M,I),I=1,7)                                GENRAT     238
      WRITE (6,20) A1,(TSR(M,I),I=1,7),NIND,IDD,NIN,IN                GENRAT     239
      IF (NFR(M) .EQ. 1) GO TO 210                                    GENRAT     240
      IF (NFR(M) .EQ. 2)  NVAR(M)=MAXO(NVAR(M),18)                    GENRAT     241
      IF (NFR(M) .EQ. 5)  NVAR(M)=MAXO(NVAR(M),11)                    9/12/79      3
      READ(IN,1020)A1,(TSR(M,I),I=8,14)                               GENRAT     243
      WRITE (6,20) A1,(TSR(M,I),I=8,14),NIND,IDD,NIN,IN               GENRAT     244
      GO TO 210                                                       GENRAT     245
C     READ FOR SHEAR BAND MODEL.                                      GENRAT     246
 208  CALL SHEAR2(LSUB(15),IN,M)                                      GENRAT     247
      NVAR(M)=MAXO(NVAR(M),5)                                         GENRAT     248
      LSUB(15)=1                                                      GENRAT     249
      IF (NFR(M) .EQ. 4) GO TO 207                                    GENRAT     250
C                                                                     GENRAT     251
C     ****   READ POROUS DATA   ****                                  GENRAT     252
210   IF (NPOR(M) .EQ. 0) GO TO 230                                   GENRAT     253
      NPORM = NPOR(M)                                                 GENRAT     254
      GO TO (211,212,225,227) NPORM                                   GENRAT     255
 211  READ(IN,1020)A1,RHO(M)                                          GENRAT     256
      WRITE (6,20) A1,RHO(M)                                          GENRAT     257
      CALL POREQST(0,IN,M,EXMAT(M,3),RHO(M),A2,A3,A4,A5,A6,CZQ(M),CWQ(M) GENRAT  258
     1  ,A7,A8,EQSTC(M),EQSTD(M),EQSTG(M),EQSTS(M),A11,A12,YO(M))     GENRAT     259
      GO TO 230                                                       GENRAT     260
212   IF (NPOR(M) .GT. 2) GO TO 225                                   GENRAT     261
      CALL PORHOLT(0,IN,M,EXMAT(M,3),RHO(M),DOLD,A1,A2,A3,A4,A5,A6,A7, GENRAT    262
     1   EQSTC(M),A9,YO(M),RHOS(M),A10)                               GENRAT     263
      GO TO 230                                                       GENRAT     264
 225  READ(5,1020)A1,RHO(M)                                           GENRAT     265
      WRITE(6,20)A1,RHO(M)                                            GENRAT     266
      CALL PEST(LSUB(14),5,A1,A2,A3,A4,A5,M,EXMAT(M,3),RHO(M),A6,RHOS(M) GENRAT  267
     1  ,A7,A8,A9,A10,A11,A12,A13,EQSTC(M),EQSTD(M),EQSTS(M),EQSTG(M), GENRAT    268
     2  A14,YO(M),A15,A16,CZQ(M),CWQ(M),EQSTH(M),EQSTE(M),EQSTN(M),EQSTV GENRAT   269
     3  (M),EQSTA(M))                                                 GENRAT     270
```

SUBROUTINE GENRAT (Continued)

```
      NVAR(M)=MAXO(NVAR(M),5)
      GO TO 230
  227 READ(5,1021)A1,RHO(M),A2,MU(M)
      WRITE(6,21)A1,RHO(M),A2, MU(M)
      MU(M)=1.333*MU(M)
      CALL CAP1(-1,IN,M,H(1),RHO(M),RHO(M),EHL(1),0.,0.,0.,0.,1.,MU(M),
     1  EQSTC(M),EQSTG(M),RHOS(M),SHL(1),SHL(1),SHL(1),SHL(1),NEM(1),
     2  K,J,NET(1))
      EXMAT(M,3)=SQRT((EQSTC(M)+MU(M))/RHO(M))
C         ****  READ  SPECIAL PRESSURE AND DEVIATOR STRESS DATA   ****
  230 IF (NDS(M) .EQ. 0) GO TO 235
      IF (NDS(M) .EQ. 7) GO TO 233
      READ(IN,1020)A1,(TSR(M,I),I=15,21)
      WRITE (6,20) A1,(TSR(M,I),I=15,21),NIND,IDD,NIN,IN
      GO TO 235
  233 CALL EP(0,M)
  235 IF (NPR(M) .EQ. 0) GO TO 245
      NPRM = NPR(M)
      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,A9,A10)
      GO TO 245
  237 CALL ESA(0,IN,M)
      GO TO 245
  238 CALL EQSTPF(0,IN,M)
      GO TO 245
  239 CALL HYPO(0,IN,M,EXMAT(M,3),RHOS(M))
      GO TO 245
  240 CALL GRAY(0,IN,M)
      GO TO 245
  241 CALL EOSTAB(0,IN,XN,YN,ZN)
      GO TO 245
C         ****  READ SPALL, VISCOSITY, YIELD AND MELT VARIABLES   ****
C     NYAM IS THE NUMBER OF CARDS
  245 IF (NYAM .EQ. 0)  GO TO 280
      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. 1H ) GO TO 250
      IF (A(I) .EQ. 1HT .AND. A(I+1) .EQ. 1HE) GO TO 252
      IF (A(I).EQ.1HC .OR. A(I).EQ.1HV) GO TO 253
      IF (A(I) .EQ. 1HY) GO TO 254
      IF (A(I) .EQ. 1HE .OR. A(I) .EQ. 1HM) GO TO 270
      IF (A(I) .EQ. 1HT .AND. A(I+1) .EQ. 1HH) GO TO 256
      IF (A(I) .EQ. 1HG  .AND. A(I+1) .EQ. 1HM) GO TO 272
      IF (A(I) .EQ. 1HS  .AND. A(I+1) .EQ. 1HP) GO TO 265
  250 CONTINUE
      PRINT 67, (X(I), I=1,8)
      GO TO 398
  252 DECODE(80,1020,X)A1,(TENS(M,I),I=1,3)
      WRITE (6,20) A1,(TENS(M,I),I=1,3),(T(I),I=1,4),NIND,IDD,NIN,IN,NFT
      GO TO 275
  253 DECODE(80,1020,X)A1,CZQ(M),CWQ(M),C2(M)
      WRITE (6,20) A1,CZQ(M),CWQ(M),C2(M),(T(I),I=1,4),NIND,IDD,NIN,IN
      GO TO 275
  254 DECODE(80,1020,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,IN,NJT,NJT,NJT
      IF (NDS(M) .NE. 5) YADD(M)=YADD(M)/(RHOS(M)*(.2-.5*YOS/MU(M)))
      IF (NPOR(M) .EQ. 0)  YO(M)= YOS
C         TEST FOR COULOMB FRICTION MODEL
      IF (EXMAT(M,1) .EQ. 0.) GO TO 275
C         READ IN EXMAT AS TAN(PHI), AND YOS AS 2C
      ENPHI=SQRT(1.+EXMAT(M,1)**2)+EXMAT(M,1)
      YO(M)=1.5*YOS*ENPHI/(1.+0.5*ENPHI**2)
      EXMAT(M,1)=1.5*(ENPHI**2-1.)/(1.+0.5*ENPHI**2)
      GO TO 275
  265 DECODE(80,1020,X)A1,SPH(M)
      WRITE(6,20)A1,SPH(M)
      GO TO 275
  270 LS = 0
      GO TO 273
  272 LS= 1
  273 DECODE (80,1020,X)A1,(A(I),I=1,7)
      WRITE (6,20) A1,(A(I), I=1,7),NIND,IDD,NIN,IN,NKT
```

```
      IF (MS .EQ. M) GO TO 2732
      DO 2731 I=1,7
      IF (LS .EQ. 0) EMELT(M,I)=A(I)
      IF (LS .EQ. 1) GMELT(M,I)=A(I)
 2731 CONTINUE
      GO TO 2733
 2732 IF (LS .EQ. 0) EMELT(M,8)=A(1)
      IF (LS .EQ. 1) GMELT(M,8)=A(1)
 2733 CALL FMELT(LS,M,EMELT(M),A1,A2,A,MS)
      GO TO 275
  256 DECODE(80,1020,X)A1,(THERM(M,I),I=1,5)
      WRITE (6,20) A1,(THERM(M,I),I=1,7),NIND,IDD,NIN,IN,NKT
  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)=EQSTS(M)
      ESC(M,5)=MU(M)    $  ESC(M,6)=YADD(M)
      ESC(M,7)=RHOS(M)   $  ESC(M,9)=EQSTG(M)
      ESC(M,10)=YO(M)
      THERM(M,6)=EMELT(M,1)
      THERM(M,8)=EQSTE(M)
  290 CONTINUE
C
C          ****   END OF M-LOOP****
      WRITE (6,69)
C
C          **** READ IN ZONING VARIABLES ****
C
      DO 291 L=1,30
      JBND(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,IN
      IF (NLAYER .LE. 10) GO TO 292
      READ (5,80) (JMAT(L),L=11,NLAYER)
      WRITE (6,80) (JMAT(L),L=11,NLAYER)
  292 INFF=INFL=0
      IF (JMAT(1) .LT. 0) INFF=1
      IF (JMAT(NLAYER) .LT. 0) INFL=1
      JMAT(1)=IABS(JMAT(1))
      JMAT(NLAYER)=IABS(JMAT(NLAYER))
      READ (5,30) (X(I),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 .LE. 7) GO TO 2921
      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
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(1),NCELLS(1,1),A3,A4,TH(1,1),A5,A6,
     1  DELX(1,1),A7,A8,DELFIN(1,1)
      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 N1=2,NZON
      READ(5,1064)A1,A2,NCELLS(L,N1),A3,A4,TH(L,N1),A5,A6,DELX(L,N1),
     1  A7,A8,DELFIN(L,N1)
  295 WRITE (6,64) A1,A2,NCELLS(L,N1),A3,A4,TH(L,N1),A5,A6,DELX(L,N1),
     1  A7,A8,DELFIN(L,N1),NIND,IDD,NIN,IN,NLT
 2951 IF (A5 .NE. 5H INCH) GO TO 300
C
C        CONVERSION OF TH(L,N), DELX(L,N) FROM INCHES TO CM
      DO 2952 N1=1,NZON
```

SUBROUTINE GENRAT (Continued)

```
      TH(L,N1)=2.54*TH(L,N1)
2952  DELX(L,N1)=2.54*DELX(L,N1)
300   CONTINUE
C
C            ****   CALCULATE ZONING AND INITIALIZE CELL COORDINATES   ***
      NULL=0  $  XZERO=0.  $  J=1  $  X(1)=0.
      DO 390 L=1,NLAYER
      IF (JMAT(L) .EQ. 0) GO TO 385
      NZON=NZONES(L)
        DO 380 NZ=1,NZON
      FN=NCELLS(L,NZ)  $  RATIO=1.  $  FI=0.  $  DX=DELX(L,NZ)
      IF (DX*DELFIN(L,NZ) .EQ. 0.)  GO TO 345
C                PREPARE FOR GEOMETRIC PROGRESSION OF CELLS
      RATIO=DELFIN(L,NZ)
      DX=(1.-RATIO)/(1.-RATIO**FN)*TH(L,NZ)   $   GO TO 360
C                PREPARE FOR ARITHMETIC PROGRESSION OF CELLS
345       IF (DX .NE. 0.) GO TO 355
      IF (DELFIN(L,NZ) .NE. 0.)   GO TO 350
      DX=TH(L,NZ)/FN  $  GO TO 360
350   DX=2.*TH(L,NZ)/FN-DELFIN(L,NZ)
355   FI=2.*(TH(L,NZ)/FN-DX)/(FN-1.)
360   JN=J+NCELLS(L,NZ)  $  J1=J+1
          DO 365 I=J1,JN
            X(I)=X(I-1)+DX
      XO(I)=X(I)
365       DX=RATIO*DX+FI
380   J=JN  $  JBND(L)=J  $  J=J+1
      XZERO=X(J)=X(J-1)
      XO(J)=X(J)
      GO TO 390
385   XZERO=X(J)=XZERO+TH(L,1)
      NULL=NULL+1
390   CONTINUE
      JINIT=1  $  JFIN=J
C   ****  RESET JBNDS IF SOME LAYERS ARE VACANT.
      NULL=0.
      DO 395 L=1,NLAYER
      IF (JMAT(L) .EQ. 0) GO TO 393
      JBND(L-NULL)=JBND(L)
      JMAT(L-NULL)=JMAT(L)
      GO TO 395
393   NULL=NULL+1
395   CONTINUE
      NLAYER=NLAYER-NULL
 396  IF (JFIN .LE. 201) GO TO 399
      WRITE ( 6,96) JFIN
 398  READ(5,30) (A(I), I=1,8)
      PRINT 30, (A(I), I=1,8)
      IF (EOF(5)) 100,398
C
C         ****  READ RADIATION SOURCE DATA  ****
399   IF (MATFL .EQ. 0) CALL DEPOS(2,IN)
C
C         ****    INITIALIZE THE J-ARRAY VARIABLES    ****
C
      DO 601 I=1,2400
601   CHL(I)=0.
      J1=1
      LVMAX=1
      DO 630 L=1,NLAYER
      M=JMAT(L)
      YOM=YO(M)
      IF (NPOR(M) .NE. 0) GO TO 602
      IF (NPR(M) .EQ. 4) GO TO 602
      DET=EXMAT(M,3)
      CJ=AMAX1(DET,SQRT((EQSTC(M)+1.333*MU(M))/RHOS(M)),5.E4)
      HH=SOLID   $   GO TO 603
602   HH=POROUS   $   CJ=EXMAT(M,3)
      IF (RHO(M) .EQ. RHOS(M)) HH = SOLID
603   JN=JBND(L)
      DO 610 J=J1,JN
      CHL(J)=CJ
      DHL(J)=RHO(M)
      H(J,1)=HH
      IF (NPR(M) .EQ. 1) CALL EXPLODE(2,IN,M,EHL,DHL,DOLD,PHL,SHL,NEM,X,
```

SUBROUTINE GENRAT (Continued)

```
     1 J,A1,A2,A3)
      IF (NVAR(M) .LE. 0) GO TO 604
      LVAR(J)=LVMAX
      LVMAX=LVMAX+NVAR(M)
  604 CONTINUE
      IF (NPOR(M) .EQ. 3) COM(LVMAX+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)=DHL(J)*(X(J+1)**NALPHA-X(J)**NALPHA)
      NDSM1=NDS(M)+1
      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
 620  CONTINUE
      T(JN)=TENS(M,3)
      H(JN,2)=LINTER
      J1=JN+1
 630  CONTINUE
      ZHL(JFIN-1)=0.
      H(1,2)=H(JFIN,2)=SPALL
      IF (INFF .EQ. 1) H(1,2)=INF
      IF (INFL .EQ. 1) H(JFIN,2)=INF
      IF (INFF .EQ. 1) ZHL(INFF-1)=ZHL(1)
C     TO ACTIVATE THIS ROUTINE, DTMAX IS NEGATIVE OF NUMBER OF CELLS
C          DESIRED IN LAYER NUMBER(-NREZON)
      IF (DTMAX .GT. 0. .OR. NREZON .GE.0)  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
 635  CONTINUE
      DTNH=1.E-12
C                CHECK FOR END OF DATA DECK AND CALL FOR ADDED READS
C                INSERT CARD HERE READING EXTRA
  638 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
      DO 810 J=1,JFIN2
 810  U(J)=UZERO
      DTNH=0.02*AMIN1((X(JFIN2)-X(JFIN2-1))/CHL(JFIN2-1),(X(JFIN2+2)-
     1 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(1,2)=PRESS
      GO TO 818
 817  JSTAR=JFIN  $  SDURM=1.  $  H(JFIN,2)=PRESS
C
C*********      VELOCITY EDIT
 818  IF (H(1,2) .EQ. SPALL) GO TO 819
      A2=5HFRONT
```

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  PRESSURE
      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  PRESSURE
      PRINT 50,A1,A2
 8195 WRITE (6,16) (DISCPT(I),I=1,10)
      IF (EHL(J) .GT. 1.) JSTAR=MAXO(JSTAR,J)
      DO 820 J=1,JFIN
  820 A(J)=X(J+1)-X(J)
      L=K=J1=1
  825 J2=MINO(JFIN-1,50*K,JBND(L))
      M=JMAT(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,I),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    ****
      IF (NJEDIT .GE. 1) CALL PRESCR
C
      RETURN
      END
```

SUBROUTINE GRAY

```
      SUBROUTINE GRAY(NPART,IN,M,AMU,EMELT,D,E,CH,P,DPDD,DPDE,IH)
C
C        GRAY 3-PHASE EOS OF ROYCE AT LLL.  REF UCRL-51121
C     MODIFIED IN THE LIQUID-VAPOR REGION BY YOUNG AT LLL.
C     REF. UCRL-51575
C
      DIMENSION A(10),ALFLS(10),AYBLS(10),C(10),CE2(10),CTA2(10),
     1  CTB1(10),CTB2(10),C1LS(10),C2LS(10),C3LS(10),DSLS(10),D1LS(10),
     2  D2LS(10),D3LS(10),E00(10),G0(10),GPLS(10),PCCLS(10),RPLS(10),
     3  S(10),TH(10),TMO(10),VB(10),VO(10),XJ(10),ZJ(10)
      DIMENSION TEMP(50,10),PRES(50,10),VMN(50,10),VMX(50,10),EMN(50,10)
     1   ,EMX(50,10),JMX(10)
      IF (NPART .GT. 0) GO TO 40
C
C     ****      READ DATA AND PRINT                                ****
      READ 1000,A1,A(M),ALFLS(M),AYBLS(M),C(M),CE2(M),CTA2(M),CTB1(M),
     1    A2,CTB2(M),C1LS(M),C2LS(M),C3LS(M),DSLS(M),D1LS(M),D2LS(M),
     2    A3,D3LS(M),E00(M),G0(M),GPLS(M),PCCLS(M),RPLS(M),S(M),
     3    A4,TH(M),TMO(M),VB(M),VO(M),XJ(M),ZJ(M)
      PRINT 1000,A1,A(M),ALFLS(M),AYBLS(M),C(M),CE2(M),CTA2(M),CTB1(M),
     1    A2,CTB2(M),C1LS(M),C2LS(M),C3LS(M),DSLS(M),D1LS(M),D2LS(M),
     2    A3,D3LS(M),E00(M),G0(M),GPLS(M),PCCLS(M),RPLS(M),S(M),
     3    A4,TH(M),TMO(M),VB(M),VO(M),XJ(M),ZJ(M)
 1000 FORMAT(A10,1P7E10.3)
      READ 1001,A1,JMX(M)
      PRINT 1001,A1,JMX(M)
1001  FORMAT(A10,I10)
      IMAX=JMX(M)
      DO 30 I=1,IMAX
      READ 1000,A1,TEMP(I,M),VMX(I,M),VMN(I,M),EMX(I,M),EMN(I,M),PRES(I,
     1     M)
      PRINT 1000,A1,TEMP(I,M),VMX(I,M),VMN(I,M),EMX(I,M),EMN(I,M),PRES(I
     1     ,M)
30    CONTINUE
      RETURN
C
C     ****       COMPUTE PRESSURE, SOUND SPEED, (DP/DD)E, AND (DP/DE)V
40    D1=D
      E1=E
      NLOOP=0
50    NLOOP=NLOOP+1
      IF (NLOOP-2) 100,55,60
55    PR1=P
      D2=D=D1+.001*D1
      GO TO 100
60    PR2=P
      D=D1
      E=E3=E+.001*E+1.E5
100   X=(VO(M)-1./D)/VO(M)
      V=1./D
      IF (V .LE. 1.04*VO(M)) GO TO 145
      EN=EMX(2,M)
      IF (E .GE. 1.2*EN) GO TO 140
C
C     USE CRITICAL POINT AND TIE LINES TO REPLACE VAN DER WAALS LOOPS IN
C        LIQUID-VAPOR PHASE.
C
      VXM1=VMM1=VMN(1,M)
      EXM1=EMM1=EMN(1,M)
      EQM=EMM1+(EN-EMN(2,M))*(V-VMM1)/(VMX(2,M)-VMN(2,M))
      JMAX=JMX(M)
      DO 110 J=2,JMAX
      VX=VMX(J,M)
      EX=EMX(J,M)
      VN=VMN(J,M)
      EN=EMN(J,M)
      EQ=EN+(EX-EN)*(V-VN)/(VX-VN)
```

```
      IF (E .LT. EQ) GO TO 105                                          GRAY  69
      EH=EN+(EMM1-EN)*(V-VN)/(VMM1-VN)                                  GRAY  70
      IF (E .GE. EH) GO TO 140                                          GRAY  71
      EL=EX+(EX-EXM1)*(V-VX)/(VX-VXM1)                                  GRAY  72
      IF (E-EL) 115,140,140                                             GRAY  73
105   EQM=EQ                                                            GRAY  74
      VXM1=VX                                                           GRAY  75
      VMM1=VN                                                           GRAY  76
      EXM1=EX                                                           GRAY  77
      EMM1=EN                                                           GRAY  78
110   CONTINUE                                                          GRAY  79
      J=JMAX                                                            GRAY  80
      EL=EX*VX/V                                                        GRAY  81
      IF (E .GE. EL) GO TO 140                                          GRAY  82
      SL=-EN/(.96/V0(M)-1.0/VN)                                         GRAY  83
      EH=EN+SL*(1.0/V-1.0/VN)                                           GRAY  84
      IF (E .GE. EH) GO TO 145                                          GRAY  85
      TM1=TEMP(J-1,M)                                                   GRAY  86
      TM=TEMP(J,M)                                                      GRAY  87
      T=(E/EQM)*(TM-300.)+300.                                          GRAY  88
      NAME=10H LIQ-VAPOR                                                GRAY  89
      GO TO 120                                                         GRAY  90
115   NAME=10H LIQ-VAPOR                                                GRAY  91
      ER=(EQM-E)/(E-EQ)                                                 GRAY  92
      TM=TEMP(J,M)                                                      GRAY  93
      TM1=TEMP(J-1,M)                                                   GRAY  94
      T=(TM*ER+TM1)/(1.+ER)                                             GRAY  95
120   PQ=PRES(J)                                                        GRAY  96
      SLP=ALOG(PRES(J-1,M)/PQ)/(1./TM1-1./TM)                           GRAY  97
      P=PQ*EXP(SLP*(1.0/T-1.0/TM))                                      GRAY  98
      IH=5R    D                                                        GRAY  99
      GO TO 700                                                         GRAY 100
C         BRANCH TO VAPOR LIQUID STATE (6)                              GRAY 101
140   IF (X .LT. XJ(M)) GO TO 600                                       GRAY 102
C         START COMPUTATIONS FOR SOLID-LIQUID STATES (5)                GRAY 103
145   E0=(C(M)*X)**2/2./(1.-S(M)*X)*(1.+S(M)*X/3.+CE2(M)*X*X)           GRAY 104
     1 +E00(M)*(1+G0(M)*X)                                              GRAY 105
      G=G0(M)-A(M)*X                                                    GRAY 106
      P1=(C(M)/(1.-S(M)*X))**2*X*D*(1.-X-0.5*G*X) +G*E*D                GRAY 107
      IF (X .GE. 0.) GO TO 150                                          GRAY 108
      TM=TM0(M)/(1.-X)**2*(1.+(CTB1(M)-2.)*X+CTA2(M)*X*X)               GRAY 109
      GO TO 155                                                         GRAY 110
150   TM=TM0(M)*(1.+CTB1(M)*X+CTB2(M)*X*X)                              GRAY 111
155   CONTINUE                                                          GRAY 112
      DT=DSLS(M)*((CTB1(M)-A(M)*X)*TM/C(M))**2/(2.4*(1.-X)*(1.+         GRAY 113
     1  AMAX1(0.,(4*S(M)-1.)*X)))                                       GRAY 114
      TMDT=TM-DT                                                        GRAY 115
      EM1=E0+TMDT*(3.*RPLS(M)+0.5*GPLS(M)*TMDT)                         GRAY 116
C         BRANCH POINT                                                  GRAY 117
      IF (E .GT. EM1) GO TO 300                                         GRAY 118
C                                                                       GRAY 119
C     ****      SOLID EQUATION OF STATE                            **** GRAY 120
200   QUAD=9.*RPLS(M)**2+2.*GPLS(M)*(E-E0)                              GRAY 121
      IF (QUAD .LT. 0.) GO TO 800                                       GRAY 122
      T=(-3.*RPLS(M)+SQRT(QUAD))/GPLS(M)                                GRAY 123
      PC=(0.666667-G)*GPLS(M)*T**2*D/2.                                 GRAY 124
      P=P1+PC+PCCLS(M)                                                  GRAY 125
      IH=5R    S                                                        GRAY 126
      GO TO 700                                                         GRAY 127
300   TMDT=TM+DT                                                        GRAY 128
      EM2=E0+TM*(DSLS(M)-0.5*ALFLS(M)) + TMDT*(3.*RPLS(M)+0.5*TMDT*     GRAY 129
     1  (GPLS(M)-ALFLS(M)/TM))                                          GRAY 130
      ALAMB =-CTB1(M)+2.*A(M)*X                                         GRAY 131
C         BRANCH POINT                                                  GRAY 132
      IF (E .GT. EM2) GO TO 400                                         GRAY 133
C                                                                       GRAY 134
```

SUBROUTINE GRAY (Concluded)

```
C     ****     MELT EQUATION OF STATE                                ****
      ENU=(E-EM1)/(EM2-EM1)
      ENUSA=ENU*(DSLS(M)-ALFLS(M))
      RNU=3.*RPLS(M)+ENUSA
      QUAD=RNU**2+2.*GPLS(M)*(E-EO+ENU*DT*ENUSA)
      IF (QUAD .LT. 0.) GO TO 800
      T=(-RNU+SQRT(QUAD))/GPLS(M)
      PC=D*(0.5*(0.666667-G)*GPLS(M)*T*T-ENUSA*(ALAMB*TM+G*(T-ENU*DT)))
      P=P1+PC+PCCLS(M)
      IH=6R     M
      GO TO 700
400   EGG=EO+TM*(28.78*RPLS(M)+DSLS(M)+46.017*GPLS(M)*TM-46.517*ALFLS(M)
     1     )
C          BRANCH POINT
      IF (E .GT. EGG) GO TO 500
C
C     ****     LIQUID EQUATION OF STATE                              ****
      GAT=GPLS(M)-ALFLS(M)/TM
      QUAD=9.*RPLS(M)**2+2.*GAT*(E-EO-TM*(DSLS(M)-0.5*ALFLS(M)))
      IF (QUAD .LT. 0.) GO TO 800
      T=(-3.*RPLS(M)+SQRT(QUAD))/GAT
      PC=D*(0.5*(0.666667-G)*GPLS(M)*T**2-TM*(DSLS(M)-0.5*ALFLS(M)*
     1  (1.+(T/TM)**2))*(ALAMB+G))
      P=P1+PC+PCCLS(M)
      IH=6R     L
      GO TO 700
C
C     ****     HOT LIQUID EQUATION OF STATE                              ****
500   QUAD=(3.*RPLS(M)-9.5934*ALFLS(M))**2+2.*GPLS(M)*(E-EO-TM*(DSLS(M)
     1     +45.517*ALFLS(M)))
      IF (QUAD .LT. 0.) GO TO 800
      T=(-(3.*RPLS(M)-9.5934*ALFLS(M))+SQRT(QUAD))/GPLS(M)
      PC=D*(0.5*(0.666667-G)*GPLS(M)*T*T-TM*(DSLS(M)+ALFLS(M)*(45.517
     1     -9.5934*T/TM)))*(ALAMB+G)
      P=P1+PC+PCCLS(M)
      IH=6R     H
      GO TO 700
C
C     ****     LIQUID-VAPOR EQUATION OF STATE                            ****
600   Z=D*VB(M)
      FE=0.5*VB(M)*(((TH(M)-ZJ(M))/(TH(M)-Z))**2*(2.*Z-2.+TH(M))
     1  -(2.*ZJ(M)-2.+TH(M)))*(TH(M)-ZJ(M))/ZJ(M)**3
      QUAD=(3.*RPLS(M)+2.*D2LS(M))**2-16.*(E+AYBLS(M)*Z-C1LS(M)*FE
     1     -D1LS(M))*(C3LS(M)*FE-D3LS(M))
      IF (QUAD .LT. 0.) GO TO 800
      T=(3.*RPLS(M)+2.*D2LS(M)-SQRT(QUAD))/4./(C3LS(M)*FE-D3LS(M))
      FP=(Z*(TH(M)-ZJ(M))/ZJ(M)/(TH(M)-Z))**3
      P=RPLS(M)*T/VB(M)*Z*(1.+Z*(1.+Z*(1.-Z)))/(1.-Z)**3-AYBLS(M)/VB(M)
     1  *Z*Z +FP*(C1LS(M)+T*(C2LS(M)+C3LS(M)*T))
      IH=6R     V
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 .GT. 0.) CH=SQRT(CH2)
      E=E1
      D=D1
      IF (E .GT. EMELT) P=AMAX1(P,0.)
      RETURN
800   CONTINUE
      IH=6R     Z
      P=0.
      RETURN
      END
```

SUBROUTINE HAFSTEP

```
      SUBROUTINE HAFSTEP                                                HAFSTEP    2
C                                                                       HAFSTEP    3
C         *  CALLED BY HYDRO TO COMPUTE X, U, D, E FOR THE              HAFSTEP    4
C         *  HALFSTEP POINT BETWEEN J AND J+1                           HAFSTEP    5
C      INPUT - J, M.                                                    HAFSTEP    6
C      OUTPUT - UHL, DHL, EHL.                                          HAFSTEP    7
C                                                                       HAFSTEP    8
      INTEGER H,POROUS,PRESS,RINTER,SOLID,SPALL                         PUFCOM     2
      REAL MATL,NEM,NET,NEMH,NETH                                       PUFCOM     3
C              MISCELLANEOUS                                            PUFCOM     4
      COMMON AZERO(1),CEF,CKS,DAVG,DELTIM,DISCPT(10),DOLD,DRHO,DTMAX,   PUFCOM     5
     1  DTMIN,DTN,DTNH,DU,DX,EOLD,F,FAC,FIRST,J,JCYCS,JINIT,            PUFCOM     6
     2  JFIN,JREZON(15),JSMAX,JSTAR,JTS,LSUB(30),M,MAXPR(30),N,NCYCS,   PUFCOM     7
     3  NEDIT,NPERN,NR,NREZON,NSCRB(6),NSEPRAT,NSPALL,NTEDT,            PUFCOM     8
     4  NTEX,NTR(15),POLD,P6(20),R(30),RLAST,SLAST,SMAX,TEDIT(50),      PUFCOM     9
     5  TF,TIME,TJ,TREZON,TS,T6(20),ULAST,UOLD,UZERO,XLAST,XNOW,XOLD    PUFCOM    10
     1  ,XJDIT(20),MS                                                   PUFCOM    11
C              HALFSTEP VALUES                                          PUFCOM    12
      COMMON DH,DHLAST,DUH,EH,PH,RH,RHLAST,SH,SHLAST,UH,UHLAST,XH,XHLAST PUFCOM    13
     1  ,NEMH,NETH                                                      PUFCOM    14
C              CONDITION INDICATORS                                     PUFCOM    15
      COMMON INF,LINTER,MIRROR,NORMAL,POROUS,PRESS,RINTER,SOLID,SPALL   PUFCOM    16
C              CELL LAYOUT                                              PUFCOM    17
      COMMON DXX(30),JBND(30),JMAT(30),NAUTO,MATL(6,2),NLAYER,NMTRLS,   PUFCOM    18
     1  THK(30)                                                         PUFCOM    19
C                                                                       PUFCOM    20
C              COORDINATE ARRAYS                                        COORDCOM   2
      COMMON/COORD/X(200),XO(200),CHL(200),DHL(200),DPDD(200),DPDE(200), COORDCOM  3
     1  EHL(200),H(200,3),NEM(200),NET(200),PHL(200),RHL(200),SDT(200), COORDCOM   4
     2  SHL(200),T(200),U(200),YHL(200),ZHL(200)                        COORDCOM   5
C              NAMED COMMON                                             EQSTCOM    2
      REAL MU,MUM                                                       EQSTCOM    3
      COMMON /EQS/  EQSTA(6),EQSTC(6),EQSTD(6),EQSTE(6),EQSTG(6),       EQSTCOM    4
     1  EQSTH(6),EQSTN(6),EQSTS(6),EQSTV(6),CZQ(6),CWQ(6),C2(6)         EQSTCOM    5
      COMMON /MELT/ EMELT(6,8),GMELT(6,8),SPH(6),THERM(6,8)             EQSTCOM    6
      COMMON /RHO/ RHO(6),RHOS(6)                                       EQSTCOM    7
      COMMON /TSR/ TSR(6,30),EXMAT(6,20),TENS(6,3)                      EQSTCOM    8
      COMMON /Y/ YO(6),YADD(6),MU(6),MUM,YADDM                          EQSTCOM    9
      COMMON /IND/ IEOS(6),INDK(20),NALPHA,NCMP(6),NFR(6),NPOR(6),      INDCOM     2
     1  NDS(6),NPR(6),NCON(6),NVAR(6)                                   INDCOM     3
      COMMON /RAD/ SSTOP(9),START(9),SDURM,SSTOPM,NSPEC,SSJ,JSS,IPLOT(4) RADCOM    2
     1  ,XMAX(4),XMIN(4),YMAX(4),YMIN(4),IA(7),ITITLE(24),NARZ,TARZ     RADCOM     3
C                                                                       HAFSTEP   14
C                                                                       HAFSTEP   15
      DX=X(J+1)-X(J)   $  EOLD=EHL(J)                                   HAFSTEP   16
      DOLD=DHL(J)                                                       HAFSTEP   17
      IF (NALPHA .GT. 1) GO TO 20                                       HAFSTEP   18
      DHL(J) = DH = DHEND = ZHL(J)/(DX+0.5*DTNH*(U(J+1)-U(J)))          HAFSTEP   19
      GO TO 25                                                          HAFSTEP   20
20    DHL(J)=DH=DHEND=ZHL(J)/((X(J+1)+0.5*DTNH*U(J+1))**NALPHA-(X(J)+   HAFSTEP   21
     1  0.5*DTNH*U(J))**NALPHA)                                         HAFSTEP   22
25    IF (NPR(M) .EQ. 7) GO TO 200                                      HAFSTEP   23
      NSC=MAX1(1.,100.*ABS((DHEND-DOLD))/(DHEND+DOLD))                  HAFSTEP   24
      NSC=MINO(NSC,10)                                                  HAFSTEP   25
      DDH = (DHEND-DOLD)/NSC                                            HAFSTEP   26
      SSC=0.  $  IF (NSPEC .NE. 0 .AND. SDURM .LT. 1.) SSC=SSCALH(J)/NSC HAFSTEP  27
      IF (NSC .EQ. 1) GO TO 50                                          HAFSTEP   28
      PRINT 1030,NSC,J,N                                                HAFSTEP   29
1030  FORMAT (* SUBCYCLING IN HAFSTEP, NSC=*I3,* FOR J=*I3,*, N=*I5)    HAFSTEP   30
      DTNS = DTN                                                        HAFSTEP   31
      DTNHS = DTNH                                                      HAFSTEP   32
      DTNH = DTNH/NSC                                                   HAFSTEP   33
50    DO 120 NS = 1,NSC                                                 HAFSTEP   34
      DHL(J) = DH = DOLD+DDH                                            HAFSTEP   35
      HDV=0.5*(1./DOLD-1./DH)                                           HAFSTEP   36
      RHOLD=SHL(J)+FAC*(RHL(J)-SHL(J))                                  HAFSTEP   37
      EH=EOLD+HDV*FAC*(2.*RHOLD+DPDD(J)*(DH-DOLD)+DPDE(J)*SSC)+SSC      HAFSTEP   38
      IF (NALPHA-2) 70,60,65                                            HAFSTEP   39
C       CYLINDRICAL CASE                                                HAFSTEP   40
60    EZ=(SHL(J)-PHL(J)-SDT(J))*(-2.*HDV-(U(J+1)-U(J))*(DTNH+DTN)/      HAFSTEP   41
     1  (X(J+1)-X(J))/(DH+DOLD))                                        HAFSTEP   42
      EH=EH+EZ   $   GO TO 70                                           HAFSTEP   43
C       SPHERICAL CASE                                                  HAFSTEP   44
65    EZ=1.5*(SHL(J)-PHL(J))*(2.*HDV-(U(J+1)-U(J))*(DTNH+DTN)/(X(J+1)-  HAFSTEP   45
     1  X(J))/(DH+DOLD))                                                HAFSTEP   46
```

SUBROUTINE HAFSTEP (Concluded)

```
      EH=EH+EZ
70    CONTINUE
C               CALL HSTRESS TO COMPUTE STRESS VARIABLES
      CALL HSTRESS
      RHL(J)=RH
      EHL(J) = EH
      IF (NPR(M) .EQ. 1 .OR. NPR(M) .EQ. 4) GO TO 90
      RHL(J)=RH=(RH+DPDE(J)*(EOLD+SSC-EH+FAC*HDV*RHOLD))/(1.-HDV*DPDE(J)
     1  *FAC)
      EHL(J)=EH=(RHOLD+SHL(J)+FAC*(RH-SHL(J)))*HDV+EOLD+SSC
      IF (NALPHA .GT. 1) EHL(J)=EH=EH+EZ
      IF (NPR(M) .NE. 3) GO TO 90
      DPDDA=EQSTG(M)*EH+EQSTC(M)/RHO(M)
      IF (DH .EQ. DOLD) GO TO 80
      DPDD(J)=(RH-RHOLD-DPDE(J)*SSC)/(DH-DOLD)
80    IF (DPDD(J) .LE. 0. .OR. DPDD(J) .GT. 1.5*DPDDA) DPDD(J)=DPDDA
90    IF (NSC .EQ. 1) GO TO 140
      DOLD = DH   $  EOLD = EH
      DTN = DTNH
120   CONTINUE
      DTN = DTNS
      DTNH = DTNHS
140   CONTINUE
      RETURN
200   IF (NSPEC .NE. 0 .AND. SDURM .LT. 1.) EHL(J)=SSCALH(J)+EHL(J)
      DEPS=(DHL(J)-RHO(M))/DHL(J)
      SDH=MU(M)*DEPS
      QH=0.   $   DUH=U(J+1)-U(J)    $    CEF=CHL(J)-DUH/2
      IF (DUH .GT. 0.) GO TO 220
      CS=CHL(J)-DUH/2.   $   CF=CWQ(M)-CZQ(M)*DUH/CS
      QH=-0.5*(CWQ(M)*CS-CZQ(M)*DUH)*DUH*(DH+DOLD)
      CEF=CS*(1.+CF*(1.+0.5*CF))
      GO TO 230
220   QH=-0.5*C2(M)*CHL(J)*DUH*(DH+DOLD)
230   PHL(J)=DEPS*(EQSTC(M)+DEPS*(EQSTD(M)+DEPS*EQSTS(M)))*(1.-0.5*
     1  EQSTG(M)*(DHL(J)/RHO(M)-1.))+RHOS(M)*EQSTG(M)*EHL(J)
      SHL(J)=PHL(J)+SDH   $   RH=RHL(J)=SHL(J)+QH
      RETURN
      END
```

SUBROUTINE HDATA

```
      SUBROUTINE HDATA(H)
      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

```
      SUBROUTINE HSTRESS                                                HSTRESS    2
C                                                                       HSTRESS    3
C     THIS ROUTINE CONTROLS SWITCHING BETWEEN EQUATIONS OF STATE        HSTRESS    4
C     COMPUTES R,S,P FOR THE HALFSTEP POINT BETWEEN J AND J+1           HSTRESS    5
C     INPUT - J, M, DOLD, EOLD, UH, DH, EH.                             HSTRESS    6
C     OUTPUT - RH, SHL, PHL, YHL, H, C.                                 HSTRESS    7
C                                                                       HSTRESS    8
      INTEGER H,POROUS,PRESS,RINTER,SOLID,SPALL                         PUFCOM     2
      REAL MATL,NEM,NET,NEMH,NETH                                       PUFCOM     3
C             MISCELLANEOUS                                             PUFCOM     4
      COMMON AZERO(1),CEF,CKS,DAVG,DELTIM,DISCPT(10),DOLD,DRHO,DTMAX,   PUFCOM     5
     1  DTMIN,DTN,DTNH,DU,DX,EOLD,F,FAC,FIRST,J,JCYCS,JINIT,            PUFCOM     6
     2  JFIN,JREZON(15),JSMAX,JSTAR,JTS,LSUB(30),M,MAXPR(30),N,NCYCS,   PUFCOM     7
     3  NEDIT,NPERN,NR,NREZON,NSCRB(6),NSEPRAT,NSPALL,NTEDT,            PUFCOM     8
     4  NTEX,NTR(15),POLD,P6(20),R(30),RLAST,SLAST,SMAX,TEDIT(50),      PUFCOM     9
     5  TF,TIME,TJ,TREZON,TS,T6(20),ULAST,UOLD,UZERO,XLAST,XNOW,XOLD    PUFCOM    10
     1  ,XJDIT(20),MS                                                   PUFCOM    11
C             HALFSTEP VALUES                                           PUFCOM    12
      COMMON DH,DHLAST,DUH,EH,PH,RH,RHLAST,SH,SHLAST,UH,UHLAST,XH,XHLAST PUFCOM    13
     1  ,NEMH,NETH                                                      PUFCOM    14
C             CONDITION INDICATORS                                      PUFCOM    15
      COMMON INF,LINTER,MIRROR,NORMAL,POROUS,PRESS,RINTER,SOLID,SPALL   PUFCOM    16
C             CELL LAYOUT                                               PUFCOM    17
      COMMON DXX(30),JBND(30),JMAT(30),NAUTO,MATL(6,2),NLAYER,NMTRLS,   PUFCOM    18
     1  THK(30)                                                         PUFCOM    19
C                                                                       PUFCOM    20
C             COORDINATE ARRAYS                                         COORDCOM   2
      COMMON/COORD/X(200),XO(200),CHL(200),DHL(200),DPDD(200),DPDE(200), COORDCOM   3
     1  EHL(200),H(200,3),NEM(200),NET(200),PHL(200),RHL(200),SDT(200), COORDCOM   4
     2  SHL(200),T(200),U(200),YHL(200),ZHL(200)                        COORDCOM   5
C             NAMED COMMON                                              EQSTCOM    2
      REAL MU,MUM                                                       EQSTCOM    3
      COMMON /EQS/  EQSTA(6),EQSTC(6),EQSTD(6),EQSTE(6),EQSTG(6),       EQSTCOM    4
     1  EQSTH(6),EQSTN(6),EQSTS(6),EQSTV(6),CZQ(6),CWQ(6),C2(6)         EQSTCOM    5
      COMMON /MELT/ EMELT(6,8),GMELT(6,8),SPH(6),THERM(6,8)             EQSTCOM    6
      COMMON /RHO/ RHO(6),RHOS(6)                                       EQSTCOM    7
      COMMON /TSR/ TSR(6,30),EXMAT(6,20),TENS(6,3)                      EQSTCOM    8
      COMMON /Y/ YO(6),YADD(6),MU(6),MUM,YADDM                          EQSTCOM    9
      COMMON /IND/ IEOS(6),INDK(20),NALPHA,NCMP(6),NFR(6),NPOR(6),      INDCOM     2
     1  NDS(6),NPR(6),NCON(6),NVAR(6)                                   INDCOM     3
      COMMON /RAD/ SSTOP(9),START(9),SDURM,SSTOPM,NSPEC,SSJ,JSS,IPLOT(4) RADCOM     2
     1  ,XMAX(4),XMIN(4),YMAX(4),YMIN(4),IA(7),ITITLE(24),NARZ,TARZ     RADCOM     3
      COMMON /PES/ LVMAX,LVTOT,LVAR(200),COM(4000)                      HSTRESS   14
      COMMON /ESC/ ESC(6,20)                                            HSTRESS   15
      DATA MM/0/                                                        HSTRESS   16
C                                                                       HSTRESS   17
C             ABORT FOR NEGATIVE DENSITY                                HSTRESS   18
      IF (DH .GT. 0.) GO TO 25                                          HSTRESS   19
      WRITE (6,4990) N,J,DH,TIME   $   LSUB(7)=1   $  RETURN            HSTRESS   20
C         COMPUTE THERMAL STRENGTH REDUCTION AND OLD DEVIATOR STRESS    HSTRESS   21
   25 F=1.                                                              HSTRESS   22
      IF (N .EQ. 0) GO TO 30                                            HSTRESS   23
      IF (MM .EQ. 1000*M+N) GO TO 30                                    HSTRESS   24
      IF (THERM(M,1) .EQ. 0.) GO TO 27                                  HSTRESS   25
      EMELT(M,1)=THERM(M,6)+(THERM(M,1)-THERM(M,6))*EXP((-TIME+0.5*DTNH) HSTRESS   26
     1  /THERM(M,2))                                                    HSTRESS   27
   27 MM=1000*M+N                                                       HSTRESS   28
      IF (THERM(M,3) .EQ. 0.) GO TO 30                                  HSTRESS   29
      EQSTE(M)=THERM(M,8)+(THERM(M,3)-THERM(M,8))*EXP((-TIME+0.5*DTNH)  HSTRESS   30
     1  /THERM(M,4))                                                    HSTRESS   31
   30 IF (EH .LT. EMELT(M)) IE=1                                        HSTRESS   32
      IF (H(J,3) .EQ. 5R    M .AND. IE .EQ. 1) H(J,3)=5R    E           HSTRESS   33
      IF (EMELT(M,1) .GT. 1.E20) GO TO 34                               HSTRESS   34
      IF (THERM(M,1) .EQ. 0.) GO TO 33                                  HSTRESS   35
      CALL TMELT(0,M,EH,F,FG)                                           HSTRESS   36
      IF (GMELT(M,1) .NE. 0.) CALL TMELT(1,M,EH,F,FG)                   HSTRESS   37
      GO TO 34                                                          HSTRESS   38
   33 CALL FMELT(4,M,EH,F,FG,X)                                         HSTRESS   39
   34 IF (F .EQ. 0.) H(J,3)=5R    M                                     HSTRESS   40
      MUM=MU(M)*FG                                                      HSTRESS   41
      IF (EXMAT(M,4).NE. 0.) MUM=(MU(M)+(DH-RHOS(M))*EXMAT(M,4))*FG     HSTRESS   42
      T(J)=TENS(M,1)*F                                                  HSTRESS   43
      YADDM=YADD(M)                                                     HSTRESS   44
      CZJ=CZQ(M)                                                        HSTRESS   45
      CWJ=CWQ(M)                                                        HSTRESS   46
```

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.
      EXY=0.
      EV=-2.*(DH-DOLD)/(DH+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 (NALPHA .EQ. 1) GO TO 45
      XOS=XO(J+1)+XO(J)
      EYEN=(X(J+1)+X(J)-XOS)/XOS
 45   CONTINUE
      GO TO (50,60,70)NALPHA
C     PLANAR GEOMETRY
  50  EX=EV
      EY=0.
      EZ=0.
      SY=SDH/2.
      SZ=SY
      GO TO 80
C     CYLINDRICAL GEOMETRY
  60  EX=DUH*DT/DX
      EY=EV-EX
      EZ=0.
      SY=-SDT(J)
      SZ=-(SX+SY)
      GO TO 80
C     SPHERICAL GEOMETRY
  70  EX=DUH*DT/DX
      EY=(EV-EX)/2.
      EZ=EY
      SY=SDH/2.
      SZ=SY
      EZEN=EYEN
  80  CONTINUE
      IF (NSCRB(6) .EQ. 0 .OR. MOD(N,NEDIT) .NE. 1) GO TO 100
      TCX=PHL(J)-SX  $  TCY=PHL(J)-SY  $  TCZ=PHL(J)-SZ
      ECXEN=-EXEN  $  ECYEN=-EYEN  $  ECZEN=-EZEN
      PRINT 81,N,J,TCX,TCY,TCZ,PHL(J),ECXEN,ECYEN,ECZEN
 81   FORMAT(* N,J=*2I4,* TCX,TCY,TCZ,PHL=*1P4E10.3,* ENG. STRAINS
     1  ECX,ECY,ECZ=*3E10.3)
C
C        ****  ROUTES FOR COMPOSITE, POROUS, AND FRACTURE MODELS  ****
C
 100  IF (NCMP(M)+NFR(M)+NPOR(M) .EQ. 0) GO TO 200
      IF (NCMP(M) .EQ. 0) GO TO 130
C
C              ROUTE FOR COMPOSITE MODEL
C
C                  --  REBAR  --
      L=LVAR(J)
      CALL REBAR(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),NET(J),
     2  YHL(J),COM(L+2),0)
      SDH=-SX
      SDT(J)=-SY
      GO TO 400
C              ROUTE FOR POROUS MODEL
130   IF (NPOR(M) .EQ. 0) GO TO 160
      IF(NPOR(M) .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
```

SUBROUTINE HSTRESS (Continued)

```
  135 CONTINUE
      NPORM=NPOR(M)
      GO TO (140,145,150,155) NPORM
C
C                        --  POREQST  --
  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  YADDM,NDS(M),NPR(M),J)
      GO TO 310
  145 CONTINUE
C
C                        --  PORHOLT  --
      CALL PORHOLT(1,5,M,CHL(J),DH,DOLD,EH,EOLD,F,PHL(J),H(J,1),J,
     1     DPDE(J),EQSTC(M),MUM,YADDM,RHOS(M),DT)
      GO TO 310
  150 NPRM = NPR(M)+1
C
C                        --  PEST  --
      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),EQSTS(M),
     3  EQSTG(M),MUM,YADDM,COM(L+3),COM(L+4),CZJ,
     4  CWJ,EQSTH(M),EQSTE(M),EQSTN(M),EQSTV(M),
     5  EQSTA(M),DPDD(J),DPDE(J),N)
      MUM= 0.75 * MUM
      GO TO 300
  155 SX=SX-PHL(J)
      SY=SY-PHL(J)
      SZ=SZ-PHL(J)
C
C                        --  CAP1  --
C     SX,SY,SZ ARE TOTAL STRESSES POSITIVE IN TENSION
  157 CALL CAP1(LSUB(8), IN,M,N,H(J,1),DH,DOLD,EH,EX,EY,EZ,EXY,F,
     1     EQSTG(M),RHOS(M),SX,SY,SZ,TXY,NEM(J),K,J,NET(J))
      PHL(J)=-(SX+SY+SZ)/3.
      SDH=-SX-PHL(J)
      SDT(J)=-SY-PHL(J)
      GO TO 400
C                ROUTES FOR FRACTURE MODELS
  160 NFRM=NFR(M)+1
      GO TO(200,170,175,180,180,195)NFRM
C
C                        --  DFRACT  --
  170 IF(PHL(J) .GT. TSR(M,5) .AND. H(J,3) .LT. 3)GO TO 200
      CALL DFRACT(SX,SY,SZ,TXY,EX,EY,EZ,EXY,PHL(J),NEM(J),NET(J),
     1  DH,DOLD,DT,EOLD,EH,EQSTC(M),EQSTG(M),MUM,RHOS(M),TSR,YHL(J),
     2  YADDM,F,M,J,K,DROT)
      SHL(J)=-SX+PHL(J)
      SDT(J)=-SY
      H(J,3)=3
      GO TO 410
C
C                        --  FRAG  --
  175 IF (H(J,3) .GE. 3) GO TO 177
        STENS=AMAX1(SX,SY,SZ)
      IF (-STENS+PHL(J) .GT. TSR(M,5)) GO TO 200
      H(J,3)=3
      LVAR(J)=LVMAX
      LVMAX=LVMAX+NVAR(M)
      IF (LVMAX .LE. LVTOT+1) GO TO 177
      LSUB(7)=1
      PRINT 1177,N,J,TIME
 1177 FORMAT(* FRAG EXCEEDED STORAGE AT N=*I4,* J=*I3,* TIME=*1PE10.3)
  177 LS=1
      IF (MOD(N,NEDIT) .EQ. 0) LS=2
      L=LVAR(J)
      CALL FRAG(LS,5,M,J,J,N,H(J,3),EQSTC(M),DH,DOLD,DT,EH,EOLD,EX,EY,
     1  EXY,F,NEM(J),MUM,EQSTG(M),RHOS(M),ROT,DROT,PHL(J),SX,SY,TXY,
     2  YHL(J),EXMAT(M,1),TSR,COM(L),COM(L+5),COM(L+10),COM(L+15),
     3  COM(L+20))
      SDH=-SX
      SDT(J)=-SY
      LSUB(12)=1
```

```
      GO TO 400                                                         HSTRESS  197
C                                                                       HSTRESS  198
C                      --  SHEAR1  --                                   HSTRESS  199
  180 IF(H(J,3)-2)177,181,183                                           HSTRESS  200
  181 IF(NFR(M) .EQ. 3)GO TO 183                                        HSTRESS  201
      STENS=AMAX1(SX,SY,SZ)                                             HSTRESS  202
      IF(-STENS .LT. TSR(M,5)*TSR(M,9) .AND. -STENS+PHL(J) .LT. TSR(M,8) HSTRESS  203
     1  )GO TO 177                                                      HSTRESS  204
  183 LS=2                                                              HSTRESS  205
      IF (MOD(N,NEDIT) .EQ. 0) LS=3                                     HSTRESS  206
      L=LVAR(J)                                                         HSTRESS  207
      CALL SHEAR2(LS,5,M,J,J,H(J,3),SX,SY,SZ,TXY,PHL(J),COM(L),DH,DOLD, HSTRESS  208
     1  DT,EH,EOLD,COM(L+1),EMELT(M,1),COM(L+2),EX,EY,EZ,EXY,F,YHL(J),  HSTRESS  209
     2  COM(L+3),ROT,ROT,ESC,COM(L+4))                                  HSTRESS  210
      SDH=-SX                                                           HSTRESS  211
      SDT(J)=-SY                                                        HSTRESS  212
      GO TO 400                                                         HSTRESS  213
C                                                                       HSTRESS  214
C                      --  BFRACT  --                                   HSTRESS  215
  195 IF(H(J,3) .NE. 2)GO TO 197                                        HSTRESS  216
      STENS=AMAX1(SX,SY,SZ)                                             HSTRESS  217
      IF (N .EQ. 0) GO TO 200                                           9/12/79   12
      IF (-STENS .GT. TSR(M,5)*TSR(M,9) .OR. -STENS+PHL(J) .GT. TSR(M,8) HSTRESS  218
     1  )GO TO 200                                                      HSTRESS  219
      H(J,3)=1                                                          HSTRESS  220
197   SY=-SY                                                            HSTRESS  224
      SZ=-SZ                                                            HSTRESS  225
      LS=LSUB(12)                                                       HSTRESS  226
      IF (LS .NE. 0 .AND. MOD(N,NEDIT) .EQ. 0) LS=2                     HSTRESS  227
      L=LVAR(J)                                                         HSTRESS  228
      CALL BFRACT(LS,SDH,SY,SZ,TXY,-EX,-EY,-EZ,-EXY,PHL(J),NEM(J),NET(J) HSTRESS  229
     1  ,RHOS(M)/DH,RHOS(M)/DOLD,DT,EOLD,EH,EQSTC(M),EQSTG(M),MUM,TSR,  HSTRESS  230
     2  YHL(J),YADDM,F,1,J,M,N,RHOS(M),DROT,ROT,COM(L),COM(L+1),COM(L+6) HSTRESS  231
     3  )                                                               HSTRESS  232
      SDT(J)=-SY                                                        HSTRESS  233
  199 LSUB(12)=1                                                        HSTRESS  234
      GO TO 400                                                         HSTRESS  235
C                                                                       HSTRESS  236
C         ****  ROUTES FOR PRESSURE CALCULATION  ****                   HSTRESS  237
C                                                                       HSTRESS  238
200   NPRM = NPR(M)+1                                                   HSTRESS  239
      GO TO (270,220,230,240,250,255,260,270) NPRM                      HSTRESS  240
C             EQUATION OF STATE FOR EXPLOSION PRODUCTS                  HSTRESS  241
220   IF (NEM(J) .GE. 0.999999) GO TO 270                               HSTRESS  242
      QH=0.  $  IF(DUHM .LT. -1.) QH=(CZJ*DUHM-CWJ*CHL(J))*DUHM*DH      HSTRESS  243
      L=LVAR(J)                                                         HSTRESS  244
      CALL EXPLODE(3,5,M,EHL,DHL,DOLD,PHL,SHL,NEM,X,J,QH,TIME,DT)       HSTRESS  245
      EH=EHL(J)                                                         HSTRESS  246
      GO TO 305                                                         HSTRESS  247
C             SIMPLE, EXTENDED EQUATION OF STATE                        HSTRESS  248
230   CALL ESA(1,5,M,CHL(J),DH,EH,PHL(J),DPDD(J),DPDE(J))               HSTRESS  249
      GO TO 300                                                         HSTRESS  250
C             PHILCO-FORD EQUATION OF STATE                             HSTRESS  251
240   CALL EQSTPF(1,5,M,CHL(J),DH,EH,PHL(J))                            HSTRESS  252
      GO TO 300                                                         HSTRESS  253
C             VARIABLE MODULI EQN. OF STATE                             HSTRESS  254
C         (IMPLEMENTED FOR PLANAR AND SPHERICAL CASES ONLY)             HSTRESS  255
250   EPS=EMU=ALOG(DH/RHO(M))                                           HSTRESS  256
      IF (NALPHA .NE. 3) GO TO 252                                      HSTRESS  257
      L=LVAR(J)                                                         HSTRESS  258
      IF (COM(L) .EQ. 0.) COM(L)=X(J)                                   HSTRESS  259
      EPS=EMU+3.*ALOG((X(J)+U(J)*DTNH/2.)/COM(L))                       HSTRESS  260
  252 NEM(J)=EMU                                                        HSTRESS  261
      NET(J)=EPS                                                        HSTRESS  262
      CALL HYPO(1,IN,M,CHL(J),DH,EMU,COM(L+1),EPS,COM(L+2),J,PHL(J),SDH) HSTRESS  263
      GO TO 400                                                         HSTRESS  264
C             LLL*S 3-PHASE EQUATION OF STATE OF ROYCE                  HSTRESS  265
255   CALL GRAY(1,IN,M,MUM,EMELT(M,1),DH,EH,CHL(J),PHL(J),DPDD(J),      HSTRESS  266
     1  DPDE(J),H(J,1))                                                 HSTRESS  267
      GO TO 305                                                         HSTRESS  268
C             SANDIA TABULAR EQUATION OF STATE                          HSTRESS  269
260   CALL EOSTAB(1,IN,DH,EH,PHL(J))                                    HSTRESS  270
      GO TO 300                                                         HSTRESS  271
C             MIE-GRUNEISEN AND PUFF EXPANSION EQUATIONS OF STATE       HSTRESS  272
270   CALL EQST(EH,DH,PHL(J),M,CHL(J),DPDD(J),DPDE(J))                  HSTRESS  273
```

```
C
C        ****  ROUTES FOR DEVIATOR STRESS CALCULATION  ****
C
  300 IF (MUM .GT. 0. .AND. YHL(J)*F .GT. 0. .AND. NPR(M) .NE. 1) GO TO
     1  310
C                MATERIAL IS MELTED OR HOTTER - NO DEVIATOR STRESS
 3D5  SDH=0.   $   GO TO 4DD
 31D  IF (NDS(M) .GT. D) GO TO 32D
C                COULOMB-MISES YIELD WITH WORK HARDENING
      IF (NALPHA .GT. 1) GO TO 312
      SDH=SDH-1.333*MUM*EV
      GO TO 318
  312 SDH=SDH-2.0*MUM*(EX-EV/3.)
      IF (NALPHA .NE. 2) GO TO 318
      SDT(J)=SDT(J)-1.333*MUM*(EV-1.5*EX)
      SN=SQRT(3.*(SDH*SDH+SDT(J)*SDT(J)+SDH*SDT(J)))
      IF (SN .LT. (YHL(J)+EXMAT(M,1)*PHL(J))*F) GO TO 400
      YHL(J)=AMIN1(AMAX1(SN,YHL(J)),YHL(J)+YADDM*ABS(DH-DOLD))
      EL=(YHL(J)+EXMAT(M,1)*PHL(J))*F/SN
      SDH=EL*SDH   $   SDT(J)=EL*SDT(J)   $   GO TO 400
 318  CONTINUE
      IF (ABS(SDH) .LT. 0.6667*(YHL(J)+EXMAT(M,1)*PHL(J))*F) GO TO 400
      YHL(J)=AMIN1(AMAX1(ABS(1.5*SDH),YHL(J)),YHL(J)+YADDM*ABS(DH-DOLD))
      SDH=SIGN(0.6667*(YHL(J)+EXMAT(M,1)*PHL(J))*F,SDH)
      GO TO 400
C                PREPARE FOR COMPLEX YIELD MODELS
  320 DRHO=DH-DOLD
      OMUM=MUM
      COEF=-2.0*MUM*(EX-EV/3.)
      IF (NALPHA .EQ. 2) GO TO 323
      DRHO=COEF*(DH+DOLD)/2.667/MUM
      GO TO 325
  323 DSR=SDH+COEF
      DST=SDT(J)-1.333*MUM*(EV-1.5*EX)
      SNE=0.66667*SQRT(3.*(DSR*DSR+DST*DST+DSR*DST))
      SNO=0.66667*SQRT(3.*(SDH*SDH+SDT(J)*SDT(J)+SDT(J)*SDH))
      COEF=SNE-SNO
      DRHO=COEF*(DH+DOLD)/2.667/MUM
      SDHO=SDH   $   SDH=SNO
 325  CONTINUE
      NDSM=NDS(M)
      GO TO (330,340,340,330,350,360,370) NDSM
C                ONE- AND TWO-PARAMETER RELAXATION MODELS (NDS=1, 4)
 330  CALL RELAX(H(J,3),SDH,YHL(J),DRHO,COEF,N,J,M,NEM(J),NET(J),DT,
     1  TSR,YADDM,YO(M),NDSM)
      GO TO 390
C                BAND AND GILMAN RELAXATION MODELS (NDS=2, 3)
 340  CALL BANDRLX(H(J,3),SDH,YHL(J),DRHO,COEF,N,J,M,NEM(J),NET(J),DT,
     1  TSR,MUM,YADDM,NDSM)
      GO TO 390
C                BAUSCHINGER EFFECT MODEL  (NDS=5)
 350  MUM=(MU(M)+TSR(M,19)*(DH/RHO(M)-1.))*F
      COEF=2.*MUM*DRHO/(DH+DOLD)
      CALL BAUSCHI(H(J,3),SDH,COEF,NEM(J),YHL(J),NET(J),TSR(M,15),TSR(M,
     1  16),TSR(M,17),TSR(M,18),MUM)
      CSQ=CHL(J)**2+AMAX1(0.,(MUM-OMUM)*2./(DH+DOLD))
      CHL(J)=0.5*(CSQ/CHL(J)+CHL(J))
      GO TO 390
C                READ RELAXATION MODEL FOR BERYLLIUM  (NDS=6)
 360  CALL STRES2(LSUB(13),0,H(J,3),M,J,N,DH,DOLD,RHOS(M),SDH,MUM,F,DT,
     1   NEM(J),NET(J),TSR)
      GO TO 390
C     NONLINEAR WORK-HARDENING.
  370 SZ=-SZ
      CALL EP(1,M,N,SDH,SDT(J),SZ,TXY,YHL(J),-EX,-EY,-EZ,-EXY,MUM,
     1  NEM(J))
C
C        ****  ARTIFICIAL VISCOSITY AND RESULTANT STRESS  ****
C
 39D  IF (NALPHA .NE. 2) GO TO 4DD
C        ADJUSTMENTS FOR DEVIATORS IN CYLINDRICAL CASE
      EL=SDH/SNE   $   SDH=EL*DSR   $   SDT(J)=EL*DST
 400  SHL(J)=PHL(J)+SDH
      IF (H(J,1) .EQ. SOLID .AND. (DH /RHOS(M)-1.) .GT. D.)SHL(J) =
     1  PHL(J)+SDH*(1.-0.5*EQSTG(M)*(DH/RHOS(M)-1.)/AMAX1(D.01,F))
```

SUBROUTINE HSTRESS (Concluded)

```
  410 RH=SHL(J)  $  CEF=CHL(J)
      IF (DUHM .GE. -1.) GO TO 450
      CF=CWJ-CZJ*DUHM/CHL(J)
      CEF=CHL(J)*(1.+CF*(1.+0.5*CF))-DUH/2.
      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.LE.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.T(J))GO TO 530
      GO TO 550
  515 SHL(J)=PHL(J)=RH=0.
      GO TO 535
C     SPALL BY LATERAL STRESS,NALPHA=1,3
  520 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
      GO TO 535
C     SPALL BY THETA STRESS,NALPHA=2
  525 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
      SDT(J)=-PHL(J)
      GO TO 535
C     SPALL BY Z STRESS, NALPHA=2
  530 RF=(2.*PHL(J)-SDT(J)-SHL(J))/(EQSTC(M)+1.333*MUM     )
      DP=EQSTC(M)*RF
      DS=MUM*RF
      Q=RH-SHL(J)
      PHL(J)=PHL(J)-DP
      SHL(J)=SHL(J)-DP+DS/2.
      SDT(J)=SDT(J)+DS/2.
      RH=SHL(J)+Q
  535 CONTINUE
      PRINT 4992,J,N,SHL(J),PHL(J),RH,Q,SDT(J),DP,DS,SHLSV,PHLSV
 4992 FORMAT(* ---J=*I3,* N=*I3,* SHL,PHL,RH,Q=*1P4E11.3,* SDT,DP,DS=*
     1  1P3E11.3/* SHLSV,PHLSV=*1P2E11.3)
      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
  550 CONTINUE
      RETURN
 4990 FORMAT(20H STOP IN HSTRESS, N=I4,4H, J=I4,4H, D=1PE10.3,
     1  7H, TIME=1PE10.3)
      END
```

SUBROUTINE HYDRO

```
      SUBROUTINE HYDRO                                                  HYDRO        2
C                                                                       HYDRO        3
C     SUBROUTINE CONTROLS THE MAIN CALCULATION CYCLE                    HYDRO        4
C        *  CONTAINS 6 PATHS -                                          HYDRO        5
C              1.  NORMAL - COORDINATES WITHIN MATERIAL                 HYDRO        6
C              2.  INTERFACE - INTERFACE BETWEEN MATERIALS              HYDRO        7
C              3.  INTERFACE SPALL - SEPARATED INTERFACE BETWEEN MATERIA HYDRO       8
C              4.  MIRROR - FIRST COORDINATE FOR A SYMMETRIC IMPACT     HYDRO        9
C              5.  PRESSURE - PRESSURE HISTORY APPLIED AT FRONT (J=1)   HYDRO       10
C              6.  LEFT INTERFACE - DUMMY PATH                          HYDRO       11
C        *  CALLS HAFSTEP FOR HALFSTEP CALCULATIONS AT EACH COORDINATE  HYDRO       12
C        *  CHECKS FOR SPALLING AND RECOMBINATION                       HYDRO       13
C        *  COMPUTES MINIMUM PERMITTED TIME STEP FOR NEXT CYCLE         HYDRO       14
C     INPUT - DTNH,DTN,FIRST,NCYCS.                                     HYDRO       15
C                                                                       HYDRO       16
C              NAMED COMMON                                             EQSTCOM      2
      REAL MU,MUM                                                       EQSTCOM      3
      COMMON /EQS/  EQSTA(6),EQSTC(6),EQSTD(6),EQSTE(6),EQSTG(6),       EQSTCOM      4
     1  EQSTH(6),EQSTN(6),EQSTS(6),EQSTV(6),CZQ(6),CWQ(6),C2(6)         EQSTCOM      5
      COMMON /MELT/ EMELT(6,8),GMELT(6,8),SPH(6),THERM(6,8)             EQSTCOM      6
      COMMON /RHO/ RHO(6),RHOS(6)                                       EQSTCOM      7
      COMMON /TSR/ TSR(6,30),EXMAT(6,20),TENS(6,3)                      EQSTCOM      8
      COMMON /Y/ YO(6),YADD(6),MU(6),MUM,YADDM                          EQSTCOM      9
      COMMON /IND/ IEOS(6),INDK(20),NALPHA,NCMP(6),NFR(6),NPOR(6),      INDCOM       2
     1  NDS(6),NPR(6),NCON(6),NVAR(6)                                   INDCOM       3
      COMMON /RAD/ SSTOP(9),START(9),SDURM,SSTOPM,NSPEC,SSJ,JSS,IPLOT(4) RADCOM      2
     1  ,XMAX(4),XMIN(4),YMAX(4),YMIN(4),IA(7),ITITLE(24),NARZ,TARZ     RADCOM       3
C                                                                       HYDRO       20
      INTEGER H,POROUS,PRESS,RINTER,SOLID,SPALL                         PUFCOM       2
      REAL MATL,NEM,NET,NEMH,NETH                                       PUFCOM       3
C              MISCELLANEOUS                                            PUFCOM       4
      COMMON AZERO(1),CEF,CKS,DAVG,DELTIM,DISCPT(10),DOLD,DRHO,DTMAX,   PUFCOM       5
     1  DTMIN,DTN,DTNH,DU,DX,EOLD,F,FAC,FIRST,J,JCYCS,JINIT,            PUFCOM       6
     2  JFIN,JREZON(15),JSMAX,JSTAR,JTS,LSUB(30),M,MAXPR(30),N,NCYCS,   PUFCOM       7
     3  NEDIT,NPERN,NR,NREZON,NSCRB(6),NSEPRAT,NSPALL,NTEDT,            PUFCOM       8
     4  NTEX,NTR(15),POLD,P6(20),R(30),RLAST,SLAST,SMAX,TEDIT(50),      PUFCOM       9
     5  TF,TIME,TJ,TREZON,TS,T6(20),ULAST,UOLD,UZERO,XLAST,XNOW,XOLD    PUFCOM      10
     1  ,XJDIT(20),MS                                                   PUFCOM      11
C              HALFSTEP VALUES                                          PUFCOM      12
      COMMON DH,DHLAST,DUH,EH,PH,RH,RHLAST,SH,SHLAST,UH,UHLAST,XH,XHLAST PUFCOM     13
     1  ,NEMH,NETH                                                      PUFCOM      14
C              CONDITION INDICATORS                                     PUFCOM      15
      COMMON INF,LINTER,MIRROR,NORMAL,POROUS,PRESS,RINTER,SOLID,SPALL   PUFCOM      16
C              CELL LAYOUT                                              PUFCOM      17
      COMMON DXX(30),JBND(30),JMAT(30),NAUTO,MATL(6,2),NLAYER,NMTRLS,   PUFCOM      18
     1  THK(30)                                                         PUFCOM      19
C                                                                       PUFCOM      20
C              COORDINATE ARRAYS                                        COORDCOM     2
      COMMON/COORD/X(200),XO(200),CHL(200),DHL(200),DPDD(200),DPDE(200), COORDCOM    3
     1  EHL(200),H(200,3),NEM(200),NET(200),PHL(200),RHL(200),SDT(200), COORDCOM     4
     2  SHL(200),T(200),U(200),YHL(200),ZHL(200)                        COORDCOM     5
C                                                                       HYDRO       23
      IF (N .EQ. 1) ISPALL=0                                            HYDRO       24
 1    DT=DTMIN=1.                                                       HYDRO       25
      SMAX=0.                                                           HYDRO       26
C        ****  ROUTINE TO RESET DTNH FOR SPALL CLOSURE   *****          HYDRO       27
      IF (NLAYER .LE. 1 .OR. ISPALL .EQ. 0) GO TO 82                    HYDRO       28
      NLM1=NLAYER-1                                                     HYDRO       29
      DO 80 LLL=1,NLM1                                                  HYDRO       30
      JB=JBND(LLL)                                                      HYDRO       31
      IF (JB .GT. JSTAR) GO TO 82                                       HYDRO       32
      IF (H(JB+1,2) .NE. SPALL) GO TO 80                                HYDRO       33
      IF (U(JB) .EQ. U(JB+1)) GO TO 80                                  HYDRO       34
      DTSP=(X(JB+1)-X(JB))/(U(JB)-U(JB+1))                              HYDRO       35
      IF (DTSP .GT. DTNH .OR. DTSP .LT. 0.) GO TO 80                    HYDRO       36
      DTNAT=AMIN1((X(JB)-X(JB-1))/CHL(JB-1),(X(JB+2)-X(JB+1))/CHL(JB+1)) HYDRO      37
      DTNH=AMIN1(DTNH,AMAX1(DTSP,0.2*DTNAT))-0.001*DTNAT                HYDRO       38
      DTMIN=0.2*DTNAT                                                   HYDRO       39
      NCYCS=1                                                           HYDRO       40
80    CONTINUE                                                          HYDRO       41
82    CONTINUE                                                          HYDRO       42
C                                                                       HYDRO       43
C              OUTER HYDRO LOOP                                         HYDRO       44
      DO 1000 NN=1,NCYCS                                                HYDRO       45
      TIME=TIME+DTNH                                                    HYDRO       46
```

```
      ISPALL = 0                                                          HYDRO
      FAC=FLOAT(MINO(N-1,20))/20.                                         HYDRO
      LL=0                                                                HYDRO
      DO 900 J=JINIT,JFIN                                                 HYDRO
C                CHECK FOR THE APPROPRIATE PATH                           HYDRO
10    XOLD=X(J)  $  UOLD=U(J)                                             HYDRO
      IF (H(J,2) .EQ. NORMAL) GO TO 100                                   HYDRO
      IF (H(J,2) .EQ. LINTER) GO TO 900                                   HYDRO
      IF (H(J,2) .EQ. RINTER) GO TO 200                                   HYDRO
      IF (H(J,2) .EQ. SPALL ) GO TO 300                                   HYDRO
      IF (H(J,2) .EQ. MIRROR) GO TO 500                                   HYDRO
      IF (H(J,2) .EQ. PRESS ) GO TO 600                                   HYDRO
      IF (H(J,2) .EQ. INF   ) GO TO 700                                   HYDRO
C                                                                         HYDRO
C*********       NORMAL PATH WITHIN A MATERIAL                          ** HYDRO
C                                                                         HYDRO
100   IF (NSPEC.GT.0 .OR. ABS(U(J)-U(J+1)).GT.0.001 .OR. EHL(J).GT.1.     HYDRO
     1 .OR. ABS(RHLAST).GT.1.  .OR. NPR(M).EQ.1) GO TO 102                HYDRO
101   UH=U(J)  $  RH=RHL(J)  $  XH=.5*(X(J+1)+X(J)+DTNH*U(J))             HYDRO
      DH=DHL(J)  $  EH=EHL(J)                                             HYDRO
      X(J)=X(J)+DTNH*U(J)  $  GO TO 800                                   HYDRO
102   CALL HAFSTEP                                                        HYDRO
C                VELOCITY CALCULATION                                     HYDRO
120   IF (NALPHA-2) 125,130,135                                           HYDRO
C         PLANAR CASE                                                     HYDRO
125   U(J)=UOLD-2.*DTNH*(RH-RHLAST)/(ZHL(J-1)+ZHL(J))                     HYDRO
      GO TO 140                                                           HYDRO
C         CYLINDRICAL CASE                                                HYDRO
130   XBAR1=SQRT((X(J-1)**2+X(J)**2)/2.)                                  HYDRO
      XBAR2=SQRT((X(J)**2+X(J+1)**2)/2.)                                  HYDRO
      U(J)=U(J)+4.*DTNH*(RHLAST*XBAR1-RH*XBAR2+(PHL(J-1)+SDT(J-1))*       HYDRO
     1   (X(J)-XBAR1)+(PHL(J)+SDT(J))*(XBAR2-X(J)))/(ZHL(J)+ZHL(J-1))     HYDRO
      GO TO 140                                                           HYDRO
C         SPHERICAL CASE                                                  HYDRO
135   XBAR1 = ((X(J-1)**3+X(J)**3)/2.)**(1./3.)                           HYDRO
      XBAR2 = ((X(J)**3+X(J+1)**3)/2.)**(1./3.)                           HYDRO
      U(J) = UOLD-2.*DTNH*(RH*XBAR2**2-RHLAST*XBAR1**2+0.5*(SHL(J-1)-3.*  HYDRO
     1  PHL(J-1))*(XBAR1+X(J))*(X(J)-XBAR1)+0.5*(SHL(J)-3.*PHL(J))        HYDRO
     1  *(XBAR2+X(J))*(XBAR2-X(J)))/(ZHL(J-1)+ZHL(J))*3.                  HYDRO
C                COORDINATE CALCULATION                                   HYDRO
140   X(J)=X(J)+0.5*DTNH*(U(J)+UOLD)                                      HYDRO
      DT=(X(J+1)+U(J+1)*DTNH-X(J))/CEF                                    HYDRO
      GO TO 800                                                           HYDRO
C                                                                         HYDRO
C*********       INTERFACE                                              ** HYDRO
C                                                                         HYDRO
C     LEFT VALUES ARE IN (J-1) CELLS AND RIGHT VALUES ARE IN (J) CELLS    HYDRO
200   IF (X(J-1) .LT. X(J)) GO TO 300                                     HYDRO
      MLAST=M  $   LL=LL+1   $   M=JMAT(LL)                               HYDRO
      CALL HAFSTEP                                                        HYDRO
C           CHECK STRESS AND SET INDICATORS FOR SPALL                     HYDRO
      J1=J-1                                                              HYDRO
      IF (R(LL) .GT. T(J1)) GO TO 205                                     HYDRO
      H(J,2)=SPALL  $  R(LL)=T(J1)=0.  $  ISPALL=ISPALL+1                 HYDRO
      IF (MAXPR(9) .LE. 0)  GO TO 205                                     HYDRO
      PRINT 5230, N,NN,LL,TIME                                            HYDRO
205   J1=J-1  $  J2=J-2                                                   HYDRO
      ULOLD=U(J1)                                                         HYDRO
      IF (NALPHA-2) 210,212,215                                           HYDRO
C         PLANAR CASE                                                     HYDRO
210   U(J)=(U(J1)*ZHL(J2)+U(J)*ZHL(J)-2.*DTNH*(RH-RHLAST))/               HYDRO
     1  (ZHL(J)+ZHL(J2))                                                  HYDRO
      GO TO 218                                                           HYDRO
C         CYLINDRICAL CASE                                                HYDRO
212   XBAR1=SQRT((X(J2)**2+X(J)**2)/2.)                                   HYDRO
      XBAR2=SQRT((X(J)**2+X(J+1)**2)/2.)                                  HYDRO
      U(J)=(U(J1)*ZHL(J2)+U(J)*ZHL(J)+4.*DTNH*(RHLAST*XBAR1-RH*XBAR2+     HYDRO
     1  (PHL(J2)+SDT(J2))*(X(J)-XBAR1)+(PHL(J)+SDT(J))*(XBAR2-X(J))))/    HYDRO
     2  (ZHL(J)+ZHL(J2))                                                  HYDRO
      GO TO 218                                                           HYDRO
C         SPHERICAL CASE                                                  HYDRO
215   XBAR1 = ((X(J2)**3+X(J)**3)/2.)**(1./3.)                            HYDRO
      XBAR2 = ((X(J)**3+X(J+1)**3)/2.)**(1./3.)                           HYDRO
      U(J) = (U(J1)*ZHL(J2)/3.+U(J)*ZHL(J)/3.-2.*DTNH                     HYDRO
     1  *(RH*XBAR2**2-RHLAST*XBAR1**2+0.5*(SHL(J2)-3.*PHL(J2))*           HYDRO
```

```
     2  (XBAR1+X(J))*(X(J)-XBAR1)+0.5*(SHL(J)-3.*PHL(J))*(XBAR2+X(J))
     3  *(XBAR2-X(J))))/(ZHL(J)+ZHL(J2))*3.
218   U(J1)=U(J)
      X(J)=X(J1)=X(J)+.5*DTNH*(U(J)+UOLD)
      R(LL)=(RH*ZHL(J2)+RHLAST*ZHL(J)+D.5*(ULOLD-UOLD)*ZHL(J2)*
     1  ZHL(J)/DTNH)/(ZHL(J)+ZHL(J2))
      DT=(X(J+1)+U(J+1)*DTNH-X(J))/CEF
22D   CONTINUE
      GO TO 8DD
C
C**********         INTERFACE SPALL                                      **
C
30D   IF (J .EQ. JINIT) GO TO 330
      MLAST=M
C                   LEFT SIDE
      J1=J-1
      XLOLD=X(J-1)  $  ULOLD=U(J-1)
      IF (NALPHA-2) 305,310,315
C        PLANAR CASE
305   U(J-1)=ULOLD+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(J-1)=U(J-1)+4.*DTNH*(RHLAST*XBAR1+(PHL(J-2)+SDT(J-2))*
     1  (X(J-1)-XBAR1))/(ZHL(J-2))
      GO TO 320
C        SPHERICAL CASE
315   XBAR1 = ((X(J-2)**3+X(J-1)**3)/2.)**(1./3.)
      U(J-1) = ULOLD+2.*DTNH*(RHLAST*XBAR1**2-D.5*(SHL(J-2)-3.*PHL(J-2)
     1  )*(XBAR1+X(J-1))*(X(J-1)-XBAR1))/ZHL(J-2)*3.
320   X(J-1)=XLOLD+0.5*DTNH*(U(J-1)+ULOLD)
      DT=1.
      IF (J .EQ. JFIN) GO TO 9DD
C                   RIGHT SIDE
330   LL=LL+1  $  M=JMAT(LL)
      R(LL)=0.
      IF (NSPEC.GT.0 .OR. ABS(U(J)-U(J+1)).GT.0.001 .OR. EHL(J).GT.1.
     1 .OR. ABS(RHLAST).GT.1. .OR. NPR(M).EQ.1) GO TO 332
331   UH=U(J)  $  RH=RHL(J)  $  XH=.5*(X(J+1)+X(J)+DTNH*U(J))
      X(J)=X(J)+DTNH*U(J)  $  DT=1.
      IF (NALPHA .GT. 1 .AND. J .EQ. JINIT) X(J)=AMAX1(XOLD,D.)
      IF (NALPHA .GT. 1 .AND. J .EQ. JINIT .AND. X(J) .EQ. D.) U(J)=D.
      DH=DHL(J)  $  EH=EHL(J)
      IF (J .EQ. JINIT) 80D,335
332   CALL HAFSTEP
      IF (RHL(J) .GT. T(J)) GO TO 334
      RH=RHL(J)=SHL(J)=PHL(J)=T(J)=0.
334   UOLD=U(J)  $  XOLD=X(J)
      IF (NALPHA - 2) 3341,3342,3343
C        PLANAR CASE
3341  U(J)=UOLD-2.*DTNH*RH/ZHL(J)
      GO TO 3344
C        CYLINDRICAL CASE
3342  XBAR2=SQRT((X(J)**2+X(J+1)**2)/2.)
      U(J)=UOLD+4.*DTNH*(-RH*XBAR2+(PHL(J)+SDT(J))*(XBAR2-X(J)))/ZHL(J)
      GO TO 3344
C        SPHERICAL CASE
3343  XBAR2 = ((X(J)**3+X(J+1)**3)/2.)**(1./3.)
      U(J) = UOLD-2.*DTNH*(RH*XBAR2**2+0.5*(SHL(J)-3.*PHL(J))*
     1  (XBAR2+X(J))*(XBAR2-X(J)))/ZHL(J)*3.
3344   X(J)=XOLD+D.5*DTNH*(U(J)+UOLD)
      IF (NALPHA .GT. 1 .AND. J .EQ. JINIT) X(J)=AMAX1(XOLD,0.)
      IF (NALPHA .GT. 1 .AND. J .EQ. JINIT .AND. X(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
      ISPALL=ISPALL+1  $  GO TO 8D0
C                   RESET ARRAY VARIABLES AND GO TO INTERFACE ROUTE
365   H(J,2)=RINTER  $  X(J)=XOLD  $  X(J-1)=XLOLD  $  U(J)=UOLD
      PRINT 1365,N,J,TIME
      U(J-1)=ULOLD
      DT=D.1*AMIN1(DT,DTP)
      IF (DT .LT. 0.) DT=1.
      IF (DT .GT. DTMIN) GO TO 205
```

```
      DTMIN=DT  $  JTS=J  $  GO TO 205
C
C*********     MIRROR AT FRONT SURFACE
C
500   LL=LL+1   $   M=JMAT(LL)
      IF (J .GE. JFIN-1) GO TO 800
      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 800
      H(J,2)=SPALL
      R(LL)=T(JFIN-1)=0.
      GO TO 800
C
C*********     PRESSURE BOUNDARY AT FRONT SURFACE                        **
C
  600 LL=LL+1
      IF (J .EQ. JFIN) GO TO 650
      M=JMAT(LL)
      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-DTNH)
      CALL HAFSTEP
      RHAF=SIGMAT(1,TIME-0.5*DTNH)
 603  CONTINUE
      IF (NALPHA-2) 605,610,615
C        PLANAR CASE
605   U(J)=UOLD-2.*(RH-RHAF)/ZHL(J)*DTNH
      GO TO 620
C        CYLINDRICAL CASE
610   XBAR2=SQRT((X(J)**2+X(J+1)**2)/2.)
      U(J)=U(J)+4.*DTNH*(RHAF*X(J)-RH*XBAR2+(PHL(J)+SDT(J))*(XBAR2-X(J))
     1  )/ZHL(J)
      GO TO 620
C        SPHERICAL CASE
615   XBAR2=((X(J)**3+X(J+1)**3)/2.)**(1./3.)
      U(J)=U(J)+2.*DTNH*(RHAF*X(J)**2-RH*XBAR2**2+0.5*(SHL(J)-3.*PHL(J))
     1  *(XBAR2+X(J))*(XBAR2-X(J)))/ZHL(J)*3.
620   X(J)=X(J)+0.5*DTNH*(U(J)+UOLD)
      DT=(X(J+1)+DTNH*U(J+1)-X(J))/CEF
      GO TO 800
C
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
 652  R(LL)=SIGMAT(2,TIME-DTNH)
      RHAF=SIGMAT(2,TIME-0.5*DTNH)
 654  CONTINUE
      UOLD=U(J-1)
      IF (NALPHA-2) 660,665,670
C        PLANAR CASE
660   U(J-1)=UOLD+2.*(RHLAST-RHAF)/ZHL(J-2)*DTNH
      GO TO 675
C        CYLINDRICAL CASE
665   XBAR1=SQRT((X(J-2)**2+X(J-1)**2)/2.)
      U(J-1)=U(J-1)+4.*DTNH*(RHLAST*XBAR1-RHAF*X(J-1)+(PHL(J-2)+SDT(J-2)
     1  )*(X(J-1)-XBAR1))/ZHL(J-2)
      GO TO 675
C        SPHERICAL CASE
670   XBAR1=((X(J-2)**3+X(J-1)**3)/2.)**(1./3.)
      U(J-1)=U(J-1)+2.*DTNH*(RHLAST*XBAR1**2-RHAF*X(J-1)**2+0.5*(SHL(J-2
     1)-3.*PHL(J-2))*(XBAR1+X(J-1))*(XBAR1-X(J-1)))/ZHL(J-2)*3.
675   X(J)=X(J-1)=X(J-1)+0.5*DTNH*(U(J-1)+UOLD)
      U(J)=U(J-1)    $    DT=(X(J-1)+U(J-1)*DTNH-X(J-2))/CHL(J-2)
      GO TO 800
C
C*********     INFINITE BOUNDARY AT FRONT SURFACE.
C
```

SUBROUTINE HYDRO (Concluded)

```
  700 IF (J .EQ. JFIN) GO TO 720
      LL=LL+1  $  M=JMAT(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=SQRT(RHOS(M)*(EQSTC(M)+1.333*MU(M)))*(U(JFIN+1)-UOLD)
      DP=EQSTC(M)/(EQSTC(M)+1.333*MU(M))*DS
      SDH=SHL(JFIN+1)-PHL(JFIN+1)+DS-DP
      PHL(JFIN+1)=PHL(JFIN+1)+DP
      U(JFIN+1)=UOLD
      IF (ABS(SDH) .GT. 0.6667* YHL(J))  SDH=SIGN(0.6667*YHL(J),SDH)
      SHL(JFIN+1)=RHLAST=SDH+PHL(JFIN+1)
      R(LL)=RHLAST
      GO TO 100
C
C*********     INFINITE BOUNDARY AT REAR SURFACE.
C
  720 LL=LL+1
      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
      UOLD=U(J)=U(J-1)
      IF (ABS(SDH) .GT. 0.6667* YHL(J-1))SDH=SIGN(0.6667*YHL(J-1),SDH)
      SHL(J-1)=RH=RHL(J-1)=SDH+PHL(J-1)
      R(LL)=RH
      U(J-1)=UOLD-DTNH*(RH-RHLAST)/ZHL(J-2)
      X(J-1)=X(J-1)+0.5*DTNH*(UOLD+U(J-1))
C
C*********     END OF HYDRO PATHS
C
C
  800 CONTINUE
      IF (LSUB(7) .EQ. 1)  RETURN
C******        END OF CYCLE RESET
      XLAST=XOLD  $  ULAST=UOLD
      XHLAST=XH  $  UHLAST=UH  $  RHLAST=RH  $  DHLAST=DH
      EHLAST=EH  $  SHLAST=SH
C
C              SMAX CALCULATION
      IF (SHL(J) .GT. SMAX)  820,822
  820 SMAX=SHL(J)  $  JSMAX=J
C
C              TIME STEP CALCULATION
  822 IF (DT .LT. 0.) DT=1.
      IF (DT .GT. DTMIN) GO TO 826
  824 DTMIN=DT  $  JTS=J
  826 DTP=DT
C
C              JSTAR CALCULATION
  850 IF (ABS(U(J)) .LT. 1.E-3 .AND. EHL(J) .LT. 1.) 851,900
  851 IF (J .GT. JSTAR) 852,900
  852 JSTAR=J-1
      GO TO 990
C              END OF HYDRO INNER LOOP
  900 CONTINUE
      JSTAR=JFIN-1
  990 DTN=DTNH
      JTS=JTS+1000*ISPALL
C              END OF HYDRO OUTER LOOP
 1000 CONTINUE
 1002 RETURN
 1365 FORMAT(*  RECOMBINATION AT CYCLE *I4,*, J=*I4,*, TIME=*1PE10.3)
 5115 FORMAT (* SPALL AT N, NN=*2I4,* FOR J=*I4,*, NSPALL=*I4,*, TIME=*
     1  1PE10.3)
 5230 FORMAT (* INTERFACE SPALL AT N, NN =*2I4,* ON LEFT OF LAYER *I2,
     1  *, TIME=*1PE10.3)
      END
```

SUBROUTINE HYPO

```
      SUBROUTINE HYPO(INDE,IN,M,CJ,DH,EMU,EMUO,EPS,EPSO,J,P,SDH)
C
C     THIRD VERSION OF VARIABLE MODULUS MODEL INCLUDING AN INTEGRAL
C     DEFINITION OF LOADING SURFACES FOR P AND SDH AND DIFFERENTIAL
C     RELATIONS ONLY FOR UNLOADING
C
C     ROUTINE IS WRITTEN FOR 1-DIMENSIONAL PLANAR AND SPHERICAL FLOW.
C     FOR 1-D CYLINDRICAL FLOW,SDH IS INTERPRETED AS 2/3XEFFECTIVE
C     STRESS AND RADIAL AND TANGENTIAL DEVIATOR STRESSES ARE COMPUTED
C     FROM SDH IN HSTRESS.
C
C     SUBROUTINE COMPUTES PRESSURE P AND DEVIATOR STRESS SDH AND
C     SOUND SPEED CJ.
C
C     SUBROUTINE IS CALLED TWICE
C     INDE=0 CALLED FROM GENRAT
C          READ MATERIAL PROPERTY DATA AND INITIALIZE VARIABLES
C     INDE=1 CALLED FROM HSTRESS
C          COMPUTE PRESSURE, DEVIATOR STRESS, AND SOUND SPEED.
      REAL K0,K1,K2,K0UN,K1UN,K2UN,KY
      DIMENSION G0(6),G1(6),G2(6),GU(6),K0(6),K1(6),K2(6),K0UN(6),
     1K1UN(6),K2UN(6),KY(6),EMUY(6)
      IF(INDE.GT.0)GO TO 200
C
C     ***READ AND INITIALIZE***
C
      READ(IN,1000)A1,G0(M),G1(M),G2(M),GU(M),KY(M),EMUY(M)
      WRITE(6,1000)A1,G0(M),G1(M),G2(M),GU(M),KY(M),EMUY(M)
      READ(IN,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*G0(M)
      G1(M)=1.333*G1(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
C
C     ***COMPUTATION OF STRESS***
C
200   DMU=EMU-EMUO
      DEPS=EPS-EPSO
C     DEVIATOR STRESS
      SDH=AMIN1(SDH+GU(M)*DEPS,EPS*(G0(M)+G1(M)*EMU+G2(M)*EPS))
C     PRESSURE
      IF(EMU.LT.EMUY(M))GO TO 220
      BULK=K0UN(M)+K1UN(M)*(EMU+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),KY(M)*EMU)
      BULK=KY(M)
300   EMUO=EMU
      EPSO=EPS
C     SOUND SPEED
      CJ=SQRT((BULK+GU(M))/DH)
      RETURN
 1000 FORMAT(A10,1P7E10.3)
      END
```

SUBROUTINE PEST

```
      SUBROUTINE PEST(LS,IN,NPRM,H,J,TJ,DT,M,C,D,DOLD,RHOS,RHOI,P,PST1,
     1  AST1,E,EOLD,F,EQSTCM,EQSTDM,EQSTSM,EQSTGM,MUM,YADDM,RVV,ENT,CZJ,
     2  CWJ,EQSTHM,EQSTEM,EQSTNM,EQSTVM,EQSTAM,DPDDJ,DPDEJ,NCYC)
C
C PEST 2, VERSION OF DEC 1976
C WRITTEN AT STANFORD RESEARCH INSTITUTE BY L. SEAMAN AND R.E. TOKHEIM
C CODE PROVIDES EQUATIONS OF STATE FOR POROUS AND SOLID MATERIALS
C UNDER COMPRESSION(C), TENSION(T) AND RECOMPRESSION(R) BY RATE-
C INDEPENDENT AND RATE-DEPENDENT MODELS. INITIALIZATION FOR ALL MODELS
C IS INCLUDED.
C
C INDICATORS OF MODELS TO BE CHOSEN FOR STATIC(S) AND DYNAMIC(D)
C CONDITIONS FOLLOW:
C KCS OR KRS: RATE-INDEPENDENT COMPRESSION
C      1 POREQST
C      2 PORHOLT
C      3 CARROLL-HOLT
C      4 HERRMANN P-ALPHA
C      5 HENDRON
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      2 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      2 N.A.G. DUCTILE FRACTURE
C      3 BRITTLE FRACTURE AND FRAGMENTATION
C
C INDICATORS(X) ARE READ IN THREE-DIGIT PAIRS FOR S AND D CONDITIONS:
C       KCS,KTS,KRS= 0X0X0X    KCD,KTD,KRD= 0X0X0X
C
C INDICATORS H AND IH
C     S SOLID
C     P POROUS-PRESSURE
C     T POROUS-TENSION
C     Q POROUS-RECOMPRESSION
C     Z FRAGMENTATION
C     R RECOMPRESSION AFTER FRAGMENTATION
C
      INTEGER H,OUT
      REAL MUM,MUP,K1C
      DIMENSION KCS(4),KCD(4),KTS(4),KTD(4),KRS(4),KRD(4)
      DIMENSION NPM(6),NREG(4)
      DIMENSION TPH(4,3),DADP(4,3),K1C(4),TEMP(8)
      DIMENSION AK(4),MUP(4),YADDP(4,5,3),TER(4,8,3)
      DIMENSION RHOP(4,6,3),COSQ(4,5,3),C1(4,5,3)
      DIMENSION PORA(4,5,3),PORB(4,5,3),PORC(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-/,JQ2/
     1  10H -POREQST-/,JQ3/10H -CARROLL-/,JQ4/5HHOLT-/,JQ5/10H -HERRMANN
     2  /,JQ6/9H P-ALPHA-/,JQ7/10H -VARIA ST/,JQ8/7HRENGTH-/
     3  ,JQ9/10H -FRACTURE/,JQ10/6H MECH-/,JQ11/10H -LINEAR V/,JQ12/
     4  9HISC VOID-/,JQ13/10H -DYNAMIC /,JQ14/8HPORHOLT-/,JQ15/
     5  10H -PORHOLT-/,JQ16/8HBUTCHER-/,JQS/5HSTAT:/,JQR/5HRATE:/
     6  ,JQ17/10H -DUCTILE /,JQ18/9HFRACTURE-/
C                                                                      C
C ***     ZEROING OF ARRAYS                                        *** C
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   YADDP(I)=COSQ(I)=C1(I)=PORA(I)=PORB(I)=PORC(I)=0.
```

```
      DO 53 I=1,72
   53 RHOP(I)=0.
      DO 54 I=1,96
   54 TER(I)=0.
      MP=0  $  DPDDJ=DPDEJ=0.  $  LS=1  $  CJ=1.
 8    MP=MP+1
      NPM(M) = MP
C                                                                      C
C  ********************                          ******************** C
C        READING OF INPUT DATA                                         C
C  ********************                          ******************** C
C                                                                      C
C ***    READ DATA USED BY ALL MODELS.                             *** C
      READ(IN,935)A1,KCS(MP),KTS(MP),KRS(MP),A2,KCD(MP),KTD(MP)
     1  ,KRD(MP)
      WRITE(6,935)A1,KCS(MP),KTS(MP),KRS(MP),A2,KCD(MP),KTD(MP)
     1  ,KRD(MP)
      WRITE(6,960)IDD,IN,JQ1
C                                                                      C
      READ(IN,919)A1,AK(MP),A2,MUP(MP),A3,YZERO,A4,RHOP(MP,6,1)
      WRITE(6,920)A1,AK(MP),A2,MUP(MP),A3,YZERO,A4,RHOP(MP,6,1)
      WRITE(6,960)IDD,IN,JQ1
      ALFO=RHOS/RHOP(MP,6,1)
      IF(AK(MP) .GT. 0. .AND. AK(MP) .LE. EQSTCM*RHOP(MP,6,1)
     1  /RHOS) GO TO 20
      IF (AK(MP) .GT. 0.) GO TO 10
C  *     IF AK IS NEGATIVE, IT IS INTERPRETED AS THE SHEAR MODULUS    * C
C  *     OF THE SOLID.                                                * C
      GS = -AK(MP)
      AK(MP) = EQSTCM/(ALFO+0.75*EQSTCM/GS*(ALFO-1.))
      MUP(MP) = GS*(1.-5.*(1.-1./ALFO)*(3.*EQSTCM+4.*GS)/(9.*EQSTCM
     1  +8.*GS))
      GO TO 15
C  *     IF AK IS TOO LARGE, IT IS REDUCED TO THE MAXIMUM PERMITTED.  * C
   10 AK(MP)=EQSTCM*RHOP(MP,6,1)/RHOS
 15   WRITE(6,950) AK(MP), MUP(MP)
      WRITE(6,960)IDD,OUT,JQ1
 20   YADDM=0.666667*YZERO  $  MUP(MP)=1.333333*MUP(MP)
      C=SQRT((AK(MP)+AMAX1(0.,MUP(MP)))/AMIN1(D,RHOP(MP,6,1)))
      J2=5HCOMP,  $  J3=J4=1H
      N=1
      KCSM=KCS(MP)  $  KCDM=KCD(MP)  $  KTSM=KTS(MP)  $  KTDM=KTD(MP)
      KRSM=KRS(MP)  $  KRDM=KRD(MP)
      IF (KTSM .EQ. 0) J3=5HTENS,   $   IF (KRSM .EQ. 0) J4=5HRECOM
C                                                                      C
C ***    READ FOR RATE-INDEPENDENT COMPRESSIVE MODEL.              *** C
C                                                                      C
      GO TO (490,510,520,530,540,550)KCSM
490   CONTINUE
C                                                                      C
C **     READ AND INITIALIZE FOR POREQST.                           ** C
      READ(IN,939)A1,NREG(MP)
      WRITE(6,940)A1,NREG(MP)
      WRITE(6,960)IDD,IN,JQ2,IDD,JQS,J2,J3,J4
      READ(IN,909)A1,(RHOP(MP,I,N),I=1,5)
      WRITE(6,910)A1,(RHOP(MP,I,N),I=1,5)
      WRITE(6,960)IDD,IN,JQ2
      DO 498 I=1,5
      COSQ(MP,I,N) = 4.0
498   C1(MP,I,N) = 0.15
501   READ(IN,905)(TEMP(I),I=1,8)
      DECODE (3,915,TEMP) A1,A2
      IF (A1 .EQ. 1HC .AND. (A2 .EQ. 1H0 .OR. A2 .EQ. 1HO)) GO TO 502
      IF (A1 .EQ. 1HC .AND. A2 .EQ. 1H1) GO TO 503
      GO TO 504
502   DECODE (80,910,TEMP)A1,(COSQ(MP,I,N),I=1,5)
      WRITE(6,910)A1,(COSQ(MP,I,N),I=1,5)
      WRITE(6,960)IDD,IN,JQ2
      GO TO 501
503   DECODE (80,910,TEMP)A1,(C1(MP,I,N),I=1,5)
      WRITE(6,910)A1,(C1(MP,I,N),I=1,5)
      WRITE(6,960)IDD,IN,JQ2
      GO TO 501
504   CZJ = COSQ(MP,5,1)
      CWJ = C1(MP,5,1)
```

SUBROUTINE PEST (Continued)

```
      NP=NREG(MP)
      DECODE (80,920,TEMP)A1,P1
      WRITE(6,920)A1,P1
      WRITE(6,960)IDD,IN,JQ2
      PORA(MP,1,N) = P1  $  PORB(MP,1,N) = PORC(MP,1,N) = 0.
      DO 505 NQ=1,NP
      READ(IN,919)A1,P2,A2,DELP,A3,YADDP(MP,NQ,N)
      WRITE(6,920)A1,P2,A2,DELP,A3,YADDP(MP,NQ,N)
      WRITE(6,960)IDD,IN,JQ2
      IF (NQ .NE. NP) GO TO 5045
      IF (RHOP(MP,NP+1,N) .GT. RHOS) GO TO 5045
      RHOP(MP,NP+1,N) = RHOS*(1.+TSQE(0,P2,0.,EQSTCM,EQSTDM,EQSTSM,
     1  EQSTGM,EQSTHM,EQSTEM,RHOS,EQSTNM,0.,EQSTVM,EQSTAM,NCYC))
      WRITE(6,932)RHOP(MP,NP+1,N)
      WRITE(6,960)IDD,OUT,JQ2
 5045 DRHO=RHOP(MP,NQ+1,N)-RHOP(MP,NQ,N)
      AA=P2-P1-4.*DELP*RHOP(MP,NQ,N)/DRHO
      PORA(MP,NQ+1,N)=P1+RHOP(MP,NQ+1,N)/DRHO*AA
      BB=P2-P1-4.*DELP*(RHOP(MP,NQ+1,N)+RHOP(MP,NQ,N))/DRHO
      PORB(MP,NQ+1,N)=-RHOP(MP,NQ+1,N)*RHOP(MP,NQ,N)/DRHO*BB
      PORC(MP,NQ+1,N)=-4.*DELP*(RHOP(MP,NQ+1,N)*RHOP(MP,NQ,N)/DRHO)**2
      YADDP(MP,NQ,N) = YADDP(MP,NQ,N)/DRHO
  505 P1=P2
      YADDP(MP,NP+1,N) = 0.
      RHOP(MP,5,N) = RHOP(MP,NP+1,N)
      GO TO 600
  510 CONTINUE
C                                                                      C
C **     READ AND INITIALIZE FOR PORHOLT.                           ** C
      READ(IN,919)A1,RHOP(MP,1,N)
      WRITE(6,920)A1,RHOP(MP,1,N)
      READ(IN,919)A1,RHOP(MP,5,N),A2,DPDRHO,A3,PY,A4,YADDP(MP,1,N)
      WRITE(6,920)A1,RHOP(MP,5,N),A2,DPDRHO,A3,PY,A4,YADDP(MP,1,N)
      WRITE(6,960)IDD,IN,JQ15,IDD,JQS,J2,J3,J4
      IF (RHOP(MP,5,N) .LT. 100.) GO TO 512
      P2 = RHOP(MP,5,N)
      RHOP(MP,5,N)=RHOS*(1.+TSQE(0,P2,0.,EQSTCM,EQSTDM,EQSTSM,EQSTGM,
     1  EQSTHM,EQSTEM,RHOS,EQSTNM,0.,EQSTVM,EQSTAM,NCYC))
      WRITE(6,932)RHOP(MP,5,N)
      WRITE(6,960)IDD,OUT,JQ15
  512 RHOP(MP,2,N)=RHOP(MP,1,N)*(PY/AK(MP)+1.)
      RHOP(MP,3,N)=RHOS/(1.-RHOS*PY/RHOP(MP,2,N)/EQSTCM)
      ALFE=RHOP(MP,3,N)/RHOP(MP,2,N)
      R=RHOP(MP,3,N)-RHOS
      PORA(MP,1,N)=ALFE*(ALFE*RHOP(MP,2,N)/EQSTCM*DPDRHO-R/RHOS)
      R1=PORA(MP,1,N)/(RHOP(MP,5,N)-RHOP(MP,2,N))
      PORB(MP,1,N)=(RHOP(MP,5,N)-RHOP(MP,3,N))/
     1  (RHOP(MP,5,N)-RHOP(MP,2,N))**2-R1
      YADDP(MP,1,N) = YADDP(MP,1,N)/(RHOP(MP,5,N)-RHOP(MP,2,N))
      WRITE(6,930)
      IF (N .GE. 2) GO TO 640
      GO TO 600
  520 CONTINUE
C                                                                      C
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,RHOP(MP,1,N),A2,YCH,A3,EPS(MP,N),A4,TER(MP,7,N)
      WRITE(6,960)IDD,IN,JQ3,JQ4,JQS,J2,J3,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)
      RV=1.-RHOP(MP,1,N)/RHOS
C *      PY AND PC KNOWN                                             * C
      RHOP(MP,5,N)=RHOS*(1.+TSQE(0,P2,0.,EQSTCM,EQSTDM,EQSTSM,EQSTGM,
     1  EQSTHM,EQSTEM,RHOS,EQSTNM,0.,EQSTVM,EQSTAM,NCYC))
      BB = BBMIN = (RHOP(MP,5,N)/RHOS-1.)*EQSTCM/PY
      ALFA = 1./(1.-RV)
      DEL(MP,N) = PY/(EQSTCM*ALOG(1.-RHOP(MP,1,1)/RHOS))
      IF ( YCH .LT. 0.) BBMIN = AMIN1(BB,1./DEL(MP,N))
      BBMIN = AMAX1(BBMIN,0.24627*ALFA**2+2.8512*ALFA-1.9633)
      IF (BB .GT. BBMIN) GO TO 521
      BB = BBMIN
      RHOP(MP,5,N) = RHOS*(1.+BB*PY/EQSTCM)
      EO = 1./BB
```

```
      WRITE(6,927)                                                      PEST   227
      GO TO 5215                                                        PEST   228
521   E0 = RV**BB                                                       PEST   229
5215  B0 = ALOG(E0)/ALOG(RV+E0)                                         PEST   230
      E2= E1 = (RV+E0)**BB                                              PEST   231
      IF (ABS(E0-E1) .LT. 1.E-05*E1) GO TO 524                          PEST   232
      B1 = ALOG(E1)/ALOG(RV+E1)                                         PEST   233
      NW = 0                                                            PEST   234
522   NW = NW+1                                                         PEST   235
      E2 = E1*EXP((BB-B1)*(ALOG(RV+E1)/(1.-BB*E1/(RV+E1))))             PEST   236
      B2 = ALOG(E2)/ALOG(RV+E2)                                         PEST   237
      AW = NW                                                           PEST   238
      IF (ABS(B2-B1) .LT. 1.E-5 .OR. AW .GE. 10.) GO TO 524             PEST   239
      E0 = E1  $  B0 = B1  $  E1 = E2  $  B1 = B2                       PEST   240
      GO TO 522                                                         PEST   241
524   EPS(MP,N) = E2                                                    PEST   242
      DEL(MP,N) = (1.-RHOP(MP,5,N)/RHOS)/ALOG(EPS(MP,N))                PEST   243
      IF (BB .LE. BBMIN) GO TO 5275                                     PEST   244
      GO TO 528                                                         PEST   245
C *      YCH AND EPS KNOWN                                            * C PEST   246
525   DEL(MP,N) = 0.66667*YCH/EQSTCM                                    PEST   247
      IF (YCH .LT. 0.) EPS(MP,N) = AMAX1(EPS(MP,N),ABS(DEL(MP,N)))      PEST   248
      PY=-0.6666667*YCH*ALOG(1.-RHOP(MP,1,N)/RHOS+EPS(MP,N))            PEST   249
      GO TO 527                                                         PEST   250
C *      PY AND EPS KNOWN                                             * C PEST   251
 526  DEL(MP,N)=-PY/EQSTCM/ALOG(1.-RHOP(MP,1,N)/RHOS+EPS(MP,N))        PEST   252
      IF (YCH .LT. 0.) EPS(MP,N) = AMAX1(EPS(MP,N),ABS(DEL(MP,N)))      PEST   253
527   RHOP(MP,5,N) = RHOS*(1.-DEL(MP,N)*ALOG(EPS(MP,N)))                PEST   254
 5275 CALL EQST(0.,RHOP(MP,5,N),P2,M,1.,A1,A2)                          PEST   255
C *      ALL C-H                                                      * C PEST   256
528   ALE(MP,N) = DEL(MP,N)*ALOG(EPS(MP,N))                             PEST   257
      APC(MP,N) = RHOS/RHOP(MP,5,N)                                     PEST   258
      WRITE(6,925)PY,P2,EPS(MP,N)                                       PEST   259
      WRITE(6,960)IDD,OUT,JQ3,JQ4                                       PEST   260
      WRITE(6,932)RHOP(MP,5,N)                                          PEST   261
      WRITE(6,960)IDD,OUT,JQ3,JQ4                                       PEST   262
      EPS(MP,N) = 1.+EPS(MP,N)                                          PEST   263
      GO TO 600                                                         PEST   264
530   CONTINUE                                                          PEST   265
C                                                                     C PEST   266
C **     READ INPUT AND INIT FOR HERRMANN P-ALPHA.                   ** C PEST   267
      READ(IN,919)A1,PC,A3,PY                                           PEST   268
      WRITE(6,920)A1,RHOP(MP,1,1),A2,PC,A3,PY                           PEST   269
      WRITE(6,960)IDD,IN,JQ5,JQ6,JQS,J2,J3,J4                           PEST   270
      PORA(MP,1,N) = PY  $  PORC(MP,1,N) = PC                           PEST   271
      GO TO 600                                                         PEST   272
540   CONTINUE                                                          PEST   273
C                                                                     C PEST   274
C **     READ AND INIT FOR HENDRON.                                  ** C PEST   275
      GO TO 600                                                         PEST   276
550   CONTINUE                                                          PEST   277
C                                                                     C PEST   278
C **     READ AND INIT FOR TBS.                                      ** C PEST   279
      GO TO 600                                                         PEST   280
600   IF (N .GE. 2) GO TO 640                                           PEST   281
      N = 2                                                             PEST   282
      J2=5HTENS   $   J3=J4=1H                                          PEST   283
C                                                                     C PEST   284
C ***    READ FOR RATE-INDEPENDENT TENSION MODEL.                   *** C PEST   285
C                                                                     C PEST   286
      IF (KTSM .EQ. 0 .AND. KCSM .EQ. 3) GO TO 610                      PEST   287
      GO TO (615,620,520) KTSM                                          PEST   288
C                                                                     C PEST   289
C **     REPEAT CARROLL-HOLT ARRAY FOR N=2.                          ** C PEST   290
610   ALE(MP,2)=-ALE(MP,1)  $  EPS(MP,2)=EPS(MP,1)                      PEST   291
      DEL(MP,2)=-DEL(MP,1)  $  TER(MP,7,2) = TER(MP,7,1)                PEST   292
      APC(MP,2)=1./(1.-ALE(MP,N))                                       PEST   293
      RHOP(MP,5,N)=RHOS/APC(MP,2)                                       PEST   294
      WRITE(6,932)RHOP(MP,5,N)                                          PEST   295
      WRITE(6,960)IDD,OUT,JQ3,JQ4,JQS,J2                                PEST   296
      GO TO 600                                                         PEST   297
615   CONTINUE                                                          PEST   298
C                                                                     C PEST   299
C **     READ AND INIT FOR VARIABLE STRENGTH.                             PEST   300
      READ(IN,919)A1,TER(MP,5,N),A2,TER(MP,7,N)                         PEST   301
```

```
      WRITE(6,920)A1,TER(MP,5,N),A2,TER(MP,7,N)
      WRITE(6,960)IDD,IN,JQ7,JQ8,JQS,J2
      GO TO 600
620   CONTINUE
C                                                                     C
C **     READ AND INIT FOR K1C.                                    ** C
      READ(IN,919)A1,K1C(MP),A2,TER(MP,7,N)
      WRITE(6,920)A1,K1C(MP),A2,TER(MP,7,N)
      WRITE(6,960)IDD,IN,JQ9,JQ10,JQS,J2
      GO TO 600
640   CONTINUE
      IF (N .EQ. 3) GO TO 700
      N = 3
      J2=5HRECOM
C                                                                     C
C ***    READ FOR RATE-INDEPENDENT RECOMPRESSION MODEL.           *** C
C                                                                     C
      IF (KRSM .GT. 0) GO TO 660
C                                                                     C
C **     REPEAT ARRAYS KRS=KCS.                                    ** C
      GO TO (641,645,647,648) KCSM
C                                                                     C
C **     POREQST.                                                  ** C
641   NPP = NP+1
      DO 642 NQ = 1,NPP
      PORA(MP,NQ,3)=PORA(MP,NQ,1)  $  YADDP(MP,NQ,3)=YADDP(MP,NQ,1)
      PORB(MP,NQ,3)=PORB(MP,NQ,1)  $  PORC(MP,NQ,3)=PORC(MP,NQ,1)
642   CONTINUE
      DO 644 NQ=1,5
      RHOP(MP,NQ,3)=RHOP(MP,NQ,1)  $  COSQ(MP,NQ,3)=COSQ(MP,NQ,1)
      C1(MP,NQ,3)=C1(MP,NQ,1)
644   CONTINUE
      GO TO 700
C                                                                     C
C **     PORHOLT.                                                  ** C
645   PORA(MP,1,3)=PORA(MP,1,1)  $  PORB(MP,1,3)=PORB(MP,1,1)
      RHOP(MP,5,3) = RHOP(MP,5,1)  $  RHOP(MP,2,3) = RHOP(MP,2,1)
      RHOP(MP,3,3) = RHOP(MP,3,1)  $  YADDP(MP,1,3) = YADDP(MP,1,1)
      RHOP(MP,1,3)=RHOP(MP,1,1)
      GO TO 700
C                                                                     C
C **     CARROLL-HOLT MODEL.                                       ** C
647   APC(MP,3)=APC(MP,1)  $  EPS(MP,3)=EPS(MP,1)
      DEL(MP,3)=DEL(MP,1)  $  RHOP(MP,5,3)=RHOP(MP,5,1)
      RHOP(MP,1,3)=RHOP(MP,1,1)
      GO TO 700
C **     HERRMANN P-ALPHA MODEL.                                   ** C
648   PORA(MP,1,3) = PORA(MP,1,1)  $  PORC(MP,1,3) = PORC(MP,1,1)
      RHOP(MP,1,3) = RHOP(MP,1,1)
      GO TO 700
660   GO TO (490,510,520,530,540,550) KRSM
C                                                                     C
C ***    READ FOR RATE EFFECTS IN COMPRESSION.                    *** C
C                                                                     C
700   N = 1
      J2=5HCOMP,  $  J3=J4=1H
      IF (KTDM .EQ. 0) J3=5HTENS,  $  IF (KRDM .EQ. 0) J4=5HRECOM
      IF (KCDM .LE. 0) GO TO 750
      GO TO (750,720,730,740) KCDM
720   CONTINUE
C                                                                     C
C **     READ AND INIT FOR LINEAR VISCOUS VOID(C) OR DUCTILE FRACTURE(T)
      READ(IN,909)A1,(TER(MP,I,N),I=1,7)
      WRITE(6,910)A1,(TER(MP,I,N),I=1,7)
      IF (N .EQ. 1 .OR. N .EQ. 3) WRITE(6,960)IDD,IN,JQ11,JQ12,JQR,
     1  J2,J3,J4
      IF (N .EQ. 2) WRITE(6,960)IDD,IN,JQ17,JQ18,JQR,J2
      IF (TER(MP,8,N) .EQ. 0.)TER(MP,8,N)=8.*3.14159*TER
     1  (MP,3,N)**3*TER(MP,4,N)
      GO TO 750
730   CONTINUE
C                                                                     C
C **     READ AND INIT DYNAMIC PORHOLT.                            ** C
      READ(IN,919)A1,TPH(MP,N)
      WRITE(6,920)A1,TPH(MP,N)
```

SUBROUTINE PEST (Continued)

```
      WRITE(6,960)IDD,IN,JQ13,JQ14,JQR,J2,J3,J4                            PEST    377
      GO TO 750                                                            PEST    378
740   CONTINUE                                                             PEST    379
C                                                                        C PEST    380
C  *     READ AND INIT DYNAMIC BUTCHER P-ALPHA-TAU.                   ** C PEST    381
      READ(IN,919)A1,TPH(MP,N)                                             PEST    382
      WRITE(6,920)A1,TPH(MP,N)                                             PEST    383
      DADP(MP,N)=-ALFO/AK(MP)*(1.-AK(MP)*ALFO/EQSTCM)                      PEST    384
      WRITE(6,960)IDD,IN,JQ13,JQ16,JQR,J2,J3,J4                            PEST    385
750   N = N+1                                                              PEST    386
C                                                                        C PEST    387
C ***    READ FOR RATE EFFECTS IN TENSION.                           *** C PEST    388
C                                                                        C PEST    389
      GO TO(700,755,770,900) N                                             PEST    390
755   J2=5HTENS                                                            PEST    391
      IF (KTDM .GT. 0) GO TO (750,720,760) KTDM                            PEST    392
      IF (KCDM .EQ. 0) GO TO 750                                           PEST    393
C                                                                        C PEST    394
C **     REPEAT ARRAYS KTD=KCD.                                       ** C PEST    395
      IF (KCDM .EQ. 1) GO TO 750                                           PEST    396
      IF (KCDM .GT. 2) GO TO 756                                           PEST    397
C                                                                        C PEST    398
C **     REPEAT LINEAR VISCOUS VOID FOR DUCTILE FRACTURE.             ** C PEST    399
      TER(MP,1,2)=TER(MP,1,1)  $  TER(MP,2,2)=-TER(MP,2,1)                 PEST    400
      TER(MP,3,2)=TER(MP,3,1)  $  TER(MP,4,2)=TER(MP,4,1)                  PEST    401
      TER(MP,5,2)=-TER(MP,5,1)  $  TER(MP,6,2)=TER(MP,6,1)                 PEST    402
      TER(MP,7,2)  =  TER(MP,7,1) $  TER(MP,8,2) = TER(MP,8,1)             PEST    403
      GO TO 750                                                            PEST    404
C                                                                        C PEST    405
C **     READ BRITTLE FRACTURE AND FRAGMENTATION.                   ** C   PEST    406
 756  CONTINUE                                                             PEST    407
760   CONTINUE                                                             PEST    408
      GO TO 750                                                            PEST    409
C                                                                        C PEST    410
C ***    READ FOR RATE EFFECTS IN RECOMPRESSION.                     *** C PEST    411
770   J2=5HRECOM                                                           PEST    412
      IF (KRDM .GT. 0) GO TO 800                                           PEST    413
C                                                                        C PEST    414
C **     REPEAT ARRAYS KRD=KCD AS FOLLOWS.                            ** C PEST    415
      IF (KCDM .EQ. 0) GO TO 900                                           PEST    416
      GO TO (900,780,785,790) KCDM                                         PEST    417
C                                                                        C PEST    418
C **     REPEAT FOR LINEAR VISCOUS VOID COMPRESSION MODEL.            ** C PEST    419
780   TER(MP,1,3)=TER(MP,1,1)  $  TER(MP,2,3)=TER(MP,2,1)                  PEST    420
      TER(MP,3,3) = TER(MP,3,1)  $  TER(MP,4,3) = TER(MP,4,1)              PEST    421
      TER(MP,5,3) = TER(MP,5,1)  $  TER(MP,6,3) = TER(MP,6,1)              PEST    422
      TER(MP,7,3) = TER(MP,7,1)  $  TER(MP,8,3) = TER(MP,8,1)              PEST    423
      GO TO 900                                                            PEST    424
C                                                                        C PEST    425
C **     REPEAT FOR DYNAMIC PORHOLT MODEL.                            ** C PEST    426
785   TPH(MP,3)=TPH(MP,1)                                                  PEST    427
      GO TO 900                                                            PEST    428
C                                                                        C PEST    429
C **     REPEAT FOR BUTCHER P-ALPHA-TAU MODEL.                        ** C PEST    430
790   TPH(MP,3)=TPH(MP,1)  $  DADP(MP,3)=DADP(MP,1)                        PEST    431
      GO TO 900                                                            PEST    432
800   GO TO (900,720,730,740) KRDM                                         PEST    433
900   RETURN                                                               PEST    434
905   FORMAT(8A10)                                                         PEST    435
 909  FORMAT(A10,7E10.3)                                                   PEST    436
 910  FORMAT(A10,1P7E10.3)                                                 PEST    437
915   FORMAT(1X,2A1)                                                       PEST    438
 919  FORMAT(4(A10,E10.3))                                                 PEST    439
 920  FORMAT(4(A10,1PE10.3))                                               PEST    440
 925  FORMAT(* PY=*1PE10.3,* PC=*1PE10.3,* EPS=*1PE10.3)                   PEST    441
927   FORMAT(* ABSOLUTE VALUE OF CONSOLIDATION PRESSURE WAS CHANGED TO     PEST    442
     1 BE WITHIN ALLOWABLE RANGE*)                                         PEST    443
930   FORMAT(/)                                                            PEST    444
 932  FORMAT(* CONSOLIDATION DENSITY=*1PE10.3)                              PEST    445
935   FORMAT(2(A10,I6,I2,I2))                                              PEST    446
 939  FORMAT(A10,I10,A10,E10.3)                                            PEST    447
 940  FORMAT(A10,I10,A10,1PE10.3)                                          PEST    448
 950  FORMAT(* BULK AND SHEAR MODULI ARE CHANGED TO*1P2E12.3,* DYN/CM2*)   PEST    449
960   FORMAT(1H+,79X,5H IND=A2,5H, IN=I2,A10,A9,4A5)                       PEST    450
C                                                                        C PEST    451
```

```
C  ********************                                  ******************** C PEST     452
C        COMPUTATION OF PRESSURE DURING WAVE PROPAGATION.                     C PEST     453
C  ********************                                  ******************** C PEST     454
C                                                                             C PEST     455
1000  MP = NPM(M)                                                               PEST     456
      IH=H                                                                      PEST     457
C                                                                             C PEST     458
C **     COMPUTE BULK AND SHEAR MODULI APPROPRIATE TO CURRENT E AND D.        C PEST     459
C                                                                             C PEST     460
      TF = 1.+E*EQSTGM*RHOS/EQSTCM                                              PEST     461
      DREF = D*TF                                                               PEST     462
      RVV1=ABS(RVV)  $  ALFD1=1./(1.-RVV1)                                      PEST     463
      IF (RVV .LT.0. .AND. DREF/RHOS .LT. 1.-ABS(RVV)) GO TO 2000               PEST     464
      IF (NCYC .LE. 1) ALFS = RHOS/RHOP(MP,6,1)                                 9/12/79   13
      IF(F .EQ. 0.) GO TO 1800                                                  PEST     466
      IF (H .EQ. 5R    S .OR. H .EQ. 5R    M) GO TO 1800                        PEST     467
      RHOPV=RHOS/TF+(RHOP(MP,5,1)-RHOS)*F                                       PEST     468
      RHOM=RHOP(MP,6,1)/TF                                                      PEST     469
      ALF=AMAX1(1.0,RHOPV/D)  $  ALFZ=RHOPV/RHOM                                PEST     470
      ELK=(EQSTCM/AK(MP)-ALFZ)/(ALFZ-1.)                                        PEST     471
      ELG=(1.-MUP(MP)*F/MUM)/(1.-1./ALFZ)                                       PEST     472
      BULK=EQSTCM*F/(ALF+ELK*(ALF-1.))                                          PEST     473
      MUM=AMAX1(0.,MUM*(1.-ELG+ELG/ALF))                                        PEST     474
      C=SQRT((BULK+MUM)/D)                                                      PEST     475
      IF (NCYC .EQ. 0) PRINT 2300,D,BULK,MUM,C,F,ELK,ELG,RHOP(MP,6,1),          9/12/79   14
     1  E                                                                       9/12/79   15
C                                                                             C PEST     477
C ***    COMPUTE PRESSURE FROM ELASTIC RELATIONS.                         *** C PEST     478
      PEL=P+BULK*((D-DOLD)/(0.5*(D+DOLD))+EQSTGM*RHOS/EQSTCM*(E-EOLD))          PEST     479
C                                                                             C PEST     480
C **     BRANCH TO TENSILE OR COMPRESSIVE ROUTES.                          ** C PEST     481
C                                                                             C PEST     482
      IF (PEL .LT. 0.) GO TO 1500                                               PEST     483
C                                                                             C PEST     484
C ***    COMPRESSION PATH.                                                *** C PEST     485
C                                                                             C PEST     486
      KCRS=KCS(MP)  $  N=1                                                      PEST     487
      IF (H .EQ. 5R    T) H = 5R    Q                                           PEST     488
      IF (H .NE. 5R    Z .AND. H .NE. 5R    R) GO TO 1090                       PEST     489
      H = 5R    R                                                               PEST     490
      KCRS = KRS(MP)                                                            PEST     491
      IF (KRS(MP) .EQ. 0) KCRS = KCS(MP)                                        PEST     492
      N = 3                                                                     PEST     493
1090  GO TO (1100,1120,1140,1160,1180) KCRS                                     PEST     494
C                                                                             C PEST     495
C ***    CALCULATION OF COMPACTION CURVE.                                 *** C PEST     496
C                                                                             C PEST     497
C **     POREQST MODEL.                                                    ** C PEST     498
1100  NC = 0                                                                    PEST     499
      PST = 0.                                                                  PEST     500
      IF (DREF .GT. RHOP(MP,5,N))GO TO 1109                                     PEST     501
1105  NC = NC+1                                                                 PEST     502
      IF (DREF .GT. RHOP(MP,NC,N)) GO TO 1105                                   PEST     503
      PST = F*(PORA(MP,NC,N)+PORB(MP,NC,N)/DREF+PORC(MP,NC,N)/DREF**2)          PEST     504
      NQ = MAXO(1,NC-1)                                                         PEST     505
      CZJ = COSQ(MP,NQ,N)  $  CWJ = C1(MP,NQ,N)                                 PEST     506
      YADDM = YADDP(MP,NQ,N)                                                    PEST     507
C                                                                             C PEST     508
C  *     CHECK FOR CONSOLIDATION IN LAST POROUS REGION.                     * C PEST     509
1108  IF (DREF .LT. RHOS) GO TO 1300                                            PEST     510
1109  GO TO (1110,1112,1114) NPRM                                               PEST     511
1110  CALL EQST(E,D,PS,M,CJ,DPDDJ,DPDEJ)                                        PEST     512
      GO TO 1118                                                                PEST     513
1112  CALL ESA(1,5,M,CJ,D,E,PS,DPDDJ,DPDEJ)                                     PEST     514
      GO TO 1118                                                                PEST     515
1114  CALL EQSTPF(1,5,M,CJ,D,E,PS)                                              PEST     516
1118  IF (PS .LT. PST) GO TO 1300                                               PEST     517
      PST = PS                                                                  PEST     518
      IH = 5R    S                                                              PEST     519
      IF (PS .LT. PEL) GO TO 1300                                               PEST     520
      PJ = PS  $  H = 5R    S  $  RVV = 0.                                      PEST     521
      GO TO 1900                                                                PEST     522
C                                                                             C PEST     523
C **     PORHOLT MODEL.                                                    ** C PEST     524
1120  DREF=AMAX1(DREF,RHOP(MP,1,N))                                             PEST     525
```

SUBROUTINE PEST (Continued)

```
      ALFS = (RHOP(MP,3,N)+(PORA(MP,1,N)+PORB(MP,1,N)*(DREF-RHOP(MP,2,N)
     1))*(DREF-RHOP(MP,2,N)))/DREF
      ALFS=AMAX1(ALFS,1.)
      DS = ALFS*DREF
      GO TO (1126,1128,1130) NPRM
 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)
 1134 PST = PS/ALFS*F
      YADDM = YADDP(MP,1,N)
      GO TO 1108
C                                                                     C
C **      CARROLL-HOLT MODEL.                                      ** C
 1140 BNEW = 1.0
      IF (DREF .GT. RHOP(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
 1141 B1 = BP+DEL(MP,N)*ALOG(EPS(MP,N)-BNEW)
      BNEW = AMIN1(BNEW+(B1-BNEW)/(1.+DEL(MP,N)/(EPS(MP,N)-BNEW)),0.9999
     19999)
      NW = NW+1
      AW = NW
      IF (ABS(BNEW-B1) .GT. 1.E-6 .AND. AW .LT. 10.) GO TO 1141
 1143 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)
 1155 PST = PS*BNEW*F
      GO TO 1108
 1160 CONTINUE
C                                                                     C
C **      HERRMANN P-ALPHA.                                        ** C
      PST = 0.
      DC = RHOS*(PORC(MP,1,N)*F/EQSTCM+1.)/TF
      DC = RHOS*(1.+TSQE(0,PORC(MP,1,N)*F,EQSTGM*DC*E,EQSTCM,
     1  EQSTDM,EQSTSM,EQSTGM,EQSTHM,EQSTEM,RHOS,EQSTNM,E,EQSTVM,EQSTAM,
     2  NCYC))
      IF (DC .LT. D) GO TO 1109
      DY = RHOP(MP,1,N)/TF*(1.+PORA(MP,1,N)/AK(MP))
      ALFY  = 1./(DY*TF/RHOS-PORA(MP,1,N)*F/EQSTCM)
      DD = AMAX1(D,DY)
      DYD = DY*ALFY/DD
      DCD = DC/DD
      B1 = (DCD-DYD)**2/(ALFY - 1.)
      B2 = DCD+B1/2.
      ALFS = B2-SQRT(B2*B2-DCD*DCD-B1)
      DS = ALFS*DD
      GO TO (1170,1172,1174) NPRM
 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)
 1178 IF(D .GE. DY) GO TO 1179
      DYD = DY*ALFY/D
      DCD = DC/D
      B1 = (DCD-DYD)**2/(ALFY - 1.)
      B2 = DCD+B1/2.
      ALFS = B2-SQRT(B2*B2-DCD*DCD-B1)
 1179 PST = PS/ALFS
      IF (PEL .LT. PST) GO TO 1300
      PJ = PST
 1180 CONTINUE
 1300 PJ = PEL
      IF (PST .LT. PEL) PJ = PST
C                                                                     C
C  *      COMPUTE RELATIVE VOID VOLUME.(RVV)                        * C
C                                                                     C
      PTH=TSQE(1,PJ*RHOS/D,EQSTGM*RHOS*E,EQSTCM,EQSTDM,EQSTSM,EQSTGM,
```

```
     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 (IH .NE. 5R    S) GO TO 1900                                    PEST     607
      RVV = 0.  $  H = 5R    S  $  GO TO 1900                            PEST     608
C                                                                      C PEST     609
C ***    DYNAMIC PRESSURE.                                         *** C PEST     610
C                                                                      C PEST     611
1310  KCRD=KCD(MP)                                                       PEST     612
      IF (H .EQ. 5R    R .AND. KRD(MP) .NE. 0) KCRD = KRD(MP)            PEST     613
      IF (KCRD .GT.1) GO TO 1320                                         PEST     614
C                                                                      C PEST     615
C **     NO RATE-DEPENDENCE.                                        ** C PEST     616
      IF (IH .EQ. 5R    S) H = 5R    S                                   PEST     617
      GO TO 1900                                                         PEST     618
1320  PELS=TSQE(1,PEL*RHOS/D,EQSTGM*RHOS*E,EQSTCM,EQSTDM,EQSTSM,EQSTGM,  PEST     619
     1  EQSTHM,EQSTEM,RHOS,EQSTNM,E,EQSTVM,EQSTAM,NCYC)                  PEST     620
      IF (PEL .NE. 0.) ALFL=PELS/PEL                                     PEST     621
      IF (PEL .EQ.0.) ALFL=PELS/D                                        PEST     622
      ALFSD = (ALFS-AST1)/DT                                             PEST     623
      ALFLD = (ALFL - ALFD1)/DT                                          PEST     624
      GO TO (1900,1340,1380,1440) KCRD                                   PEST     625
C                                                                      C PEST     626
C **     LINEAR VISCOUS VOID COMPACTION.                            ** C PEST     627
1340  VVE = 1.-1./ALFL                                                   PEST     628
      DV = DVO = 1./D-1./DOLD                                            PEST     629
      NLOOP=MAX1(1.,-DV*EQSTCM*D/AMAX1(PST,P)/ALF+0.8,-4.*TER(MP,1,N)*DT PEST     630
     1  *(P-PST1))                                                       PEST     631
      VOLD = 1./DOLD  $  VSO = (1.-RVV1)/DOLD                            PEST     632
      NTRY = 0                                                           PEST     633
      RVVL = RVV1                                                        PEST     634
      PTHL = PTHO = PST1*AST1                                            PEST     635
      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-PTHO)                           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-PTHO                                                    PEST     648
1341  DELV = DV/NLOOP  $  VH = VOLD      $  DTN = DELV/DVO*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
      PTHH = PTHO+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
      PLO = PTHH  $  PUP = PELH = AMAX1(P,PST1)/(1.-RVV1)+(PELS-AMAX1(P, PEST     660
     1  PST1)/(1.-RVV1))*RATIO                                           PEST     661
      PSA = PELH  $  ZG = RVVL*VH                                        PEST     662
      IF (PTHH .GT. PELH) GO TO 1346                                     PEST     663
      PSJ = (DELV+VSO-VSTH+PTHH*RVP+PSO*DVDP-RVVL*VH*(XG*(1.+A1/2.*      PEST     664
     1  (-PTHH-PSO+PTHL))-1.))/(RVP+DVDP+RVVL*VH*XG*A1/2.)               PEST     665
      NC = 0                                                             PEST     666
1342  NC = NC+1                                                          PEST     667
      DP = (AMAX1(0.,PSJ-PTHH)+AMAX1(0.,PSO-PTHL))/2.                    PEST     668
      ZG = RVVL*VH  $  IF (DP .GE. 0.) ZG = ZG*EXP(A1*DP)                PEST     669
      DELVA = VSTH-VSO+RVP*(PSJ-PTHH)+DVDP*(PSJ-PSO)+ZG-RVVL*VH          PEST     670
      PSA = PSJ                                                          PEST     671
      AC = NC                                                            PEST     672
      IF (ABS(DELVA-DELV) .LT. 1.E-5*VH .OR. (PSJ .LE. PTHH .AND. AC     PEST     673
     1  .GT. 1.)) GO TO 1346                                             PEST     674
      IF (NC .GE. 10) GO TO 1348                                         PEST     675
```

```
      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
      PSJ = PSJ+(DELV-DELVA)/(RVP+DVDP+ZG*A1/2.)
      GO TO 1344
C        INTERPOLATION ESTIMATE OF PRESSURE IN SOLID
1343  PSJ = PSA+(DELV-DELVA)/(DELVB-DELVA)*(PSB-PSA)
1344  CONTINUE
      IF (PSJ .GT. PUP) PSJ = PUP-1.E7
      IF (PSJ .LT. PLO) PSJ = PLO+1.E7
      IF (NC .EQ. 1) GO TO 1345
      IF (ABS(DELVA-DELV) .GT. ABS(DELVB-DELV)) GO TO 1342
1345  PSB = PSA  $  DELVB = DELVA  $  GO TO 1342
C        CONCLUSION OF LOOP
1346  RVVL = ZG/VH  $   PTHL = PTHH   $   PSA = PSO = AMAX1(PTHH,AMIN1
     1 (PELH,PSA))
      VSO = VH-ZG  $   ENT = ENT*VOLD/VH
1347  CONTINUE
      PJ = (1.-RVVL)*PSA  $  RVV = RVVL  $  GO TO 1900
C        PROVISION FOR ABORT FOR ITERATION FAILURE
1348  NTRY = NTRY+1  $  IF (NTRY .GE. 5) GO TO 1349
      VOLD = VH-DELV  $  DV = 1./D-VOLD
      NLOOP = MAX1(3.,-2.**NTRY*DV*EQSTCM*D/AMAX1(PST,P)/ALF+0.8)
      GO TO 1341
1349  WRITE(6,2349)M,P,DV,DELVA,DELVB
      GO TO 1346
C
C **     PORHOLT MODEL - DYNAMIC.                                    **
1380  ALFD = TPH(MP,N)*ALFLD +AST1 +ALFSD*(DT-TPH(MP,N))+(ALFD1-TPH(
     1 MP,N)*ALFLD-AST1+TPH(MP,N)*ALFSD)*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)
1400  PJ=AMIN1(PEL,AMAX1(PST,PS/ALFD))
      PS1=TSQE(1,PJ*RHOS/D,EQSTGM*RHOS*E,EQSTCM,EQSTDM,EQSTSM,EQSTGM,
     1 EQSTHM,EQSTEM,RHOS,EQSTNM,E,EQSTVM,EQSTAM,NCYC)
      IF (PJ .NE. 0.) RVV=AMAX1(0.,1.-PJ/PS1)
      IF (PJ .EQ. 0.) RVV=AMAX1(0.,1.-D/PS1)
      GO TO 1900
C **     BUTCHER P-ALPHA-TAU                                         **
1440  CONTINUE
      BT=TPH(MP,N)*(ALFL-ALFS)/DADP(MP,N)/(PEL-PST)
      ALFD=((ALFL-ALFD1)*BT/DT-ALFS+ALFD1)*EXP(DT/BT)+ALFS-(ALFL-ALFD1)*
     1 BT/DT
      IF (ALFD .LT. ALFS) ALFD = ALFS
      IF (ALFD .GT. ALFL) ALFD = ALFL
      GO TO 1382
C
C ***    TENSILE PATH.                                              ***
C **     STATIC FRACTURE THRESHOLD CURVE.                            **
C
1500  KTSS = KTS(MP)
      IF(KTSS .EQ. 0) KTSS = KCS(MP)
      N = 2
      GO TO (1520,1540,1560) KTSS
C
C **     VARIABLE STRENGTH.
1520  PTH = TER(MP,5,N)*F
      PST = D*PTH*(1./RHOS+EQSTGM*E/EQSTCM)/(1.+PTH/EQSTCM)
      GO TO 1600
C
C **     FRACTURE MECHANICS.                                         **
1540  GO TO 1520
C
C **     CARROLL-HOLT THRESHOLD STRESS.                              **
1560  PST = PEL
      IF (DREF .GT. RHOP(MP,5,N)) GO TO 1600
      BNEW = BP = DREF/RHOS
      NW = 0
1565  B1 = BP+DEL(MP,N)*ALOG(EPS(MP,N)-BNEW)
```

SUBROUTINE PEST (Continued)

```
      BNEW = AMIN1(BNEW+(B1-BNEW)/(1.+DEL(MP,N)/(EPS(MP,N)-BNEW)),0.9999
     19999)
      NW = NW+1
      AW = NW
      IF (ABS(BNEW-B1) .GT. 1.E-6 .AND. AW .LT. 10.) GO TO 1565
      DS = DREF/BNEW
      GO TO (1570,1572,1574) NPRM
 1570 CALL EQST(0.,DS,PS,M,CJ,DPDDJ,DPDEJ)
      GO TO 1580
 1572 CALL ESA(1,5,M,CJ,DS,0.,PS,DPDDJ,DPDEJ)
      GO TO 1580
 1574 CALL EQSTPF(1,5,M,CJ,DS,0.,PS)
 1580 PST = PS*BNEW*F
      IF (PST .GT. PS) GO TO 1600
      PST = PS
      IH = 5R    S
      IF (PEL .GT. PS) GO TO 1600
      PJ = PS
      H = 5R    S  $  RVV = 0.
      GO TO 1900
 1600 PJ = PEL
      IF (H .NE. 5R    S) H = 5R    T
      IF (PEL .LT. PST) H = 5R    T
      IF (PEL .LT. PST) PJ = PST
C                                                                      C
C **     COMPUTE RELATIVE VOID VOLUME.(RVV)                         ** C
C                                                                      C
      PTH=TSQE(1,PJ*RHOS/D,EQSTGM*RHOS*E,EQSTCM,EQSTDM,EQSTSM,EQSTGM,
     1  EQSTHM,EQSTEM,RHOS,EQSTNM,E,EQSTVM,EQSTAM,NCYC)
      IF (PJ .NE. 0.) RVV = AMAX1(0.,1.-PJ/PTH)
      IF (PJ .EQ. 0.) RVV=AMAX1(0.,1.-D/PTH)
      ALFS = 1./(1.-RVV)
      IF (RVV .GT. TER(MP,7,N)) GO TO 2000
      IF (PEL .GE. PST) GO TO 1900
C                                                                      C
C **     DYNAMIC TENSILE PRESSURE.                                  ** C
C                                                                      C
      KTDD = KTD(MP)
      IF (KTDD .EQ. 0) KTDD = KCD(MP)
      IF (KTDD .EQ. 0 .AND. KCDM .EQ. 0) KTDD = 1
      GO TO (1615,1620,1660) KTDD
C                                                                      C
C **     NO RATE DEPENDENCE.                                        ** C
 1615 PJ = PST
      GO TO 1635
C                                                                      C
C **     N. A. G. DUCTILE FRACTURE MODEL.                           ** C
 1620 DV = DVO = 1./D-1./DOLD
      VVE =1.-PEL/TSQE(1,PEL*RHOS/D,EQSTGM*RHOS*E,EQSTCM,EQSTDM,EQSTSM,
     1  EQSTGM,EQSTHM,EQSTEM,RHOS,EQSTNM,E,EQSTVM,EQSTAM,NCYC)
      IF (AST1 .EQ. 0.) AST1 = ALFS
      PELS = PEL/(1.-VVE)
      NLOOP=MAX1(1.,-DV*EQSTCM*D/AMIN1(PST,P)/ALF+0.8,4.*TER(MP,1,N)*DT
     1  *(P-PST1))
      VOLD = 1./DOLD  $  VSO = (1.-RVV1)/DOLD
      NTRY = 0
      RVVL = RVV1
      PTHL = PTHO = PST1*AST1
      PSO = AMIN1(P,PST1)/(1.-RVV1)
      IF(PST1 .GT. 0.) PSO=PTHL=PTHO=0.
      IF (1.- RVV1 - 1./AST1 .GT. 0..AND. PSO .LT. PTHO) GO TO 16201
      DRVP = 0.
      RVPO = -1./(DOLD*EQSTCM)
      GO TO 16203
16201 RVPO = (1.-RVV1-1./AST1)/DOLD/(PSO-PTHO)
      DRVP = (RVV-VVE)/D/(PELS-PTH)-RVPO
16203 VSTHO = 1./(DOLD*AST1)
      DVSTH = (1.-RVV)/D-VSTHO
      DVDP = (VVE/D-RVV1/DOLD)/(PELS-PSO)
      IF (PST1 .EQ. 0. .OR. PST1 .LT. P) PTHL = PTHO = PTH
      DPTH = PTH-PTHO
 1621 DELV = DV/NLOOP  $  VH = VOLD  $  DTN = DELV/DVO*DT
      A1 = TER(MP,1,N)*DTN
C        BEGIN DO LOOP FOR SUBCYCLING
      DO 1632 NL = 1,NLOOP
```

```
      VH = VH+DELV  $  RATIO = (VH-1./DOLD)/DVO
      RVP = RVPO+DRVP*RATIO
      VSTH = VSTHO+DVSTH*RATIO
      PTHH = PTHO+DPTH*RATIO
C        FIRST ESTIMATE OF PRESSURE IN SOLID
      DP = AMIN1(0.,PSO-PTHL)
      XG = 1.  $  XN = 0.
      IF (DP .GE. 0.) GO TO 1622
      XG = EXP(A1*DP)
      XN = EXP(DP/TER(MP,6,N))
 1622 PLO = PTHH  $  PUP = PELH = AMIN1(P,PST1)/(1.-RVV1)+(PELS-AMIN1(P,
     1 PST1)/(1.-RVV1))*RATIO
      ZG = RVVL*VH  $  ZN = 0.  $  PSA = PELH
      IF (PTHH .LT. PELH) GO TO 1630
      PSJ = (DELV+VSO-VSTH+PTHH*RVP+PSO*DVDP-RVVL*VH*(XG*(1.+A1/2.*
     1 (-PTHH-PSO+PTHL))-1.)-TER(MP,8,N)*VH*DTN*XN*(1.-(PTHH+PSO-PTHL)/
     2 2./TER(MP,6,N)))/(RVP+DVDP+RVVL*VH*XG*A1/2.+TER(MP,8,N)*VH*DTN*
     3 XN/2./TER(MP,6,N))
      NC = 0
 1623 NC = NC+1
      DP = (AMIN1(0.,PSJ-PTHH)+AMIN1(0.,PSO-PTHL))/2.
      ZG = RVVL*VH  $  ZN = 0.
      IF (DP .GE. 0.) GO TO 1624
      ZG = ZG*EXP(A1*DP)
      ZN = TER(MP,8,N)*VH*DTN*EXP(DP/2./TER(MP,6,N))
 1624 DELVA = VSTH-VSO+RVP*(PSJ-PTHH)+DVDP*(PSJ-PSO)+ZG-RVVL*VH+ZN
      PSA = PSJ
      AC = NC
      IF (ABS(DELVA-DELV) .LT. 1.E-5*VH .OR. (PSJ .GE. PTHH .AND. AC
     1 .GT. 1.)) GO TO 1630
      IF (NC .GE. 10) GO TO 1640
      IF (DELVA .LT. DELV) PLO = AMIN1(PLO,PSA)
      IF (DELVA .GT. DELV) PUP = AMAX1(PSA,PUP)
C        MAKE 2ND ESTIMATE OF PRESSURE IN THE SOLID
      IF (MOD(NC,2) .EQ. 0) GO TO 1625
      PSJ = PSJ+(DELV-DELVA)/(RVP+DVDP+ZG*A1/2.+ZN/2./TER(MP,6,N))
      GO TO 1626
C        INTERPOLATION ESTIMATE OF PRESSURE IN SOLID
 1625 PSJ = PSA+(DELV-DELVA)/(DELVB-DELVA)*(PSB-PSA)
 1626 IF (PSJ .LT. PUP) PSJ = PUP+1.E7
      IF (PSJ .GT. PLO) PSJ = PLO-1.E7
      IF (NC .EQ. 1) GO TO 1627
      IF (ABS(DELVA-DELV) .GT. ABS(DELVB-DELV)) GO TO 1623
 1627 PSB = PSA  $  DELVB = DELVA
      GO TO 1623
C        CONCLUSION OF LOOP
 1630 RVVL = (ZG+ZN)/VH  $   PTHL = PTHH   $   PSA=PSO=AMIN1(PTHH,AMAX1
     1 (PELH,PSA))
      VSO = VH-ZG-ZN
      ENT = ENT*VOLD/VH+TER(MP,4,N)*EXP(DP/2./TER(MP,6,N))*DTN
 1632 CONTINUE
      PJ = (1.-RVVL)*PSA
      RVV = RVVL
 1635 IF (RVV .GT. TER(MP,7,N)) GO TO 2000
      GO TO 1900
C        PROVISION FOR ABORT FOR ITERATION FAILURE
 1640 NTRY = NTRY+1
      IF (NTRY .GE. 5) GO TO 1643
      VOLD = VH-DELV  $  DV = 1./D-VOLD
      NLOOP = MAX1(3.,-2.**NTRY*DV*EQSTCM*D/AMIN1(PST,P)/ALF+0.8)
      GO TO 1621
 1643 WRITE(6,2349)M,P,DV,DELVA,DELVB
      GO TO 1630
C        BRITTLE FRACTURE AND FRAGMENTATION.
 1660 GO TO 1900
C                                                                      C
C **     SOLID AND POROUS MELT AND SOLID BEHAVIOR                   ** C
C                                                                      C
 1800 GO TO (1805,1810,1815) NPRM
 1805 CALL EQST(E,D,PS,M,C,DPDDJ,DPDEJ)
      GO TO 1840
 1810 CALL ESA(1,5,M,C,D,E,PS,DPDDJ,DPDEJ)
      GO TO 1840
 1815 CALL EQSTPF(1,5,M,C,D,E,PS)
 1840 IF (H .NE. 5R    S) GO TO 1850
```

SUBROUTINE PEST (Concluded)

```
      IF (F .EQ. 0.) GO TO 1850                                         PEST     901
      PJ=PST=PEL=PS                                                     PEST     902
      GO TO 1860                                                        PEST     903
1850  PJ = PST = PEL = AMAX1(0.,PS)                                     PEST     904
      IF (PJ .GT. 0.) GO TO 1855                                        PEST     905
      PTH = TSQE(1, PJ*RHOS/D, EQSTGM*RHOS*E, EQSTCM, EQSTDM, EQSTSM,   PEST     906
     1  EQSTGM,EQSTHM,EQSTEM,RHOS,EQSTNM,E,EQSTVM,EQSTAM,NCYC)          PEST     907
      RVV= AMAX1(0., 1. - D/PTH)                                        PEST     908
      H = 5R    M                                                       PEST     909
      GO TO 1860                                                        PEST     910
1855  H=5R     S  $  RVV=0.                                             PEST     911
1860  IF (PEL .LT. 0.) GO TO 1500                                       PEST     912
C                                                                     C PEST     913
C **     ENDING ROUTINE.                                           ** C PEST     914
C                                                                     C PEST     915
      IF (H .EQ. 5R    M .OR. H .EQ. 5R    S) GO TO 1905                PEST     916
 1900 DPDDJ=DPDEJ=0.                                                    PEST     917
 1905 P=PJ                                                              PEST     918
      PST1=PST  $  AST1=ALFS                                            PEST     919
      RETURN                                                            PEST     920
C        FRAGMENTATION.                                                 PEST     921
 2000 P=PST1=TJ=0.                                                      PEST     922
      RVV = -ABS(RVV)                                                   PEST     923
      AST1 = 1./(1.+RVV)                                                PEST     924
      H = 5R    Z                                                       PEST     925
      RETURN                                                            PEST     926
2300  FORMAT(* D,BULK,MUM,C,F,ELK,ELG,RHOP1,E=*1P9E10.3)                9/12/79   16
 2349 FORMAT(*    ITERATION FAILURE,M=*I2,* P=*1PE10.3,* DV=*1PE10.3,   PEST     928
     1  * DELVA=*1PE10.3,* DELVB=*1PE10.3)                              PEST     929
      END                                                               PEST     930
```

SUBROUTINE POREQST

```
      SUBROUTINE POREQST(NCALL,IN,M,C,D,DOLD,E,EOLD,F,P,CZJ,CWJ,H,DPDE,
     1  EQSTCM,EQSTDM,EQSTGM,EQSTSM,MUM,RHOSM,YADDM,NDSM,NPRM,J)
C
C     ROUTINE READS INPUT DATA FOR POROUS MATERIAL AND COMPUTES PRESSURE
C
C     READ INPUT (NCALL=0). CALL IS FROM GENRAT
C       INPUT  - NCALL, IN, M, MATERIAL PROPERTY CARDS
C       OUTPUT - ORGANIZES DATA AND FILLS AK, MUP, PORA, RHOP, YADDP
C       PREPARE- D = RHOP(M,1),   CZJ = CZQ(M)  = COSQ(M,6)
C                YADDM = Y0(M),   CWJ = CWQ(M)  = C1(M,6)
C                C = EXMAT(M,3) = SOUND SPEED
C     COMPUTE PRESSURE (NCALL=1)
C       INPUT  - NCALL,M,C,D,DOLD,E,EOLD,F,P=POLD,H,EQSTCM,EQSTGM,RHOSM,
C                NDSM,NPRM
C       OUTPUT - C,P,H,CZJ,CWJ,DPDE,MUM,YADDM
C       NOTE CHANGE IN INPUT SO THAT FIRST VALUE OF P2 IS YIELD AND
C       PERTAINS TO D .LE. RHOP(M,2)
C
      REAL MUM,MUP
      INTEGER H
      COMMON /POR/ AK(6),MUP(6),NREG(6),PORA(6,5),PORB(6,5),PORC(6,5),
     1  RHOP(6,6),YADDP(6,5)
      DIMENSION COSQ(6,6),C1(6,6),TEMP(8)
C
      DATA  NAT,NBT,NCT,NDT,NET,NFT/10H -POREQST-,10H, ,G/CM3  ,
     1  10H,G/CM3     ,10H,DYN/CM2,=,10H,DYN-CM/G ,10H,=        /
C
      IF (NCALL .EQ. 1) GO TO 200
C
C        ****  READ INPUT DATA FOR POROUS MATERIAL  ****
C
      READ (IN,1192) A1,AK(M),A2,MUP(M),A3,YZERO
      PRINT 1130,A1,AK(M),A2,MUP(M),A3,YZERO,IN,NAT,NDT,NFT
      READ (IN,1100) A1,NREG(M)
      CJ=1.
      WRITE (6,1100) A1,NREG(M)
      WRITE(6,1110) IN,NAT,NBT
      READ (IN,1120) A1,(RHOP(M,I),I=1,6)
      PRINT 1131,A1,(RHOP(M,I),I=1,6),IN,NAT,NCT
      DO 50 I = 1,6
      COSQ(M,I) = 4.0
   50 C1(M,I) = 0.15
55    READ (IN,1005)(TEMP(I),I=1,8)
      DECODE (3,1125,TEMP) A1,A2
      IF (A1 .EQ. 1HC .AND. (A2 .EQ. 1H0 .OR. A2 .EQ. 1HO)) GO TO 60
      IF (A1 .EQ. 1HC .AND. A2 .EQ. 1H1) GO TO 62
      GO TO 65
60    DECODE (80,1120,TEMP) A1,(COSQ(M,I),I=1,6)
      PRINT 1131,A1,(COSQ(M,I),I=1,6),IN,NAT
      GO TO 55
62    DECODE (80,1120,TEMP) A1,(C1(M,I),I=1,6)
      PRINT 1131,A1,(C1(M,I),I=1,6),IN,NAT
      GO TO 55
   65 CZJ = COSQ(M,6)    $   CWJ = C1(M,6)
      NP=NREG(M)    $    P1=0.
      DECODE (80,1192,TEMP) A1,P1
      PRINT 1132,A1,P1,IN,NAT,NDT
      PORA(M,1)=P1    $    PORB(M,1)=PORC(M,1)=0.
      YADDP(M,1)=0.
      DO 110 N = 1,NP
      READ (IN,1192) A1,P2,A2,DELP,A3,YADDP(M,N)
      PRINT 1130,A1,P2,A2,DELP,A3,YADDP(M,N),IN,NAT,NDT,NET
      DRHO=RHOP(M,N+1)-RHOP(M,N)
      PORA(M,N+1)=P1+RHOP(M,N+1)/DRHO*(P2-P1-4.*DELP*RHOP(M,N)/DRHO)
      PORB(M,N+1)=-RHOP(M,N+1)*RHOP(M,N)/DRHO*(P2-P1-4.*DELP*(RHOP(M,N+1
     1   )+RHOP(M,N))/DRHO)
      PORC(M,N+1)=-4.*DELP*(RHOP(M,N+1)*RHOP(M,N)/DRHO)**2
110   P1=P2
```

SUBROUTINE POREQST (Concluded)

```
 170  YADDM = YZERO
      C = SQRT(( AK(M) + 1.333 * MUP(M))/D)
190   RETURN
C
C       **** CALCULATION OF PRESSURE IN A POROUS MATERIAL  ****
C
200   TF=1.+E*EQSTGM*RHOSM/EQSTCM   $   DREF=D*TF   $   NC=5
C        FIND APPROPRIATE DENSITY REGION OF POROUS RELATIONS
      IF (DREF .GT. RHOP(M,6)) GO TO 280
      P2 = 0.
      IF (DREF .GT. RHOP(M,5) .OR. H .EQ. 5R    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,PS,M,CJ,A1,A2)
      IF (H .EQ. 5R    Q) GO TO 225
      IF (PS .LT. P2) GO TO 230
225   P=PS   $   H=5R    Q   $  NC=5   $  RETURN
C        COMPUTE PRESSURE ON INTERMEDIATE SURFACE
230   RHOM=RHOP(M,1)/TF   $   RHOPV=F*(RHOP(M,5)-RHOSM)+RHOSM/TF
      RATIO=AMIN1(1.,(RHOM-D)/(RHOM-RHOPV)*RHOPV/O*(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-OOLD)/(.5*(D+DOLD))+EQSTGM*RHOSM/EQSTCM*(E-EOLD))
C        CHECK WHETHER STATE POINT IS ON INTERMEDIATE OR YIELD SURFACE
      P=P2   $   IF (PBULK .GT. P2) GO TO 250
      P=PBULK   $   IF (DREF .GT. RHOSM) P=AMAX1(PBULK,PS)
C        COMPUTE SOUND SPEED
 250  CSQ =(BULK +1.333 *MUM)/D
      IF (CSQ .LT. 0.) GO TO 270
      CQ=CSQ+C**2   $   C=CSQ*C/CQ + 0.25*CQ/C   $   DPDE=0.
C        COMPUTE ARTIFICIAL VISCOSITY COEFFICIENTS
270   RATIO=0.
      DELR=RHOP(M,NC+1)-RHOP(M,NC)
      IF (DELR .NE. 0.) RATIO=(DREF-RHOP(M,NC))/DELR
      CZJ=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. NREG(M)) YADOM=YADDP(M,NC)
      RETURN
C        COMPUTE PRESSURE IN CONSOLIDATED MATERIAL
280   H=5R    S
      CALL EQST(E,O,P,M,C,OPDO,DPDE)
 1100 FORMAT(A10,I10,A10,1PE10.3)
1005  FORMAT (8A10)
1110  FORMAT(1H+,79X,7H IND=  ,5H, IN=,I2,4A10)
 1120 FORMAT(A10,7E10.3)
 1125 FORMAT (1X,2A1)
 1130 FORMAT(3(A10,1PE10.3),20X,7H IND=  ,5H, IN=,I2,3A10)
 1131 FORMAT(A10,1P6E10.3,10X,7H IND=  ,5H, IN=,I2,3A10)
 1132 FORMAT(A10,1PE10.3,60X,12H IND=  , IN=,I2,3A10)
 1192 FORMAT(4(A10,E10.3))
      RETURN
      END
```

SUBROUTINE PORHOLT

```
      SUBROUTINE PORHOLT(NPART,IN,M,C,DH,DOLD,EH,EOLD,F,P,IH,J,DPDEJ,   PORHOLT2
     1  EQSTCM,MUM,YADDM,RHOS,DT)                                        PORHOLT3
C                                                                        PORHOLT4
C        BASIC EQUATIONS OF THIS MODEL ARE BY AL HOLT OF LLL.            PORHOLT5
C                                                                        PORHOLT6
      REAL MUM,MUP                                                       PORHOLT7
      DIMENSION RHO(6),RHOC(6),ALFO(6),     AK(6),MUP(6),YADDP(6),PY(6),PORHOLT8
     1     Y0(6),ALF(300),A(6),B(6),RHOE(6),RHOES(6),TPH(6)              PORHOLT9
C                                                                        PORHOL10
      IF (NPART .EQ. 1) GO TO 200                                        PORHOL11
C                                                                        PORHOL12
C        *****      READ AND INITIALIZE      *****                       PORHOL13
C                                                                        PORHOL14
C        READ IN SPECIAL PROPERTIES FOR POROUS MATERIAL                  PORHOL15
      WRITE (6,1010)                                                     PORHOL16
      WRITE (6,1011)                                                     PORHOL17
      READ (IN,1001) A1,RHO(M),A2,RHOC(M),A3,TPH(M),A4,PY(M)             PORHOL18
      WRITE (6,1001) A1,RHO(M),A2,RHOC(M),A3,TPH(M),A4,PY(M)             PORHOL19
      READ (IN,1001) A1,AK(M),A2,MUP(M),A3,Y0(M),A4,YADDP(M)             PORHOL20
      WRITE (6,1001) A1,AK(M),A2,MUP(M),A3,Y0(M),A4,YADDP(M)             PORHOL21
      READ (IN, 1001) A1,DPDRHO                                          PORHOL22
      WRITE (6,1001) A1,DPDRHO                                           PORHOL23
C        INITIALIZE YIELD AND DENSITY FOR GENRAT                         PORHOL24
      YADDM = Y0(M)                                                      PORHOL25
      DH=RHO(M)  $  C=SQRT((AK(M)+1.333*MUP(M))/DH)                      PORHOL26
C        INITIALIZE VARIABLES FOR PORHOLT                                PORHOL27
      ALFO(M)=RHOS/RHO(M)                                                PORHOL28
      RHOE(M)=RHO(M)*(PY(M)/AK(M)+1.)                                    PORHOL29
      RHOES(M)=RHOS*(PY(M)/EQSTCM+1.)                                    PORHOL30
      ALFE=RHOES(M)/RHOE(M)                                              PORHOL31
      A(M)=ALFE*(ALFE*RHOE(M)/EQSTCM*DPDRHO-(RHOES(M)-RHOS)/RHOS)        PORHOL32
      B(M)=(RHOC(M)-RHOES(M))/(RHOC(M)-RHOE(M))**2-A(M)/(RHOC(M)-RHOE    PORHOL33
     1     (M))                                                          PORHOL34
      PRINT 1002,A(M),B(M),RHOES(M),RHOE(M)                              PORHOL35
      WRITE (6,1010)                                                     PORHOL36
      CJ=1.                                                              PORHOL37
      DO 150 I=1,300                                                     PORHOL38
150   ALF(I)=0.                                                          PORHOL39
      RETURN                                                             PORHOL40
C                                                                        PORHOL41
C        *****      COMPUTE PRESSURE      *****                          PORHOL42
C                                                                        PORHOL43
C        INITIALIZE ALF                                                  PORHOL44
200   CONTINUE                                                           PORHOL45
      IF (ALF(J) .EQ. 0.) ALF(J)=ALFO(M)                                 PORHOL46
      IF (ALF(J) .LE. 1.) GO TO 300                                      PORHOL47
C        COMPUTE ELASTIC VALUE OF ALF ON UNLOAD OR RELOAD CURVES         PORHOL48
C             AAKC=(CURRENT BULK MODULUS)/(BULK MODULUS OF SOLID)        PORHOL49
      AAKC=AK(M)*(ALFO(M)-1.)/(AK(M)*(ALFO(M)-1.)+(EQSTCM-ALFO(M)*       PORHOL50
     1     AK(M))*(1.-1./ALF(J)))/ALF(J)                                 PORHOL51
      ALFE=ALF(J)*(DOLD/RHOS-P/EQSTCM)/(DH/RHOS-P/EQSTCM-AAKC*(DH-DOLD)/PORHOL52
     1     RHO(M))                                                       PORHOL53
      IF (DH .LT. DOLD) GO TO 250                                        PORHOL54
C        COMPUTE STATIC VALUE OF ALF ON THE FLOW CURVE                   PORHOL55
      ALFS=(RHOES(M)+(A(M)+B(M)*(DH-RHOE(M)))*(DH-RHOE(M)))/DH           PORHOL56
C        CHECK WHETHER FLOW STRESS HAS BEEN REACHED DURING LOADING       PORHOL57
      IF (ALFS .GT. ALFE) GO TO 250                                      PORHOL58
C        COMPUTE DYNAMIC VALUE OF ALF                                    PORHOL59
      ALF(J)=AMAX1(1.,(ALFS+TPH(M)*ALFE/DT)/(1.+TPH(M)/DT))              PORHOL60
      GO TO 255                                                          PORHOL61
250   ALF(J)=ALFE                                                        PORHOL62
C        COMPUTE DENSITY IN THE SOLID                                    PORHOL63
255   DS=ALF(J)*DH                                                       PORHOL64
      POLD=P                                                             PORHOL65
C        COMPUTE PRESSURE IN THE SOLID MATERIAL                          PORHOL66
      CALL EQST(EH,DS,PS,M,CJ,DPDEJ)                                     PORHOL67
C        COMPUTE GROSS PRESSURE                                          PORHOL68
      P=PS/ALF(J)                                                        PORHOL69
```

SUBROUTINE PORHOLT (Concluded)

```
      MUM=((ALFO(M)-ALF(M))*MUM+(ALF(J)-1.)*MUP(M))/(ALFO(M)-1.)        PORHOL70
      YADDM=YADDP(M)                                                    PORHOL71
       CSQ=(AAKC*EQSTCM+1.333*MUM)/DH                                   PORHOL72
      IF (CSQ .LT. 1.E6) GO TO 270                                      PORHOL73
      C=SQRT(CSQ)                                                       PORHOL74
270   RETURN                                                            PORHOL75
C                                                                       PORHOL76
C        *****     PRESSURE IN CONSOLIDATED MATERIAL   *****            PORHOL77
C                                                                       PORHOL78
300   IH=5R    S                                                        PORHOL79
      CALL EQST(EH,DH,P,M,C,DPDEJ)                                      PORHOL80
      RETURN                                                            PORHOL81
 1001 FORMAT(4(A10,1PE10.3))                                            PORHOL82
 1002 FORMAT(*   A,B,RHOES,RHOE =*1P4E13.4)                             PORHOL83
1010  FORMAT(/)                                                         PORHOL84
1011  FORMAT(*    READ IN PORHOLT*)                                     PORHOL85
      END                                                               PORHOL86
```

SUBROUTINE PRESCR

```
      SUBROUTINE PRESCR                                                 PRESCR 2
      INTEGER H,POROUS,PRESS,RINTER,SOLID,SPALL                         PUFCOM 2
      REAL MATL,NEM,NET,NEMH,NETH                                       PUFCOM 3
C              MISCELLANEOUS                                            PUFCOM 4
      COMMON AZERO(1),CEF,CKS,DAVG,DELTIM,DISCPT(10),DOLD,DRHO,DTMAX,   PUFCOM 5
     1  DTMIN,DTN,DTNH,DU,DX,EOLD,F,FAC,FIRST,J,JCYCS,JINIT,            PUFCOM 6
     2  JFIN,JREZON(15),JSMAX,JSTAR,JTS,LSUB(30),M,MAXPR(30),N,NCYCS,   PUFCOM 7
     3  NEDIT,NPERN,NR,NREZON,NSCRB(6),NSEPRAT,NSPALL,NTEDT,            PUFCOM 8
     4  NTEX,NTR(15),POLD,P6(20),R(30),RLAST,SLAST,SMAX,TEDIT(50),      PUFCOM 9
     5  TF,TIME,TJ,TREZON,TS,T6(20),ULAST,UOLD,UZERO,XLAST,XNOW,XOLD    PUFCOM10
     1  ,XJDIT(20)                                                      PUFCOM11
C              HALFSTEP VALUES                                          PUFCOM12
      COMMON DH,DHLAST,DUH,EH,PH,RH,RHLAST,SH,SHLAST,UH,UHLAST,XH,XHLASTPUFCOM13
     1  ,NEMH,NETH                                                      PUFCOM14
C              CONDITION INDICATORS                                     PUFCOM15
      COMMON INF,LINTER,MIRROR,NORMAL,POROUS,PRESS,RINTER,SOLID,SPALL   PUFCOM16
C              CELL LAYOUT                                              PUFCOM17
      COMMON DXX(30),JBND(30),JMAT(30),NAUTO,MATL(6,2),NLAYER,NMTRLS,   PUFCOM18
     1  THK(30)                                                         PUFCOM19
C                                                                       PUFCOM20
C              COORDINATE ARRAYS                                        COORDC02
      COMMON/COORD/X(200),X0(200),CHL(200),DHL(200),DPDD(200),DPDE(200),COORDC03
     1  EHL(200),H(200,3),NEM(200),NET(200),PHL(200),RHL(200),SDT(200), COORDC04
     2  SHL(200),T(200),U(200),YHL(200),ZHL(200)                        COORDC05
      COMMON /JED/JEDIT(100),JNUM(100),JTYP(100),NAME2(40),JEDSIZ,      JEDCOM 2
     1  MODLUS,NERR,NJEDIT,NTAPE                                        JEOCOM 3
      COMMON/NSC/A(5000)                                                NSCCOM 2
      DIMENSION NN(20),NAME(40),LA(1)                                   PRESCR 7
      EQUIVALENCE (LA,A)                                                PRESCR 8
      DATA (NAME(I),I=1,33)/3HX  ,3HX0 ,3HC  ,3HD  ,4HDPDD,4HDPDE,3HE  ,PRESCR 9
     1  3HH1 ,3HH2 ,3HH3 ,3HNEM,3HNET,3HP  ,3HR  ,3HSDT,3HS1 ,3HT  ,3HU PRESCR10
     2 ,3HY  ,3HZ  ,4HCOM1,4HCOM2,4HCOM3,4HCOM4,4HCOM5,5HS-INT,3HS2 ,   PRESCR11
     3  3HS3 ,  3HIMP,3HV  ,3HSD1,3HSD2,3HS03/                          PRESCR12
      DATA (NAME2(1),1=1,33)/1RX,2RX0,1RC,1RD,4RDPDD,4RDPDE,1RE,2RH1,   PRESCR13
     1  2RH2,2RH3,3RNEM,3RNET,1RP,1RR,3RSDT,2RS1,1RT,1RU,               PRESCR14
     2  1RY,1RZ,4RCOM1,4RCOM2,4RCOM3,4RCOM4,4RCOM5,5RS-INT,2RS2,2RS3,   PRESCR15
     3  3RIMP,1RV,3RSD1,3RSD2,3RSD3/                                    PRESCR16
C       JK FIRST CARD OF A(100+ ) TO BE READ FROM                       PRESCR17
C       JF FIRST WORD OF A( ) TO BE READ INTO                           PRESCR18
C       KB,KE BEGINNING AND ENDING VALUE J FOR GROUP                    PRESCR19
C       JE COUNTER FOR JEDITS                                           PRESCR20
C       JFIRST FIRST JEDIT OF A GROUP                                   PRESCR21
C       JTLAST INDICATOR THAT THE PREVIOUS GROUP WAS ALPHA (=0) OR      PRESCR22
C       INTEGER (=1)                                                    PRESCR23
C       K CHARACTER COUNTER                                             PRESCR24
C       JKMAX NUMBER OF CHARACTERS USED AT A TIME                       PRESCR25
C       NCARD NUMBER OF RECORDS DECODED                                 PRESCR26
C       IW COUNTER FOR ALPHA GROUPS                                     PRESCR27
C       K  COUNTER FOR CHARACTERS ON A CARD                             PRESCR28
C       KP  PERIOD INDICATOR                                            PRESCR29
C       JTYP( )  TITLE ARRAY FOR HEADINGS IN SCRIBE                     PRESCR30
C       JNUMC( )  LOCATION IN ARRAY COMMON                              PRESCR31
C       JEDIT( )  J(CELL OR COORDINATE) NUMBER                          PRESCR32
      NJD=IABS(NJEDIT)                                                  PRESCR33
      IF (NJEDIT .LT. 0) NJD=(-NJEDIT-1)/14+1                           PRESCR34
      JK=1                                                              PRESCR35
      NLAY1=NLAYER+1                                                    PRESCR36
      DO 50 I=1,NLAY1                                                   PRESCR37
      JEDIT(1)=I                                                        PRESCR38
      JTYP(1)=5HS-INT                                                   PRESCR39
   50 JNUM(1)=5000                                                      PRESCR40
   60 NLL=0                                                             PRESCR41
      IF (H(1,2) .EQ. SPALL) GO TO 70                                   PRESCR42
      NLL=NLL+1                                                         PRESCR43
      JEDIT(NLAYER)=0                                                   PRESCR44
   70 IF (H(JFIN,2) .EQ. SPALL) GO TO 75                                PRESCR45
      JEDIT(NLAYER+NLL)=NLAYER                                          PRESCR46
      NLL=NLL+1                                                         PRESCR47
```

```
   75 JE=NLAYER+NLL                                                     PRESCR48
      JFIRST=NLAYER+NLL                                                 PRESCR49
      JTLAST=0                                                          PRESCR50
      AB=1H                                                             PRESCR51
      NCARD=0                                                           PRESCR52
      K=1                                                               PRESCR53
      KB=1                                                              PRESCR54
      JKMAX=70                                                          PRESCR55
      JF=1                                                              PRESCR56
      IW=1                                                              PRESCR57
C                                                                       PRESCR58
C        SELECT A GROUP OF CHARACTERS                                   PRESCR59
80    DECODE (80,1024,A(4000+JK))(A(L),L=JF,JKMAX)                      PRESCR60
      JF1=JF-1                                                          PRESCR61
      JK1=4000+JK-1                                                     PRESCR62
      KP=0                                                              PRESCR63
      NCARD=NCARD+1                                                     PRESCR64
100   IF (K .GT. JKMAX) GO TO 300                                       PRESCR65
      IF (LA(K) .EQ. 1H ) GO TO 150                                     PRESCR66
      IF ((LA(K) .GE. 1HA .AND. LA(K) .LE. 1HZ) .OR.                    PRESCR67
     1  (LA(K) .GE. 1H0 .AND. LA(K) .LE. 1H9)) GO TO 140                PRESCR68
      IF (LA(K) .NE. 1H.) GO TO 150                                     PRESCR69
      KP=K                                                              PRESCR70
  140 K=K+1                                                             PRESCR71
      IF (K .GT. JKMAX) GO TO 160                                       PRESCR72
      GO TO 100                                                         PRESCR73
  150 IF (KB .LT. K) GO TO 160                                          PRESCR74
      KB=KB+1                                                           PRESCR75
      K=K+1                                                             PRESCR76
      GO TO 100                                                         PRESCR77
160   KE=K-1                                                            PRESCR78
C                                                                       PRESCR79
C        EXAMINE A GROUP OF CHARACTERS FOR TYPE                         PRESCR80
      NK=KE-KB+1                                                        PRESCR81
      NNL=KP-KB                                                         PRESCR82
      JFR=KE                                                            PRESCR83
      IF (KP .NE. 0) GO TO 220                                          PRESCR84
      IF (A(KB) .GE. 1HA .AND. A(KB) .LE. 1HZ) GO TO 180                PRESCR85
C        INTEGER DATA                                                   PRESCR86
      KN =10-NK                                                         PRESCR87
      ENCODE(10,1021,A1)(AB,L=1,KN),(A(L),L=KB,KE)                      PRESCR88
      DECODE (10,1020,A1) JEDIT(JE)                                     PRESCR89
      IF (JE .NE. NLAYER .OR. IW .NE. 1) GO TO 175                      PRESCR90
      NN(IW)=2HS1                                                       PRESCR91
      IW=IW+1                                                           PRESCR92
175   JTLAST=1                                                          PRESCR93
      JE=JE+1                                                           PRESCR94
      GO TO 260                                                         PRESCR95
C        ALPHABETIC DATA                                                PRESCR96
180   IF (JTLAST .EQ. 0) GO TO 210                                      PRESCR97
C        SET TYPE INDICATORS FOR ALL JEDITS OF A SET AFTER THE NEXT     PRESCR98
C        ALPHA GROUP HAS OCCURRED                                       PRESCR99
      JE=JE-1                                                           PRESC100
      NDJ=0                                                             PRESC101
      IW1=IW-1                                                          PRESC102
      NDJ=-(JE-JFIRST+1)                                                PRESC103
      DO 205 I=1,IW1                                                    PRESC104
C                                                                       PRESC105
C        CHECK LEGITIMACY OF ALPHA DATA                                 PRESC106
      IF ((NN(I) .AND. 7777770000000000000B) .EQ. 3LCOM) GO TO 190      PRESC107
      DO 185 IK=1,33                                                    PRESC108
      IF (NN(I) .EQ. NAME(IK)) GO TO 190                                PRESC109
  185 CONTINUE                                                          PRESC110
      GO TO 205                                                         PRESC111
  190 NDJ=JE-JFIRST+1+NDJ                                               PRESC112
      DO 200 J=JFIRST,JE                                                PRESC113
      JTYP(J+NDJ)=NN(I)                                                 PRESC114
      IF (I .EQ. 1) GO TO 200                                           PRESC115
      JEDIT(J+NDJ)=JEDIT(J)                                             PRESC116
```

```
200   CONTINUE                                                          PRESC117
 205  CONTINUE                                                          PRESC118
      JE=JFIRST=JE+NDJ+1                                                PRESC119
      JTLAST=0                                                          PRESC120
      IW=1                                                              PRESC121
C        DECODE THE ALPHA GROUP OF A SET                                PRESC122
210   ENCODE (NK,1021,NN(IW))(A(L),L=KB,KE)                             PRESC123
      IW=IW+1                                                           PRESC124
      GO TO 260                                                         PRESC125
C        JEDIT LISTED AS LAYER AND FRACTION                             PRESC126
220   NNL=KP-KB                                                         PRESC127
      JFR=KE-KP+1                                                       PRESC128
      KP1=KP-1                                                          PRESC129
      KN=10-NNL                                                         PRESC130
      ENCODE (10,1021,A1)(AB,L=1,KN),(A(L),L=KB,KP1)                    PRESC131
      DECODE(10,1020,A1)NL                                              PRESC132
      ENCODE (JFR,1021,A1)(A(L),L=KP,KE)                                PRESC133
      DECODE(10,1025,A1) FR                                             PRESC134
      JEND=JBND(NL)                                                     PRESC135
      JBEG=1                                                            PRESC136
      IF (NL .GT. 1) JBEG=JBND(NL-1)+1                                  PRESC137
      DIST=X(JBEG)+FR*(X(JEND)-X(JBEG))                                 PRESC138
      J=JBEG                                                            PRESC139
240   J=J+1                                                             PRESC140
      IF (X(J) .LT. DIST) GO TO 240                                     PRESC141
      JEDIT(JE)=J-1                                                     PRESC142
      JE=JE+1                                                           PRESC143
      JTLAST=1                                                          PRESC144
260   CONTINUE                                                          PRESC145
      K=KB=K+1                                                          PRESC146
      KP=0                                                              PRESC147
      GO TO 100                                                         PRESC148
C                                                                       PRESC149
C        PREPARE FOR NEXT CARD OF DATA                                  PRESC150
300   IF (NCARD .GE. NJD) GO TO 400                                     PRESC151
      JF=1                                                              PRESC152
      JK=1+8*NCARD                                                      PRESC153
      JKMAX=70                                                          PRESC154
      KDIF=K-KB                                                         PRESC155
      IF (KB .EQ. K) GO TO 330                                          PRESC156
      KB1=KB-1                                                          PRESC157
      KDIF1=KDIF+1                                                      PRESC158
      DO 320   KK=1,KDIF1                                               PRESC159
320   A(KK)=A(KB1+KK)                                                   PRESC160
      JF=1+KDIF1                                                        PRESC161
      JKMAX=70+KDIF+1                                                   PRESC162
      KB=1                                                              PRESC163
      K=KB+KDIF+1                                                       PRESC164
      GO TO 80                                                          PRESC165
330   KB=K=1                                                            PRESC166
      GO TO 80                                                          PRESC167
400   CONTINUE                                                          PRESC168
      JE=JE-1                                                           PRESC169
      NDJ=0                                                             PRESC170
      IW1=IW-1                                                          PRESC171
      DO 420 I=1,IW1                                                    PRESC172
      NDJ=(JE-JFIRST+1)*(I-1)                                           PRESC173
      DO 420 J=JFIRST,JE                                                PRESC174
      JTYP(J+NDJ)=NN(I)                                                 PRESC175
      IF (I .EQ. 1) GO TO 420                                           PRESC176
      JEDIT(J+NDJ)=JEDIT(J)                                             PRESC177
 420  CONTINUE                                                          PRESC178
      NJEDIT=JE+NDJ                                                     PRESC179
C        PRINT AND PUNCH JEDIT VALUES                                   PRESC180
      JFIRST=1                                                          PRESC181
      IEND=0                                                            PRESC182
      JTYPE=JTYP(1)                                                     PRESC183
      DO 450 J=2,NJEDIT                                                 PRESC184
```

SUBROUTINE PRESCR (Concluded)

```
      IF (JTYP(J) .EQ. JTYPE) GO TO 445
      JEND=J-1
425   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. 550000000000B) GO TO
     1  427
      DECODE(10,1052,JTYPE)JA
      GO TO 428
 427  DECODE(10,1051,JTYPE)JA
 428  JNUMB=3999+JA
      I1=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 (11 .EQ. 20) GO TO 439
      IF(II .EQ. 25) GO TO 437
      ENCODE(10,1430,JTYP(I))NAME2(II),JEDIT(1)
      GO TO 440
437   I1=1+1
      ENCODE(10,1431,JTYP(I))NAME2(25),I,I1
      GO TO 440
 439  ENCODE(10,1432,JTYP(1))JA,JEDIT(I)
440   JNUM(I)=JNUMB
      JFIRST=J
      JTYPE=JTYP(J)
445   CONTINUE
      IF (J .LT. NJEDIT .OR. IEND .EQ. 1) GO TO 450
      JEND=NJEDIT
      IEND=1
      JTYPE=JTYP(J)
      GO TO 425
450   CONTINUE
      PRINT 1450,(JTYP(1),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(,I3,1H))
1431  FORMAT(R5,1H(,I1,1H,I1,1H))
 1432 FORMAT(3HCOM,I2,1H(,I3,1H))
1450  FORMAT (/*    OUTPUT FROM PRESCR*/(4X,12(A10)))
      END
```

SUBROUTINE REBAR

```
      SUBROUTINE REBAR(LL,IN,JC,IC,M,N,IH,DH,DOLD,SX,SY,SZ,TXY,E,P,
     1 DEX,DEY,DEZ,DEXY,F,THETA,DTHETA,ESC,FS,DSTL,SRS,
     2 ZEVP,TEVP,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
      DIMENSION SR(4),SRS(4),SR1(4),SR2(4),DEC(4),DES(4),DE(4),SR3(4),
     1 SR4(4),THET(6),IMC(6),IMS(6),FSTEEL(6),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,IMC(M),A4,IMS(M)
 1004 FORMAT(A10,E10.3,A10,E10.3,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))*
     1 (ESC(MC,2)+1.33*ESC(MC,5)))/(FSTEEL(M)*ESC(MS,1)+(1.-FSTEEL(M))*
     2 ESC(MC,1)))
      DH=FSTEEL(M)*ESC(MS,1)+(1.-FSTEEL(M))*ESC(MC,1)
      Y=ESC(MS,10)
      RETURN
15    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)
   18 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)
C        ROTATE STRAIN INCREMENTS TO AXIS OF REBARS
      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
C *******                                                    ***********
C        BEGINNING OF COMPUTATIONAL LOOP FOR EACH STRAIN INCREMENT
120   PS=PS1=-(SRS(1)+SRS(2)+SRS(3))/3.
      FS=FS1
      PC=PC1=(P-PS1*FS)/(1.-FS)
      DO 170 I=1,4
      SR1(I)=SRS(I)+PS1
      IF (I .EQ. 4) SR1(4)=SRS(4)
      SR2(I)=(SR(I)-SR1(I)*FS)/(1.-FS)
  170 DEC(I)=DES(I)=DE(I)*RR
      DES(1)=DEC(1)*ESC(MC,2)/ESC(MS,2)
      DEC(1)=(DE(1)*RR-DES(1)*FS)/(1.-FS)
```

```
  180 NC=0
C *******                                                    ***********
C       BEGINNING OF ITERATION LOOP
200   NC=NC+1
      DO 210 I=1,4
      SR3(I)=SR1(I)
  210 SR4(I)=SR2(I)
      TEVP=TEVPSV
      ZEVP=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=ROLD*(2.-DEST)/(2.+DEST)
      IF (IPRINT .EQ. 1) PRINT 1002,RH,ROLD,RX,RY,RZ,RXY,ZEVP,TEVP
      CALL CAP1(LS,IN,MC,N,IH,RH,ROLD,E,DEC(1),DEC(2), DEC(3),DEC(4),
     1  RX,RY,RZ,RXY,ZEVP,IC,JC,TEVP)
      IF (IPRINT .EQ. 1) PRINT 1003,RH,ROLD,RX,RY,RZ,RXY,ZEVP,TEVP
      PC=-(RX+RY+RZ)/3.
      SR4(1)=RX+PC $ SR4(2)=RY+PC $ SR4(3)=RZ+PC $ SR4(4)=RXY
      DEST=(DES(1)+DES(2)+DES(3))/3.
      D=DSTL*(2.-DEST)/(2.+DEST)
      CALL EPLAS(JC,IC,MS,SR3,PS,DES,ESC,D,Y)
      SCTEST= SR4(1)-PC   $   SSTEST = SR3(1)-PS
      IF (IPRINT .EQ. 1) PRINT 1001,NC,DES(1),DEC(1),PC,PS,(SR1(I),I=1,4
     1  ),(SR2(I),I=1,4),(SR3(I),I=1,4),(SR4(I),I=1,4),SCTEST,SSTEST
      IF (ABS(SR4(1)-PC-SR3(1)+PS) .LT. CRIT) GO TO 290
      DEZA=DES(1)    $    DSZA=SR4(1)-SR3(1)   -PC+PS
      IF (NC .EQ. 1) GO TO 250
      IF (NC .LT. 12) GO TO 260
      IF (NTRY .LT. 5) GO TO 450
C       ABORT PROVISION
      PRINT 1240,JC,IC,N,PS,PC,SSTEST,SCTEST,SR1,SR2,SR3,SR4,DES,DEC
 1240 FORMAT(1X,* ABORT IN REBAR FOR NTRY EQUALS 5 FOR J=*,I5,* I=*,I5,
     1 * ON CYCLE *,I5,/,1X,* PS=*,E10.3,* PC=*,E10.3,* SSTEST=*,E10.3,
     2 * SCTEST=*,E10.3/,* SR1=*,4E10.3,* SR2=*,4E10.3,/,* SR3=*,4E10.3,
     3 * SR4=*,4E10.3,/,* DES=*,4E10.3,* DEC=*,4E10.3)
      GO TO 320
C       PREPARATION FOR SECOND ITERATION
250   DES(1)=DES(1)+(SR4(1)-PC-SR3(1)+PS)/(ESC(MC,2)*FS/(1.-FS)+ESC(MS,2
     1  ))
      DEC(1)=(DE(1)*RR    -DES(1)*FS)/(1.-FS)
      GO TO 280
C       REGULA FALSI BRANCHES
260   IF (NC .EQ. 2) GO TO 262
      IF (DSZC .GT. 0.) GO TO 265
      IF (DSZB .LT. 0.) GO TO 262
      IF (DSZA .GT. 0.) GO TO 265
262   DES(1)=DEZA+(DEZB-DEZA)/(DSZB-DSZA)*(-DSZA)
      IF (NC .EQ. 6 .OR. NC .EQ. 10) DES(1)=0.5*(DEZA+DEZB)
      GO TO 270
265   DES(1)=DEZA+(DEZC-DEZA)/(DSZC-DSZA)*(-DSZA)
      IF (NC .EQ. 6 .OR. NC .EQ. 10) DES(1)=0.5*(DEZA+DEZC)
270   DEC(1)=(DE(1)*RR    -DES(1)*FS)/(1.-FS)
      IF (NC .GT. 2) GO TO 275
      IF (DSZA .LT. DSZB) 283,279
275   IF (DSZA .GT. DSZB .OR. DSZA .LT. DSZC) GO TO 277
      IF (DSZA .LT. 0.) 283,280
277   IF (DSZB .LT. 0. .AND. DSZA .GT. DSZB) GO TO 279
      IF (DSZC .GT. 0. .AND. DSZA .GT. DSZC) 282,200
279   DSZC=DSZB    $    DEZC=DEZB
280   DSZB=DSZA    $    DEZB=DEZA    $    GO TO 200
282   DSZB=DSZC    $    DEZB=DEZC
283   DSZC=DSZA    $    DEZC=DEZA    $    GO TO 200
C *******                                                    ***********
C       END OF ITERATION LOOP, RESET FOR NEXT STRAIN INCREMENT
290   DO 295 I=1,4
      SR1(I)=SR3(I)
295   SR2(I)=SR4(I)
      IHSV=IH
      YSV=Y
      TEVPSV=TEVP
      ZEVPSV=ZEVP
      FS=FS*(1.+DES(1))/(FS*(1.+DES(1))+(1.-FS)*(1.+DEC(1)))
```

SUBROUTINE REBAR (Concluded)

```
      DSTL=D
      ROLD=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(I)=SR4(I)*(1.-FS)+SR3(I)*FS
330   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
C        PROVISION TO CUT STRAIN INCREMENTS
450   NTRY=NTRY+1
      IF (NTRY .EQ. 5) IPRINT=1
      RR=RR/3.
      GO TO 120
1001  FORMAT(1X,* 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,ROLD=*1P2E10.3,
     1  * RX,RY,RZ,RXY=*4E10.3,* ZEVP,TEVP=*2E10.3)
1003  FORMAT (* AFTER CAP,  RH,ROLD=*1P2E10.3,
     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

```
      SUBROUTINE REDR(NA,NB,IN,NO)
C        TO PREPARE TAPE4 FOR READING, COPY FROM INPUT TO TAPE4 WITH THE
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
C        CARDS BUT THERE ARE NO SEPARATORS BETWEEN MATERIALS.
      REWIND IN
      NN=0
      IDD=1H
10    READ (IN,100) IND,NAA,NBB
      NN=NN+1
      IF (EOF(IN)) 15,20
15    PRINT 110,IDD,NA,NB,IND,NAA,NBB
      STOP 2254
20    IF (NA.NE.NAA .OR. (NB.NE.NBB .AND. NO.EQ.2)) GO TO 10
      REWIND IN
      NN=NN-1
      IF (NN .EQ. 0) RETURN
      DO 40 N=1,NN
      READ (IN,100) IND
40    CONTINUE
      RETURN
100   FORMAT (A1,A9,A10)
110   FORMAT (15H SEARCHING FOR A1,A9,A10,8H, FOUND A1,A9,A10)
      END
```

SUBROUTINE RELAX

```
      SUBROUTINE RELAX (ICON,SD,Y2,DRO,COEF,N,J,M,ANM,ANT,DT,TSR,YD,
     1 Y1,INSR)
C
C     CALLED BY HSTRESS TO COMPUTE DEVIATOR STRESS ACCORDING TO
C     STANDARD ANELASTIC AND TWO-PARAMETER YIELD MODELS, NDS=1 AND 4.
C     INPUT - ALL FORMAL PARAMETERS
C     OUTPUT - SD, ICON, YOLD=YNEW.
C
      DIMENSION TSR(6,30)
      YOLD =0.6667*Y2
      YAD = 0.6667*YD
      YNOT= 0.6667*Y1
C
      TRLX=TSR(M,15)   $   TY=TSR(M,16)   $   SDO=SD   $   ICOR=ICON
      YNEW=YOLD        $   L=0
      IN=MAX0(2,INSR)/2
      IF (ICON .EQ. 2)18,2
C     INITIAL CONDITION OUTSIDE OF ELASTIC ZONE
2     L=1  $ GO TO (4,3)IN
3     XPY=EXP(-DT/TY)
      YNEW=YNOT+(YOLD-YNOT)*XPY+ABS(COEF)*TY/DT*(1.-XPY)*.5*(1.+SIGN(1.,
     1DRO)*SIGN(1.,SDO))
      YAVG = (YNEW+YOLD)/2.   $   YSTAR = SIGN(YAVG,SDO)   $ GO TO 5
4     YSTAR = SIGN(YOLD,SDO)
5     XPO=EXP(-DT/TRLX)
      SD = YSTAR + (SDO-YSTAR)*XPO + COEF*TRLX/DT*(1.-XPO)
C     CHECK IF DEVIATOR CROSSES INTO ELASTIC ZONE. IF SO, RECALCULATE RE
      IF(ABS(SD).GE.YNEW.AND.SIGN(1.,SD).EQ.SIGN(1.,SDO))30,6
6     TC = (SIGN(YNEW,SDO)-SDO)/(SD-SDO)*DT   $ L=2
      GO TO (9,7)IN
7     YNEW=YNOT+(YOLD-YNOT)*EXP(-TC/TY)
      YAVG = (YNEW+YOLD)/2.   $   YSTAR = SIGN(YAVG,SDO)
9     XPO = EXP(-TC/TRLX)
      SD=YSTAR+(SDO-YSTAR)*XPO+COEF*TRLX/DT*(1.-XPO)+COEF *(DT-TC)/DT
C     CHECK IF DEVIATOR CROSSES OVER INTO OTHER SIDE OF ZONE
      IF (ABS(SD).GT.YNEW)11,10
10    ICON = 2  $ L=3  $ GO TO 35
11    IF (SIGN(1.,SD).EQ.SIGN(1.,SDO))30,12
C     RECALCULATE TIME DURING WHICH RELAXATION OCCURS
12    TK = (SD+SIGN(YNEW,SDO))/(SD-SIGN(YNEW,SDO))*(DT-TC) $L=4
      GO TO (25,13)IN
13    XPY=EXP(-TK/TY)
      YNEW=YNOT+(YNEW-YNOT)*XPY+ABS(COEF)*TY/DT*(1.-XPY)
      YAVG = (YNEW+YOLD)/2. $ YSTAR = SIGN(YAVG,SD)  $ GO TO 25
C     NOW CONSIDER INITIAL CONDITIONS INSIDE ELASTIC ZONE
18    SD =SDO + COEF
C     CHECK IF DEVIATOR CROSSES ZONE BOUNDARY
      IF (ABS(SD).GT.YOLD)19,35
C     CHANGE CONDITION VARIABLE AND RECALCULATE DEVIATOR WITH RELAXATION
19    YSTAR =  SIGN(YOLD,SD) $ L=5  $ TK=(SD-YSTAR)/(SD-SDO)*DT
      GO TO (25,20)IN
20    YNEW=YNOT+ABS(COEF)*TY/DT*(1.-EXP(-TK/TY))
      YAVG = (YNEW+YNOT)/2.   $   YSTAR = SIGN(YAVG,SD)
   25 ICON = 2 - IFIX(SIGN(1.,SD))
      SD =  YSTAR +COEF*TRLX/DT*(1-EXP(-TK/TRLX))
30    GO TO (31,35)IN
C     RECALCULATE YIELD STRENGTH TO ACCOUNT FOR STRAIN HARDENING
 31   YNEW = AMIN1(AMAX1(ABS(SD),YOLD),YOLD+YAD*ABS(DRO))
      IF (YNEW.EQ.ABS(SD)) 32,35
32    ICON=2   $ L=L+10
   35 CONTINUE
      Y2 = 1.5* YNEW
      RETURN
      END
```

SUBROUTINE REZONE

```
      SUBROUTINE REZONE                                                 REZONE     2
C                                                                       REZONE     3
C     INCREASES CELL SIZES TO GIVE MORE UNIFORM DISTRIBUTION            REZONE     4
C        *  STARTS REZONING AT JREZON AND WORKS TOWARD JINIT            REZONE     5
C        *  DOES NOT DISTURB LOCATION OF INTERFACES, JEDITS, OR SPALLS  REZONE     6
C                                                                       REZONE     7
C     INPUT - NREZON, SSTOPM.                                           REZONE     8
C     OUTPUT - ARRAY VARIABLES X, C, CHL, D, DHL, EHL, H, NEM, NET, P, P REZONE     9
C           R, S, SHL, T, U, UHL, YHL, ZHL, AND JEDIT, JBND.            REZONE    10
C                                                                       REZONE    11
      INTEGER H,POROUS,PRESS,RINTER,SOLID,SPALL                         PUFCOM     2
      REAL MATL,NEM,NET,NEMH,NETH                                       PUFCOM     3
C               MISCELLANEOUS                                           PUFCOM     4
      COMMON AZERO(1),CEF,CKS,DAVG,DELTIM,DISCPT(10),DOLD,DRHO,DTMAX,   PUFCOM     5
     1  DTMIN,DTN,DTNH,DU,DX,EOLD,F,FAC,FIRST,J,JCYCS,JINIT,            PUFCOM     6
     2  JFIN,JREZON(15),JSMAX,JSTAR,JTS,LSUB(30),M,MAXPR(30),N,NCYCS,   PUFCOM     7
     3  NEDIT,NPERN,NR,NREZON,NSCRB(6),NSEPRAT,NSPALL,NTEDT,            PUFCOM     8
     4  NTEX,NTR(15),POLD,P6(20),R(30),RLAST,SLAST,SMAX,TEDIT(50),      PUFCOM     9
     5  TF,TIME,TJ,TREZON,TS,T6(20),ULAST,UOLD,UZERO,XLAST,XNOW,XOLD    PUFCOM    10
     1  ,XJDIT(20),MS                                                   PUFCOM    11
C               HALFSTEP VALUES                                         PUFCOM    12
      COMMON DH,DHLAST,DUH,EH,PH,RH,RHLAST,SH,SHLAST,UH,UHLAST,XH,XHLAST PUFCOM    13
     1  ,NEMH,NETH                                                      PUFCOM    14
C               CONDITION INDICATORS                                    PUFCOM    15
      COMMON INF,LINTER,MIRROR,NORMAL,POROUS,PRESS,RINTER,SOLID,SPALL   PUFCOM    16
C               CELL LAYOUT                                             PUFCOM    17
      COMMON DXX(30),JBND(30),JMAT(30),NAUTO,MATL(6,2),NLAYER,NMTRLS,   PUFCOM    18
     1  THK(30)                                                         PUFCOM    19
C                                                                       PUFCOM    20
C               COORDINATE ARRAYS                                       COORDCOM   2
      COMMON/COORD/X(200),XO(200),CHL(200),DHL(200),DPDD(200),DPDE(200), COORDCOM   3
     1  EHL(200),H(200,3),NEM(200),NET(200),PHL(200),RHL(200),SDT(200), COORDCOM   4
     2  SHL(200),T(200),U(200),YHL(200),ZHL(200)                        COORDCOM   5
      COMMON /JED/JEDIT(100),JNUM(100),JTYP(100),NAME2(40),JEDSIZ,      JEDCOM     2
     1  MODLUS,NERR,NJEDIT,NTAPE                                        JEDCOM     3
C               NAMED COMMON                                            EQSTCOM    2
      REAL MU,MUM                                                       EQSTCOM    3
      COMMON /EQS/  EQSTA(6),EQSTC(6),EQSTD(6),EQSTE(6),EQSTG(6),       EQSTCOM    4
     1  EQSTH(6),EQSTN(6),EQSTS(6),EQSTV(6),CZQ(6),CWQ(6),C2(6)         EQSTCOM    5
      COMMON /MELT/ EMELT(6,8),GMELT(6,8),SPH(6),THERM(6,8)             EQSTCOM    6
      COMMON /RHO/ RHO(6),RHOS(6)                                       EQSTCOM    7
      COMMON /TSR/ TSR(6,30),EXMAT(6,20),TENS(6,3)                      EQSTCOM    8
      COMMON /Y/ YO(6),YADD(6),MU(6),MUM,YADDM                          EQSTCOM    9
      COMMON /IND/ IEOS(6),INDK(20),NALPHA,NCMP(6),NFR(6),NPOR(6),      INDCOM     2
     1  NDS(6),NPR(6),NCON(6),NVAR(6)                                   INDCOM     3
      COMMON /RAD/ SSTOP(9),START(9),SDURM,SSTOPM,NSPEC,SSJ,JSS,IPLOT(4) RADCOM     2
     1  ,XMAX(4),XMIN(4),YMAX(4),YMIN(4),IA(7),ITITLE(24),NARZ,TARZ     RADCOM     3
      COMMON/SS/SS(500)                                                 SSCOM      2
      COMMON/PES/COM(2000),LVAR(200),LVMAX                              REZONE    19
C                                                                       REZONE    20
      DIMENSION CC(20),EC(20),HC(20,3),MASS(21),MOM(20,2),DC(20),       10/8/79    4
     1  PC(20),RC(21),SC(20),XC(20),YC(20),ANEM(20),ANET(20)            REZONE    22
      DIMENSION NEWJED(100)                                             REZONE    23
      DIMENSION ASC(20),PSC(20),RSC(20),RVSC(20),ENSC(20)               REZONE    24
      DIMENSION SSS(5),SSC(20,5)                                        REZONE    25
      INTEGER HC,HJOLD2                                                 REZONE    26
C                                                                       REZONE    27
      REAL MASS,MOM,MASLAST,MOMLAST,MASNEXT                             REZONE    28
C                                                                       REZONE    29
C                                                                       REZONE    30
1     CALL SECOND(XNOW)                                                 REZONE    31
C                                                                       REZONE    32
C          SECTION 1 - LOCATE JREZON WITH RESPECT TO MATERIAL AND JEDITS REZONE    33
C                                                                       REZONE    34
      JREZ=JREZON(NR)                                                   REZONE    35
      DTD=0.                                                            REZONE    36
3     IF (NREZON .GT. 0)  GO TO 7                                       REZONE    37
      IF (JTS .GE. JFIN-2)  RETURN                                      REZONE    38
      CALL EDIT                                                         REZONE    39
      DTS=(X(JTS+1)-X(JTS))/CHL(JTS)                                    REZONE    40
      DTD=AMIN1(2.*DTS,1.4*DTMAX)                                       REZONE    41
      JREZ=JTS                                                          REZONE    42
      JBEG=1                                                            REZONE    43
      DO 5 L=1,NLAYER                                                   REZONE    44
      JBNDM=JBND(L)-1                                                   REZONE    45
```

```
      DO 4 JS=JBEG,JBNDM
      IF((X(JS+1)-X(JS))/CHL(JS) .GT. DTD) GO TO 4
      JREZ=JS+1
4     CONTINUE
      JBEG=JBNDM+2
5     CONTINUE
7     JREZ=JLAST=MIN0(JREZ,JFIN-1)
      IF (JREZ .LE. JINIT+1) RETURN
      L=0
8     L=L+1
      IF (JREZ .GT. JBND(L)+1)  GO TO 8
      IF (JREZ .EQ. JBND(L)+1)  JLAST=JLAST-1
      MASLAST=ZHL(JLAST)
      MOMLAST=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 QUANTITIES ARE
C     **  SET 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
11    L1=L-1  $  DXX(L)=DX=AMAX1(DX,DTD*CHL(JLAST-1) )
      DT=DX/CHL(JLAST-1)
      JOLD=JNEW=JLAST-1  $  NCEL=NPART=0
      IF (L .EQ. 1)   GO TO 13
      M=JMAT(L)
      DO 12 I=1,L1
      MI=JMAT(I)
12    DXX(I)=DXX(L)*SQRT(EQSTC(MI)*RHOS(M)/(EQSTC(M)*RHOS(MI)))
13    DO 10 I=1,NJEDIT
10    NEWJED(I)=JEDIT(I)
      LOC=11
      WRITE (6,5011) JREZ,DT,DTNH
      WRITE (6,5000) LOC,JOLD,JNEW,JLAST,L,NJ,NCEL,NPART
      WRITE (6,5015) (I,DXX(I),I=1,L)
C
C           SECTION 2 - FIND REZONABLE SET OF CELLS
C
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   H(JNEW+1,2)=HJOLD2  $  X(JNEW)=X(JOLD-1)
      L=L-1  $  JBND(L)=JNEW  $  TLAST=T(JOLD-1)  $  JNEW=JNEW-1
      JOLD=JOLD-2  $  XN=X(JOLD)
      LOC=125
      WRITE (6,5000) LOC,JOLD,JNEW,JLAST,L,NJ,NCEL,NPART
      GO TO 50
C
C           TERMINATION AT INITIAL BOUNDARY  (PART 2)
 155  IF(JOLD-JINIT) 790,160,255
160   NPART=2  $  HJOLD2=H(JOLD,2)  $  GO TO 500
200   H(JNEW+1,2)=HJOLD2  $  DO 205 NJ=1,NJEDIT
      IF (JOLD .NE. JEDIT(NJ)) GO TO 205
      NEWJED(NJ)=JNEW+1
      GO TO 800
205   CONTINUE
      GO TO 800
C
255   CONTINUE
300   CONTINUE
C
C           TERMINATION WHEN NUMBER OF REZONABLE OLD CELLS IS 20  (PART
355   IF ((X(JLAST)-X(JOLD))/DXX(L)-18.)  420,360,360
360   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
      WRITE (6,5000) LOC,JOLD,JNEW,JLAST,L,NJ,NCEL,NPART
      GO TO 50
```

SUBROUTINE REZONE (Continued)

```
420   JOLD=JOLD-1  $  GO TO 50                                          REZONE   121
C                                                                       REZONE   122
C          SECTION 3 - COMPUTE NEW CELL COORDINATES AND PROPERTIES      REZONE   123
C                                                                       REZONE   124
500   NQ=0                                                              REZONE   125
      LOC=500                                                           REZONE   126
      WRITE (6,5000) LOC,JOLD,JNEW,JLAST,L,NJ,NCEL,NPART                REZONE   127
510   NCEL=MINO(20,MAX1((X(JLAST)-X(JOLD))/DXX(L)+.65,1.))              REZONE   128
      IF ((NCEL-1)*(NQ-1) .EQ. 0) GO TO 610                             REZONE   129
C              CHECK WHETHER REGION OF LARGE CELLS LIES TO LEFT         REZONE   130
601   DXMIN=DXX(L)  $  JLASTP=JLAST-1                                   REZONE   131
      LOC=601                                                           REZONE   132
      WRITE (6,5000) LOC,JOLD,JNEW,JLAST,L,NJ,NCEL,NPART                REZONE   133
      DO 603 JX=JOLD,JLASTP                                             REZONE   134
        DELX=X(JX+1)-X(JX)                                              REZONE   135
      IF (DELX-DXMIN) 602,603,603                                       REZONE   136
602   DXMIN=DELX  $  JXMIN=JX                                           REZONE   137
603   CONTINUE                                                          REZONE   138
      IF (DXMIN-0.8*DXX(L)) 604,750,750                                 REZONE   139
604   JX=JXMIN+1                                                        REZONE   140
      DO 605 I=JOLD,JXMIN                                               REZONE   141
      JX=JX-1  $ DELX=X(JX+1)-X(JX)                                     REZONE   142
      IF (DELX-DXX(L)) 605,605,608                                      REZONE   143
605   CONTINUE  $  GO TO 610                                            REZONE   144
608   JOLD=JX+1  $  NPART=4                                             REZONE   145
      LOC=608                                                           REZONE   146
      WRITE (6,5000) LOC,JOLD,JNEW,JLAST,L,NJ,NCEL,NPART                REZONE   147
      NQ=1  $  GO TO 510                                                REZONE   148
C                                                                       REZONE   149
C              BEGIN COMPUTATIONS FOR NEW COORDINATES                   REZONE   150
610   NCEL=MINO(JLAST-JOLD,NCEL)                                        REZONE   151
      JOLDR=JLAST                                                       REZONE   152
      DX=(X(JLAST)-X(JOLD))/NCEL                                        REZONE   153
      XSTART=X(JLAST)  $  XN=XSTART-DX                                  REZONE   154
C     **  XN IS NEW COORDINATE LOCATION                                 REZONE   155
C     **  DX IS NEW CELL DIMENSION                                      REZONE   156
      MOM(1,1)=MOMLAST                                                  REZONE   157
      MASS(1)=MASLAST                                                   REZONE   158
      LOC=610                                                           REZONE   159
      WRITE ( 6,5002) LOC,NCEL,XSTART,DX,XN,RSLAST,MASLAST,MOMLAST      REZONE   160
      WRITE ( 6,5610)                                                   REZONE   161
      M=JMAT(L)                                                         REZONE   162
      IF (NALPHA .GT. 1) XSTART=XSTART**NALPHA                          10/8/79    6
      XNAOLD=XSTART                                                     10/8/79    7
      DO 650 I=1,NCEL                                                   REZONE   163
      MASS(I+1)=AMAVG=AMSLP=ENGY=CS=RS=PS=SX=YS=0.  $  ASUM=PSUM=RSUM=  REZONE   164
     1  RVSUM=ENSUM=0.                                                  REZONE   165
      XNAVALF=XN+DX/2.                                                  10/8/79    8
      DXALF=DX                                                          10/8/79    9
      IF (NALPHA .LE. 1) GO TO 611                                      10/8/79   10
      XNA=XN**NALPHA                                                    10/8/79   11
      XNAVALF=0.5*(XNA+XNAOLD)                                          10/8/79   12
      DXALF=XNAOLD-XNA                                                  10/8/79   13
      XNAOLD=XNA                                                        10/8/79   14
611   CONTINUE                                                          10/8/79   15
      DO 612 INS = 1,NSPEC                                              REZONE   166
612   SSS(INS) = 0.                                                     REZONE   167
      HC(I,1)=SOLID                                                     REZONE   168
      HC(I,2)=NORMAL                                                    REZONE   169
      ANEMS=ANETS=0.                                                    REZONE   170
      HC(I,3)=2                                                         REZONE   171
615   IF (JLAST .LT. 1) GO TO 625                                       REZONE   172
      XEND1=XEND=AMAX1(X(JLAST),XN)                                     10/8/79   16
      XJLALF1=X(JLAST+1)                                                10/8/79   17
      XJLALF=X(JLAST)                                                   10/8/79   18
      IF (NALPHA .LE. 1) GO TO 616                                      10/8/79   19
      XEND=XEND**NALPHA                                                 10/8/79   20
      XJLALF1=XJLALF1**NALPHA                                           10/8/79   21
      XJLALF=XJLALF**NALPHA                                             10/8/79   22
616   CONTINUE                                                          10/8/79   23
      IF (XSTART-XEND) 621,621,619                                      REZONE   174
619   DMASS=ZHL(JLAST)*(XSTART-XEND)/(XJLALF1-XJLALF)                   10/8/79   24
      MASS(I+1)=MASS(I+1)+DMASS                                         REZONE   176
      UJ=U(JLAST)                                                       REZONE   177
      DUOLD=U(JLAST+1)-UJ                                               REZONE   178
```

SUBROUTINE REZONE (Continued)

```
      DXOLD=XJLALF1-XJLALF                                               10/8/79    25
      XS1=0.5*(XSTART+XEND)-XNAVALF                                      10/8/79    26
      XS2=XSTART-XEND                                                    REZONE    181
      U1=UJ+DUOLD*(XEND-XJLALF)/DXOLD                                    10/8/79    27
      U2=UJ+DUOLD*(XSTART-XJLALF)/DXOLD                                  10/8/79    28
      AMAVG=0.25*DMASS*(U1+U2) + AMAVG                                   REZONE    184
      AMSLP=DMASS/DXALF*(1.5*(U2+U1)*XS1+0.25*(U2-U1)*XS2)+AMSLP         10/8/79    29
      ENGY=ENGY+DMASS*EHL(JLAST)                                         REZONE    186
      IF (TIME .GT. SSTOPM) GO TO 620                                    REZONE    187
      DO 6201 INS=1,NSPEC                                                REZONE    188
      JF=JFIN*(INS-1)+JLAST                                              REZONE    189
 6201 SSS(INS)=SSS(INS)+DMASS*SS(JF)                                     REZONE    190
620   CONTINUE                                                           REZONE    191
      RS=RS+DMASS*RHL(JLAST)                                             REZONE    192
      PS=PS+DMASS*PHL(JLAST)                                             REZONE    193
      SX=SX+DMASS*SHL(JLAST)                                             REZONE    194
      YS=YS+DMASS*YHL(JLAST)                                             REZONE    195
      CS=CS+DMASS*CHL(JLAST)                                             REZONE    196
      ANEMS=ANEMS+DMASS*NEM(JLAST)                                       REZONE    197
      ANETS=ANETS+DMASS*NET(JLAST)                                       REZONE    198
      LL=LVAR(JLAST)                                                     REZONE    199
      IF (LL .EQ. 0) GO TO 6205                                          REZONE    200
      ASUM=ASUM+DMASS*COM(LL+2)                                          REZONE    201
      PSUM=PSUM+DMASS*COM(LL+1)                                          REZONE    202
      RSUM=RSUM+DMASS*COM(LL)                                            REZONE    203
      RVSUM=RVSUM+DMASS*COM(LL+3)                                        REZONE    204
      ENSUM=ENSUM+DMASS*COM(LL+4)                                        REZONE    205
 6205 CONTINUE                                                           REZONE    206
      HC(I,2)=MIN0(HC(I,2),H(JLAST,2))                                   REZONE    207
      XSTART=XEND                                                        REZONE    208
      IF ((H(JLAST,1) .EQ. 5R    P .OR. H(JLAST,1) .EQ. 5R    Q .OR. H   REZONE    209
     1  (JLAST,1) .EQ. 5R    T) .AND. HC(I,1) .EQ. SOLID)HC(I,1)=POROUS  REZONE    210
      IF (H(JLAST,1) .EQ. 5R    R .AND. HC(I,1) .NE. 5R    Z)HC(I,1)=5R  REZONE    211
     1   R                                                               REZONE    212
      IF (H(JLAST,1) .EQ. 5R    Z)HC(I,1)=5R    Z                        REZONE    213
      HC(I,3)=MAX0(H(JLAST,3),HC(I,3))                                   REZONE    214
621   IF (XEND1 .LE. XN) GO TO 625                                       10/8/79    30
      JLAST=JLAST-1  $  GO TO 615                                        REZONE    216
625   XC(I)=XN  $  DC(I)=MASS(I+1)/DXALF  $  EC(I)=ENGY/MASS(I+1)        10/8/79    31
      YC(I)=YS/MASS(I+1) $  SC(I)=SX/MASS(I+1) $  PC(I)=PS/MASS(I+1)     REZONE    218
      CC(I)=CS/MASS(I+1)  $  RC(I)=RS/MASS(I+1)                          REZONE    219
      ASC(I) = ASUM/MASS(I+1)  $  PSC(I) = PSUM/MASS(I+1)                REZONE    220
      RSC(I) = RSUM/MASS(I+1)  $  RVSC(I) = RVSUM/MASS(I+1)              REZONE    221
      ENSC(I) = ENSUM/MASS(I+1)                                          REZONE    222
      IF (TIME .GT. SSTOPM) GO TO 630                                    REZONE    223
      DO 628 INS=1,NSPEC                                                 REZONE    224
628   SSC(I,INS)=SSS(INS)/MASS(I+1)                                      REZONE    225
630   CONTINUE                                                           REZONE    226
      MOM(I,2)=AMAVG+AMSLP                                               REZONE    227
      MOM(I+1,1)=AMAVG-AMSLP                                             REZONE    228
      ANEM(I)=ANEMS/MASS(I+1)  $  ANET(I)=ANETS/MASS(I+1)                REZONE    229
643   K=JNEW+1-I                                                         REZONE    230
      LOC=643                                                            REZONE    231
      WRITE (6,5003) LOC,K,XC(I),DC(I),MOM(I,2),MOM(I+1,1),EC(I),RC(I),  10/8/79    32
     1  PC(I),SC(I),YC(I),MASS(I+1),HC(I,1)                              REZONE    233
650   XN=AMAX1(XN-DX,X(JOLD))                                            REZONE    234
      T(JNEW+1)=TLAST                                                    REZONE    235
      DO 6550 NJD=1,NJEDIT                                               REZONE    236
      IF (JEDIT(NJD) .GT. JOLDR .OR. JEDIT(NJD) .LT. JOLD) GO TO 6550    REZONE    237
      JED=JEDIT(NJD)                                                     REZONE    238
      NEWJED(NJD)=JNEW+1-NCEL                                            REZONE    239
      XJED=0.5*(X(JED)+X(JED+1))                                         REZONE    240
      DO 6545 I=2,NCEL                                                   REZONE    241
      IF (XJED .LT. 0.5*(XC(I)+XC(I-1))) GO TO 6545                      REZONE    242
      NEWJED(NJD)=JNEW+2-I                                               REZONE    243
      GO TO 6550                                                         REZONE    244
6545    CONTINUE                                                         REZONE    245
6550    CONTINUE                                                         REZONE    246
      DO 670 I=1,NCEL                                                    REZONE    247
      J=JNEW+1-I         $  CHL(J)=CC(I)        $  DHL(J)=DC(I)          10/8/79    33
      EHL(J)=EC(I)        $  PHL(J)=PC(I)        $  SHL(J)=SC(I)         REZONE    249
      YHL(J)=YC(I)        $  ZHL(J)=MASS(I+1)    $  H(J,1)=HC(I,1)       REZONE    250
      NET(J)=ANET(I)  $  NEM(J)=ANEM(I)  $  RHL(J)=RC(I)                 REZONE    251
      JL=JOLDR+1-I                                                       REZONE    252
      LVAR(J)=LVAR(JL)                                                   REZONE    253
```

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. SSTOPM) GO TO 660
      DO 655 INS=1,NSPEC
      JF=JFIN*(INS-1)+J
  655 SS(JF)=SSC(I,INS)
 660  CONTINUE
      U(J+1)=2.*(MOM(I,1)+MOM(I,2))/(MASS(I)+MASS(I+1))
      T(J)=TENS(M,1)  $  X(J)=XC(I)  $  H(J,2)=HC(I,2)
      H(J,3)=HC(I,3)
 670  CONTINUE
      MOMLAST=MOM(NCEL+1,1)
      MASLAST=MASS(NCEL+1)
      TLAST=T(JOLD)
      GO TO (680,680,700,700,685) NPART
 680  CONTINUE
 685  T(J)=TLAST
      U(J)=2.*MOMLAST/MASLAST
      MOMLAST=MASLAST=RSLAST=0.
 700  CONTINUE
      LOC=700
      WRITE (6,5000) LOC,JOLD,JNEW,JLAST,L,NJ,NCEL,NPART
C     SET JNEW AND JLAST IN PREPARATION FOR THE NEXT ZONE CALCULATIONS
      JNEW=J-1  $  JLAST=JOLD
C     RETURN TO APPROPRIATE PART OF REZONE FOR FINAL RESETTING
      GO TO (100,200,300,400) NPART
C
C                RENUMBER CELLS WITHOUT REZONING
 750  T(JNEW+1)=TLAST
      LOC=750
      TLAST=T(JOLD)
      WRITE ( 6,5750)
 752  JLAST=JLAST-1              $  DHL(JNEW)=DHL(JLAST)
      EHL(JNEW)=EHL(JLAST)      $  PHL(JNEW)=PHL(JLAST)
      IF (TIME .GT. SSTOPM) GO TO 754
      DO 753 INS=1,NSPEC
      JF=JFIN*(INS-1)
  753 SS(JF+JNEW)=SS(JF+JLAST)
 754  CONTINUE
      SHL(JNEW)=SHL(JLAST)      $  YHL(JNEW)=YHL(JLAST)
      CHL(JNEW)=CHL(JLAST)      $  ZHL(JNEW)=ZHL(JLAST)
      H(JNEW,1)=H(JLAST,1)      $  H(JNEW,2)=H(JLAST,2)
      MASNEXT=ZHL(JLAST)  $  RHL(JNEW)=RHL(JLAST)
      U(JNEW+1)=(2.*MOMLAST+MASNEXT*U(JLAST+1))/(MASLAST+MASNEXT)
      MASLAST=MASNEXT
      MOMLAST=.5*MASLAST*U(JLAST)
      T(JNEW)=T(JLAST)          $  X(JNEW)=X(JLAST)
      NEM(JNEW)=NEM(JLAST)      $  NET(JNEW)=NET(JLAST)
      H(JNEW,3)=H(JLAST,3)
      LVAR(JNEW)=LVAR(JLAST)
      DO 7550 NJD=1,NJEDIT
      IF (JEDIT(NJD) .EQ. JLAST) NEWJED(NJD)=JNEW
 7550  CONTINUE
      I=JNEW                    $  JNEW=JNEW-1
      WRITE ( 6,5003) LOC,I,X(I),DHL(I),U(I+1),EHL(I),RHL(I),PHL(I),
     1  SHL(I),YHL(I),T(I),ZHL(I),H(I,1)
      IF (JLAST-JOLD) 790,755,752
C     **  JNEW IS TO LEFT OF LAST RENUMBERED CELL. JLAST=JOLD, THE LAST
C     **  OLD COORDINATE RENUMBERED.
 755  CONTINUE
      LOC=755
      WRITE (6,5000) LOC,JOLD,JNEW,JLAST,L,NJ,NCEL,NPART
      GO TO (760,760,300,400,765) NPART
 760  CONTINUE
 765  U(JNEW+1)=2.*MOMLAST/MASLAST
      T(JNEW+1)=T(JLAST)
      MOMLAST=MASLAST=RSLAST=0.
      LOC=760
      WRITE (6,5000) LOC,JOLD,JNEW,JLAST,L,NJ,NCEL,NPART
```

SUBROUTINE REZONE (Concluded)

```
      GO TO (100,200,300,400) NPART
C
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
C           ENDING ROUTINE - INTERFACE AND BOUNDARY ADJUSTMENTS
C
800   JINIT=JNEW+1
      IF (H(JINIT,2) .EQ. SPALL)  R(1)=0.
      DO 820 L=1,NLAYER
      JB=JBND(L)  $  H(JB,2)=LINTER
      IF (H(JB+1,2) .EQ. SPALL)  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))
820   CONTINUE
      WRITE (6,5825)
      WRITE (6,5826) (JEDIT(NJ),NJ=1,NJEDIT)
      WRITE (6,5827) (NEWJED(NJ),NJ=1,NJEDIT)
      DO 825 I=1,NJEDIT
825   JEDIT(I)=NEWJED(I)
840   CALL EDIT
900   CONTINUE
      CALL SECOND(TWIX)  $  DUR=TWIX-XNOW
      WRITE ( 6,5010) JINIT,DUR
      RETURN
1000  FORMAT (24H ERROR IN REZONE, NPART=I3,6H JOLD=I3,6H JNEW=I3,
     1  7H JLAST=I3,4H NJ=I3,11H JEDIT(NJ)=I3,3H L=I3,9H JBND(L)=I3)
5000  FORMAT (13H REZONE, LOC=I3,7H, JOLD=I3,7H, JNEW=I3,8H, JLAST=I3,
     1  4H, L=I3,5H, NJ=I3,7H, NCEL=I3,8H, NPART=I3)
 5002 FORMAT(13H REZONE, LOC=I3,7H, NCEL=I3,15H, XSTART,DX,XN=1P3E10.3,
     1  9H, RSLAST=1PE10.3,10H, MASLAST=1PE10.3,10H, MOMLAST=1PE10.3)
 5003 FORMAT(2I5,1P10E10.3,3X,R1)
 5010 FORMAT(19H0END REZONE, JINIT=I3,17H, TIME IN REZONE=1PE10.3,5H SEC
     1 )
5011  FORMAT (//5X,*BEGINNING OF REZONING,  JREZ=*I3,*, INTENDED DT=*
     1  1PE10.3,*, DTNH=*1PE10.3)
 5015  FORMAT(21H REZONE, DXX VALUES   ,6(I8,F9.6)/21X,6(I8,F9.6))
5610  FORMAT (120H  LOC     J          XC          DC   MOM(I,2) MOM(I+1,1)
     1     EC       RC           PC          SC          YC MASS(I+1)   HC(I,1)
     2)
5750  FORMAT (120H  LOC     J          X          DHL     U(I+1)         EHL
     1 RHL         PHL         SHL         YHL         T          ZHL     H(I,1)
     2)
5825  FORMAT (*  TRANSFORMATION OF JEDIT VALUES*)
5826  FORMAT (* OLD JEDITS = *18I5)
5827  FORMAT (* NEW JEDITS = *18I5)
      END
```

SUBROUTINE SCATTO

```
      SUBROUTINE SCATTO (XS,ES,ECAL,NPOINT,NS,L,ESUM)                  SCATTO 2
C                                                                      SCATTO 3
C        CALLED BY OEPDS  TO DISTRIBUTE ABSORBED ENERGY INTO PUFF CELLS.SCATTO 4
C        THE ENERGY HAS BEEN PREVIOUSLY COMPUTED BY A FLUORESCENCE AND  SCATTO 5
C        SCATTERING CODE (SUCH AS FSCATT OF S.S.S.) OR DETERMINED       SCATTO 6
C        EXPERIMENTALLY.  DEPOSITION COORDINATES MAY BE SPACED ARBI-    SCATTO 7
C        TRARILY.  INTERPOLATION FUNCTION IS A PARABOLA IN LOG(E) VS X  SCATTO 8
C        FITTED THROUGH 3 DEPOSITION COORDINATES.                       SCATTO 9
C     INPUT - FORMAL PARAMETERS WHICH CONTAIN COORDINATES (XS) AT WHICH SCATTO10
C          DOSE (ES) IS KNOWN, NPOINT (NUMBER OF COORDINATE POINTS, ECALSCATTO11
C          (CALIBRATION FACTOR), AND NS (SPECTRUM NUMBER).  ES SHOULD   SCATTO12
C          BE IN CAL/CM2, XS IN CM.                                     SCATTO13
C     OUTPUT - FILLS SS ARRAY.                                          SCATTO14
C                                                                       SCATTO15
      INTEGER H,PORDUS,PRESS,RINTER,SOLIO,SPALL                         PUFCOM 2
      REAL MATL,NEM,NET,NEMH,NETH                                       PUFCOM 3
C              MISCELLANEOUS                                            PUFCOM 4
      COMMON AZERO(1),CEF,CKS,DAVG,DELTIM,OISCPT(10),DDLD,DRHO,DTMAX,   PUFCOM 5
     1  DTMIN,OTN,DTNH,OU,DX,EOLD,F,FAC,FIRST,J,JCYCS,JINIT,            PUFCOM 6
     2  JFIN,JREZON(15),JSMAX,JSTAR,JTS,LSUB(30),M,MAXPR(30),N,NCYCS,   PUFCOM 7
     3  NEDIT,NPERN,NR,NREZON,NSCRB(6),NSEPRAT,NSPALL,NTEDT,            PUFCOM 8
     4  NTEX,NTR(15),PDLD,P6(20),R(30),RLAST,SLAST,SMAX,TEDIT(50),      PUFCOM 9
     5  TF,TIME,TJ,TREZON,TS,T6(20),ULAST,UOLD,UZERO,XLAST,XNOW,XOLD    PUFCOM10
     1  ,XJDIT(20)                                                      PUFCOM11
C              HALFSTEP VALUES                                          PUFCOM12
      COMMON DH,DHLAST,OUH,EH,PH,RH,RHLAST,SH,SHLAST,UH,UHLAST,XH,XHLASTPUFCOM13
     1  ,NEMH,NETH                                                      PUFCOM14
C              CONDITION INDICATORS                                     PUFCOM15
      COMMON INF,LINTER,MIRROR,NORMAL,POROUS,PRESS,RINTER,SOLID,SPALL   PUFCOM16
C              CELL LAYOUT                                              PUFCOM17
      COMMON OXX(30),JBNO(30),JMAT(30),NAUTD,MATL(6,2),NLAYER,NMTRLS,   PUFCOM18
     1  THK(30)                                                         PUFCOM19
C                                                                       PUFCOM20
C              COORDINATE ARRAYS                                        COORDC02
      COMMON/COORD/X(200),XO(200),CHL(200),DHL(200),OPDD(200),OPOE(200),COORDC03
     1  EHL(200),H(200,3),NEM(200),NET(200),PHL(200),RHL(200),SDT(200), COORDC04
     2  SHL(200),T(200),U(200),YHL(200),ZHL(200)                        COORDC05
C              NAMED COMMON                                             EQSTCOM2
      REAL MU,MUM                                                       EQSTCOM3
      COMMON /EQS/  EQSTA(6),EQSTC(6),EQSTD(6),EQSTE(6),EQSTG(6),       EQSTCOM4
     1  EQSTH(6),EQSTN(6),EQSTS(6),EQSTV(6),CZQ(6),CWQ(6),C2(6)         EQSTCOM5
      COMMON /MELT/ EMELT(6,5),SPH(6),THERM(6,8)                        EQSTCOM6
      COMMON /RHO/ RHO(6),RHOS(6)                                       EQSTCOM7
      COMMON /TSR/ TSR(6,30),EXMAT(6,20),TENS(6,3)                      EQSTCOM8
      COMMON /Y/ Y0(6),YADD(6),MU(6),MUM,YADDM                          EQSTCOM9
      COMMON /RAD/ SSTOP(5),START(5),SOURM,SSTOPM,NSPEC,SSJ,JSS,IPLOT(4)RADCOM 2
     1  ,XMAX(4),XMIN(4),YMAX(4),YMIN(4),IA(7),ITITLE(24),NARZ,TARZ     RADCOM 3
      COMMON/SS/SS(500)                                                 SSCOM  2
C                                                                       SCATTO21
      DIMENSION XS(1),ES(1)                                             SCATTO22
C                                                                       SCATTO23
C                                                                       SCATTO24
      FACTOR=4.186E7/(SSTOP(NS)-START(NS))                              SCATTO25
      ESS=ESUM=0.                                                       SCATTO26
      JFINNS=JFIN*(NS-1)                                                SCATTO27
      JS1=J=1                                                           SCATTO28
C              BEGIN LOOP FOR EACH MATERIAL                             SCATTO29
      IF (L .GT. 1) J=JBNO(L-1)+1                                       SCATTO30
      XENO=X(J)                                                         SCATTO31
      JBNOM=JBNO(L)                                                     SCATTO32
      XTH=X(JBNOM)                                                      SCATTO33
492   JS2=JS1+1  $  JS3=JS1+2                                           SCATTO34
      XSTOP=0.5*(XS(JS2)+XS(JS3))                                       SCATTO35
      IF (JS3 .EQ. NPOINT) XSTOP=XTH                                    SCATTO36
C        SET UP FOR INTEGRATION BY SIMPSONS RULE.  SEMILOG PARABOLA IN  SCATTO37
C                    LOG(E)=Z1*LOG(E1)+Z2*LOG(E2)+Z3*LOG(E3)            SCATTO38
C        WHERE Z(J)=-R(I,J)*R(J,K)    AND  R(I,J)=(X-X(K))/(X(J)-X(I))  SCATTO39
      ES1=ALOG(AMAX1(ES(JS1),1.E-10))                                   SCATTO40
```

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   XBEG=XEND
      XEND=AMIN1(XSTOP,X(J+1))
      R12=(XEND-X3)/DX12 $  R23=(XEND-X1)/DX23  $  R31=-(XEND-X2)/DX13
      ESS3=EXP(-ES1*R12*R31-ES2*R23*R12-ES3*R31*R23)
      XM=(XBEG+XEND)/2.
      R12=(XM  -X3)/DX12 $  R23=(XM  -X1)/DX23  $  R31=-(XM  -X2)/DX13
      ESS2=EXP(-ES1*R12*R31-ES2*R23*R12-ES3*R31*R23)
      ESS=(XEND-XBEG)/6.*(ESS1+4.*ESS2+ESS3) + ESS
      ESS1=ESS3
      IF (ABS(XEND-X(J+1)) .LT. 1.E-10) GO TO 496
C              PREPARE FOR NEW SET OF THREE XS COORDINATE POINTS
      IF (JS3 .LT. NPOINT) JS1=JS1+1
      GO TO 492
496   DX=X(J+1)-X(J)
      SS(JFINNS+J)=ESS*FACTOR/DX
      ESUM=ESUM+ESS
C              PREPARE FOR NEXT PUFF CELL
      IF (XEND.GT. XS(NPOINT) .AND. ES(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

```
      SUBROUTINE SCRIBE                                                 SCRIBE 2
      INTEGER H,POROUS,PRESS,RINTER,SOLID,SPALL                         PUFCOM 2
      REAL MATL,NEM,NET,NEMH,NETH                                       PUFCOM 3
C              MISCELLANEOUS                                            PUFCOM 4
      COMMON AZERO(1),CEF,CKS,DAVG,DELTIM,DISCPT(10),DOLD,DRHO,DTMAX,   PUFCOM 5
     1  DTMIN,DTN,DTNH,DU,DX,EOLD,F,FAC,FIRST,J,JCYCS,JINIT,            PUFCOM 6
     2  JFIN,JREZON(15),JSMAX,JSTAR,JTS,LSUB(30),M,MAXPR(30),N,NCYCS,   PUFCOM 7
     3  NEDIT,NPERN,NR,NREZON,NSCRB(6),NSEPRAT,NSPALL,NTEDT,            PUFCOM 8
     4  NTEX,NTR(15),POLD,P6(20),R(30),RLAST,SLAST,SMAX,TEDIT(50),      PUFCOM 9
     5  TF,TIME,TJ,TREZON,TS,T6(20),ULAST,UOLD,UZERO,XLAST,XNOW,XOLD    PUFCOM10
     1  ,XJDIT(20)                                                      PUFCOM11
C              HALFSTEP VALUES                                          PUFCOM12
      COMMON DH,DHLAST,DUH,EH,PH,RH,RHLAST,SH,SHLAST,UH,UHLAST,XH,XHLASTPUFCOM13
     1  ,NEMH,NETH                                                      PUFCOM14
C              CONDITION INDICATORS                                     PUFCOM15
      COMMON INF,LINTER,MIRROR,NORMAL,POROUS,PRESS,RINTER,SOLID,SPALL   PUFCOM16
C              CELL LAYOUT                                              PUFCOM17
      COMMON DXX(30),JBND(30),JMAT(30),NAUTO,MATL(6,2),NLAYER,NMTRLS,   PUFCOM18
     1  THK(30)                                                         PUFCOM19
C                                                                       PUFCOM20
C              COORDINATE ARRAYS                                        COORDC02
      COMMON/COORD/X(200),X0(200),CHL(200),DHL(200),DPOD(200),DPOE(200),COORDC03
     1  EHL(200),H(200,3),NEM(200),NET(200),PHL(200),RHL(200),SOT(200), COORDC04
     2  SHL(200),T(200),U(200),YHL(200),ZHL(200)                        COORDC05
      COMMON/NSC/A(5000)                                                NSCCOM 2
      COMMON /JED/JEDIT(100),JNUM(100),JTYP(100),NAME2(40),JEDSIZ,      JEDCOM 2
     1  MODULUS,NERR,NJEDIT,NTAPE                                       JEDCOM 3
      DIMENSION JV(13)                                                  SCRIBE 7
      NTAPE=3                                                           SCRIBE 8
      REWIND 7                                                          SCRIBE 9
      WRITE (7) N                                                       SCRIBE10
      CALL SECOND(XSTART)                                               SCRIBE11
      IF (NERR .GT. 0) PRINT 1083,NERR                                  SCRIBE12
      NSC=(NJEDIT+2)/10+1                                               SCRIBE13
      NBUF=(N-1)/MODULUS+1                                              SCRIBE14
      NPERP=50/MODLUS                                                   SCRIBE15
C                                                                       SCRIBE16
C        BEGIN DO LOOP OVER EACH SCRIBE LISTING                         SCRIBE17
C                                                                       SCRIBE18
      DO 900 NS=1,NSC                                                   SCRIBE19
      LENGTH=MODULUS                                                    SCRIBE20
      IPAG=1                                                            SCRIBE21
      IF (UNIT(NTAPE)) 650,990,640                                      SCRIBE22
640   PRINT 1082                                                        SCRIBE23
      REWIND NTAPE                                                      SCRIBE24
C                                                                       SCRIBE25
C        BUFFER IN FIRST RECORD OF TAPE                                 SCRIBE26
650   BUFFER IN (NTAPE,1) (A1,A1)                                       SCRIBE27
      IF (UNIT(NTAPE)) 655,990,652                                      SCRIBE28
652   PRINT 1082                                                        SCRIBE29
655   BUFFER IN (NTAPE,1) (A(1),A(JEDSIZ*MODULUS))                      SCRIBE30
      IF (NS .GT. 1) GO TO 680                                          SCRIBE31
      JEND=MIN0(12,NJEDIT+5)                                            SCRIBE32
      JBEG=3                                                            SCRIBE33
      JV(1)=10H(0PF6.0,                                                 SCRIBE34
      JV(2)=10HF10.3                                                    SCRIBE35
      JV(3)=10H,F10.3                                                   SCRIBE36
      JV(4)=10H,F10.3                                                   SCRIBE37
      JV(5)=10H,F6.0                                                    SCRIBE38
      JV(13)=10H)                                                       SCRIBE39
      DO 670 I=1,7                                                      SCRIBE40
      JJ=I+5                                                            SCRIBE41
      JV(JJ)=10H,1PE11.3                                                SCRIBE42
      IF (JNUM(I) .GE. 1400 .AND. JNUM(I) .LT. 2000) JV(JJ)=9H,10X,A1   SCRIBE43
      IF (I .GT. NJEDIT) JV(JJ)=1H                                      SCRIBE44
670   CONTINUE                                                          SCRIBE45
      GO TO 695                                                         SCRIBE46
680   JBEG=JEND+1                                                       SCRIBE47
```

SUBROUTINE SCRIBE (Concluded)

```
      JEND=MIN0(JBEG+9,NJEDIT+5)
      JD=JBEG+9
      DO 690 I=JBEG,JD
      JJ=I-JBEG+3
      JE=I-5
      JV(JJ)=I0H,IPE11.3
      IF (JNUM(JE) .GE. I400 .AND. JNUM(JE) .LT. 2000)
     I  JV(JJ)=9H,I0X,A1
      IF (I .GT. NJEDIT+5) JV(JJ)=1H
690   CONTINUE
695   DO 850 NB=1,NBUF
      IF (UNIT(NTAPE)) 710,990,700
700   PRINT 1082
710   IF (NB .NE. 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,1)(A(JBUF+1),A(JBUF+2500))
770   CONTINUE
      JI=MOD(NB-1,2)*2500
      J2=(LENGTH-I)*JEDSIZ+JI
      JB=JBEG-5
      JD=JEND-5
      IF (IPAG .EQ. I .AND. NS .GT. 1 ) PRINT 1200,DISCPT,NS,(JTYP(I),
     1  I=JB,JD)
      IF (IPAG .EQ. I .AND. NS .EQ. 1) PRINT 1100,DISCPT,NS,(JTYP(I),
     1  I=1,JD)
      PRINT JV,((A(I+J),I=1,2),(A(I+J),I=JBEG,JEND),J=J1,J2,JEDSIZ)
      IPAG=MOD(IPAG,NPERP)+I
850   CONTINUE
      REWIND NTAPE
900   CONTINUE
      CALL SECOND(XEND)
      DUR=XEND-XSTART
      DUR2=XEND-FIRST
      PRINT 1900,DUR,DUR2
      RETURN
990   PRINT I084
      RETURN
1082  FORMAT (32H PARITY ERROR ON NTAPE IN SCRIBE)
1083  FORMAT (* EOFS AND PARITY ERRORS ON TAPE 3, NERR =*I3)
1084  FORMAT (29H EOF FOUND ON NTAPE IN SCRIBE)
II00  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*,6X,*DTNH*,4X,*DELTIM*,3X,*JTS*,7(1X,AI0))
1200  FORMAT (1H1,10A10/* SCRIBE NO. *I2,* USUAL UNITS ARE DYN, CM, SEC,
     2 GRAM*/   5X,*N*,6X,*TIME*,10(1X,A10))
1900  FORMAT (17H0TIME IN SCRIBE = F10.3/I7H COMPUTING TIME = F10.3)
      END
```

SUBROUTINE SHEAR2

```
      SUBROUTINE SHEAR2(NCALL,IN,M,K,J,IH3,SX,SY,SZ,SXY,P,TAU,DH,DOLD,  SHEAR2 2
     1DTO,EH,EOLD,EN,EMELT,EP,EX,EY,EZ,EXY,F,YHL,PLEN,ROT,OROT,ESC,CN)  SHEAR2 3
C                                                                       SHEAR2 4
C           ROUTINE FOR COMPUTATION OF STRESSES WITH RATE-DEPENDENT     SHEAR2 5
C        YIELD MODEL FOR DEVIATORS AND MIE-GRUNEISEN FOR PRESSURE.      SHEAR2 6
C        IF THRESHOLD PLASTIC STRAIN IS REACHED, SHEAR BANDS ARE        SHEAR2 7
C        NUCLEATED AND GROWN IN 6 ORIENTATIONS.                         SHEAR2 8
C           SX, SY, SZ ARE DEVIATORS IN EXTERNAL SIGN CONVENTION.       SHEAR2 9
C        P IS POSITIVE IN COMPRESSION.  INTERNAL SIGN CONVENTION IS     SHEAR210
C        POSITIVE IN COMPRESSION FOR ALL STRESS AND STRAIN QUANTITIES   SHEAR211
C        ST IS TOTAL STRESS AT PREVIOUS TIME.  SE IS NEW DEVIATOR.      SHEAR212
C        EX, EY, EZ, EXY ARE STRAIN INCREMENTS IN EXTERNAL SIGN CON.    SHEAR213
C        SS, SSE CHANGE EXTERNAL SIGN CONVENTION TO INTERNAL FOR        SHEAR214
C        STRESS AND STRAIN, RESPECTIVELY.                               SHEAR215
C                                                                       SHEAR216
      DIMENSION BFR(6,35),NSIZE(30,9),FNUC(9),TAUZ(6),EFR(3),VFR(6),    SHEAR217
     1ST(4),ES(4),SE(4),TEP(6),ESC(6,20),NSIZT(6)                       SHEAR218
     2  ,CN(100),DEP(4),CLA(100),CNA(100),VMAX(6)                       SHEAR219
      EQUIVALENCE (CNA,CLA)                                             SHEAR220
C     STRESS IS NEG IN TENSION                                          SHEAR221
      DATA SS,SSE/-1.,-1./                                              SHEAR222
      NC1=NCALL+1                                                       SHEAR223
      GO TO (10,10,100,100,900)NC1                                      SHEAR224
 10   READ(IN,1002) A1,A2,(BFR(M,I),I=22,35)                            SHEAR225
      PRINT 1002, A1,A2,(BFR(M,I),I=22,35)                              SHEAR226
 1002 FORMAT(2A5,7E10.3/10X,7E10.3)                                     SHEAR227
      REAO(IN,1003)A1,A2,(NSIZE(M,I),I=1,9)                             SHEAR228
      PRINT 1003,A1,A2,(NSIZE(M,I),I=1,9)                               SHEAR229
 1003 FORMAT(2A5,1415)                                                  SHEAR230
      VMAX(M)=0.                                                        SHEAR231
      NSIZT(M)=NSIZE(M,1)                                               SHEAR232
      DO 14 I=2,9                                                       SHEAR233
  14  NSIZT(M)=NSIZT(M)+NSIZE(M,I)                                      SHEAR234
      VFR(M)=1.                                                         SHEAR235
      IF (NCALL .EQ. 1)  GO TO 65                                       SHEAR236
      NANG=BFR(M,32)                                                    SHEAR237
      KLAST=0                                                           SHEAR238
      DO 16 I=1,3                                                       SHEAR239
      FNUC(I)=.111111                                                   SHEAR240
 16   FNUC(I+3)=.222222                                                 SHEAR241
      IF(NANG - 6) 20,40,30                                             SHEAR242
 20   FNUCI=.333333                                                     SHEAR243
      IF (NANG .GE.4) FNUCI =.25                                        SHEAR244
      DO 25 I=1,4                                                       SHEAR245
 25   FNUC(I)=FNUCI                                                     SHEAR246
      IF (NANG .EQ.2) FNUC(2) =.6666667                                 SHEAR247
      IF (NANG .NE. 5)  GO TO 30                                        SHEAR248
      FNUC(4)=.125                                                      SHEAR249
      FNUC(7)=.125                                                      SHEAR250
      FNUC(5)=0.                                                        SHEAR251
      FNUC(6)=0.                                                        SHEAR252
      GO TO 40                                                          SHEAR253
 30   DO 35 I=7,NANG                                                    SHEAR254
      FNUC(I)=.111111                                                   SHEAR255
 35   FNUC(I-3)=.111111                                                 SHEAR256
 40   CONTINUE                                                          SHEAR257
 65   RETURN                                                            SHEAR258
C*********                                                              SHEAR259
C                                                                       SHEAR260
C          COMPUTE STRESS AND DAMAGE                                    SHEAR261
C                                                                       SHEAR262
C*********                                                              SHEAR263
 100  IF (IH3 .GE. 25) GO TO 800                                        SHEAR264
      IF (VMAX(M) .EQ. 0.) VMAX(M)=SQRT(ESC(M,5)/ESC(M,1))              SHEAR265
C      COMPUTE STRESS REDUCTION FACTORS TAUZ(I)                         SHEAR266
      TAU=0.                                                            SHEAR267
      JN=0                                                              SHEAR268
      DO 110 NG=1,NANG                                                  SHEAR269
```

```
      TAUZ(NG)=0.
      IF(NSIZE(M,NG) .EQ. 0 .OR. CN(JN+1) .EQ. 0.)GO TO 110
      NSIZEM=NSIZE(M,NG)
      DO 120 I=1,NSIZEM
      JNN=JN+2*I-1
  120 TAUZ(NG)=TAUZ(NG)+CN(JNN)*CN(JNN+1)**3
      TAU=TAU+TAUZ(NG)
  110 JN=JN+2*NSIZE(M,NG)
C*********
C         INITIAL TRANSFORMATION
C*********
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)*EMU/2.)+DOLD*ESC(M,9)*EOLD
      IF(PS .GT. 0.) P=P-PS*TAU
      SA=(SX+SY)*SS/2.+P
      SOR=SIN(RT)
      COR=COS(RT)
      SB=((SX-SY)/2.*COR+SXY*SOR)*SS
      DO 140  I=1,4
  140 ST(I)=0.
      G2=2.*ESC(M,5)
      ST(1)= (SA+SB)/AMAX1(0.02,(1.-(3.*TAUZ(1)+1.5*(TAUZ(4)+TAUZ(5)))
     1 *VFR(M)))
      ST(2)=(SA-SB)/AMAX1(0.02,(1.-(3.*TAUZ(2)+1.5*(TAUZ(4)+TAUZ(6)))
     1 *VFR(M)))
      ST(3)=(P-(SX+SY)*SS)/AMAX1(0.02,(1.-(3.*TAUZ(3)+1.5*(TAUZ(5)+
     1 TAUZ(6)))*VFR(M)))
      ST(4)=((SY-SX)/2.*SOR+SXY*COR)*SS/(1.-(1.5*(TAUZ(1)+TAUZ(2))+3.
     1 *TAUZ(4))*VFR(M))
      IF(J.EQ. 17 .AND. TAU .GT. 0.05) PRINT 1400,SX,SY,SZ,P,ST,EX,EY,DH
 1400 FORMAT(10H SX,SY,SZ=3E10.3,3H P=E10.3,4H ST=4E10.3/7H EX,EY=2E10.3
     1,4H DH=F10.5)
      P=(ST(1)+ST(2)+ST(3))/3.
C          ROTATE STRAINS TO BAND ORIENTATIONS
      EA=(EX+EY)/2.
      EB=(EX-EY)/2.*COR+EXY*SOR
      EBAR=0.6667*(DH-DOLD)/(DH+DOLD)
      NSTEP=SQRT((ABS(EA)+ABS(EB)+ABS(EBAR))/.002)
      NSTEP=MAX0(NSTEP,1)
      ES(1)= (EA+EB)*SSE/NSTEP
      ES(2)= (EA-EB)*SSE/NSTEP
      ES(3)=3.*EBAR/NSTEP-ES(1)-ES(2)
      ES(4)=(((EY-EX)/2.*SOR+EXY*COR)*SSE)/NSTEP
      DO 600 NS=1,NSTEP
      DO 160  I=1,3
  160 SE(I)=ST(I)+G2*(ES(I)-EBAR/NSTEP)-P
      SE(4)= ST(4)+G2*ES(4)
      SN=SQRT(1.5*(SE(1)**2+SE(2)**2+SE(3)**2+2.*SE(4)**2))
      DHN=DOLD+FLOAT(NS)/FLOAT(NSTEP)*(DH-DOLD)
      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.)+DHN*ESC(M, 9)*EH
      Y1=AMAX1(0.,YHL+BFR(M,33)*PE)
      IF (SN .LT. Y1)  GO TO 500
C**********
C          YIELD AND PLASTIC STRAIN CALCULATIONS
C**********
      EXPT = EXP(-DTO/BFR(M,30)/NSTEP)
      YEG=(Y1+BFR(M,31)*SN/2./G2)/(1.+BFR(M,31)/2./G2)/SN
      DO 180  I=1,3
  180 SE(I)=(ST(I)-P)*EXPT+(YEG*SE(I)+BFR(M,30)*(SE(I)-ST(I)+P)/
     1DTO*NSTEP)*(1.-EXPT)
      SE(4)=ST(4)*EXPT+(YEG*SE(4)+BFR(M,30)*(SE(4)-ST(4))/DTO*NSTEP)*
     1 (1.-EXPT)
      DO 200  I=1,3
```

```
  200 DEP(I)=ES(I)-EBAR/NSTEP-(SE(I)-ST(I)+P)/G2
      DEP(4)=ES(4)-(SE(4)-ST(4))/G2
      DGAMMA=SQRT(1.5*(DEP(1)**2+DEP(2)**2+DEP(3)**2)+0.75*DEP(4)
     1**2)
      YHL=YHL+BFR(M,31)*DGAMMA
      DPLENR=((SE(1)+ST(1)-P)*DEP(1)+(SE(2)+ST(2)-P)*DEP(2)+(SE(3)+S
     1T(3)-P)*DEP(3)+(SE(4)+ST(4))*DEP(4))/2./DHN*AMAX1(0.,1.-TAU)
      DPLENR=ABS(DPLENR)
      EP=EP+DGAMMA
      PLEN=PLEN+DPLENR
C**********
C          COMPUTE PLASTIC STRAIN IN EACH ORIENTATION
C**********
      STR1=ABS(SE(4))
      STR2=ABS(SE(4))
      STR3=0.
      STR4=ABS(SE(1)-SE(2))/2.
      STR5=SQRT((SE(1)-SE(3))**2+2.*SE(4)**2)/2.
      STR6=SQRT((SE(2)-SE(3))**2+2.*SE(4)**2)/2.
      SN=SQRT(1.5*(SE(1)**2+SE(2)**2+SE(3)**2+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**********
C         GROWTH PROCESS
C**********
      NTOT=2*NSIZT(M)
      DO 250  I=1,NTOT
  250 CNA(I)=CN(I)
      IF (EN .EQ. 0) GO TO 360
      JN=0
      DC=VMAX(M)*DTO/NSTEP
      DO 350  NG=1,NANG
      DGAM=0.
      IF (NSIZE(M,NG) .EQ. 0. .OR. CN(JN+1) .EQ. 0 .OR. TEP(NG) .LE. 0.)
     1 GO TO 345
      EXPE=EXP(BFR(M,22)*TEP(NG))
      NSIZEM=NSIZE(M,NG)
      DO 300 I=1,NSIZEM
      JN2=JN+2*(NSIZE(M,NG)+1-I)
      CLA(JN2)=AMIN1(CN(JN2)*EXPE,CN(JN2)+DC)
  300 DGAM=DGAM+CN(JN2-1)*3.14*BFR(M,27)*(CLA(JN2)**3-CN(JN2)**3)
      IF(DGAM .LE. TEP(NG))GO TO 345
      RR=TEP(NG)/DGAM
      DCR=DC*RR
      EXPE=EXPE**RR
      NSIZEM=NSIZE(M,NG)
      DO 340 I=1,NSIZEM
      JN2=JN+2*(NSIZE(M,NG)+1-I)
  340 CLA(JN2)=AMIN1(CN(JN2)*EXPE,CN(JN2)+DCR)
  345 TEP(NG)=AMAX1(0.,TEP(NG)-DGAM)
  350 JN=JN+NSIZE(M,NG)*2
  360 CONTINUE
      DO 365 NG=1,NANG
365   CN(NTOT+NG)=CN(NTOT+NG)+TEP(NG)
C**********
C          NUCLEATION PROCESS
C**********
      TEPM=0
      DO 370 NG=1,NANG
370   TEPM=AMAX1(TEPM,CN(NTOT+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
```

```
      IF (CN(NTOT+NG) .LT. BFR(M,26) .OR. TEP(NG) .LT. 1.E-5)           SHEAR206
     1 GO TO 450                                                        SHEAR207
      DNO=TEP(NG)*BFR(M,25)*FNUC(NG)*(DPLENR/DTO*NSTEP/BFR(M,35))**2    SHEAR208
      CNR=0.                                                            SHEAR209
      NSIZEM=NSIZE(M,NG)                                                SHEAR210
      DO 440 I=1,NSIZEM                                                 SHEAR211
      II=NSIZE(M,NG)+1-I                                                SHEAR212
      JNI=JN+2*II                                                       SHEAR213
      IF(CLA(JNI) .NE. 0.) GO TO 420                                    SHEAR214
      CLA(JNI)=BFR(M,28)*(1.-BFR(M,29)**II)/(1.-BFR(M,29)**NSIZE(M,NG)) SHEAR215
      CN(JNI)=CLA(JNI)                                                  SHEAR216
 420  CNL=DNO*EXP(-(CLA(JNI)+CN(JNI))/2./BFR(M,24))                     SHEAR217
      JNN=JN+2*II-1                                                     SHEAR218
      CNA(JNN)=CNL-CNR+CN(JNN)                                          SHEAR219
  440 CNR=CNL                                                           SHEAR220
      EN=EN+CNL                                                         SHEAR221
  450 JN=NSIZE(M,NG)*2+JN                                               SHEAR222
 470   CONTINUE                                                         SHEAR223
C**********                                                             SHEAR224
C         COMPUTE TAU AND REFILL MAIN ARRAYS                            SHEAR225
C**********                                                             SHEAR226
      TAU=0.                                                            SHEAR227
      JN=0                                                              SHEAR228
      IF(EN .EQ. 0.)  GO TO 500                                         SHEAR229
      DO 490   NG=1,NANG                                                SHEAR230
      TAUZ(NG)=0.                                                       SHEAR231
      IF (NSIZE(M,NG) .EQ. 0) GO TO 490                                 SHEAR232
      IF (CNA(JN+1) .EQ. 0.) GO TO 490                                  SHEAR233
      NSIZEM=NSIZE(M,NG)                                                SHEAR234
      DO 480 I=1,NSIZEM                                                 SHEAR235
      JNN=JN+2*I-1                                                      SHEAR236
      CN(JNN+1)=CLA(JNN+1)                                              SHEAR237
      CN(JNN)=CNA(JNN)                                                  SHEAR238
 480  TAUZ(NG)=TAUZ(NG)+CNA(JNN)*CLA(JNN+1)**3                          SHEAR239
      TAU=TAU+TAUZ(NG)                                                  SHEAR240
 490  JN=JN+NSIZE(M,NG)*2                                               SHEAR241
      IF(TAU*VFR(M) .GE. 1.) GO TO 800                                  SHEAR242
 500  CONTINUE                                                          SHEAR243
      P=PE                                                              SHEAR244
      DO 550 I=1,3                                                      SHEAR245
 550  ST(I)=SE(I)+P                                                     SHEAR246
      ST(4)=SE(4)                                                       SHEAR247
 600  CONTINUE                                                          SHEAR248
C**********                                                             SHEAR249
C     TRANSFORMATION TO GLOBAL ORIENTATION                              SHEAR250
C**********                                                             SHEAR251
      IF(J .EQ. 17 .AND. TAU .GT.  .05) PRINT 1601,ST,P                 SHEAR252
 1601 FORMAT(4H ST= 1P4E10.3,4H PS= E10.3)                              SHEAR253
      ST(4)=ST(4)*AMAX1(0.,(1.-(1.5*(TAUZ(1)+TAUZ(2))+3.*TAUZ(4))*VFR(M)SHEAR254
     1  ))                                                              SHEAR255
      ST(1)=ST(1)*AMAX1(0.,(1.-(3.*TAUZ(1)+1.5*(TAUZ(4)+TAUZ(5)))*VFR(M)SHEAR256
     1  ))                                                              SHEAR257
      ST(2)=ST(2)*AMAX1(0.,(1.-(3.*TAUZ(2)+1.5*(TAUZ(4)+TAUZ(6)))*VFR(M)SHEAR258
     1  ))                                                              SHEAR259
      ST(3)=ST(3)*AMAX1(0.,(1.-(3.*TAUZ(3)+1.5*(TAUZ(5)+TAUZ(6)))*VFR(M)SHEAR260
     1  ))                                                              SHEAR261
      P=(ST(1)+ST(2)+ST(3))/3.                                          SHEAR262
      SA=(ST(1)+ST(2))/2.                                               SHEAR263
      SB=(ST(1)-ST(2))/2.*COR-ST(4)*SOR                                 SHEAR264
      SXY=((ST(1)-ST(2))/2.*SOR+ST(4)*COR)*SS                           SHEAR265
      SX=(SA+SB-P)*SS                                                   SHEAR266
      SY=(SA-SB-P)*SS                                                   SHEAR267
      SZ=(ST(3)-P)*SS                                                   SHEAR268
      IF (PE .GT. 0.) P=P+PE*TAU                                        SHEAR269
      IH3=20.*TAU+2.9                                                   SHEAR270
      IF(J .EQ. 17 .AND. TAU .GT. .05) PRINT 1400,SX,SY,SZ,P,ST,EX,EY,DHSHEAR271
      IF (NCALL .GE. 3) GO TO 900                                       SHEAR272
      RETURN                                                            SHEAR273
```

```
C**********
C     COMPLETE SEPARATION
C**********
800   EMU=DH/ESC(M,1)-1.
      PH=EMU*(ESC(M,2)+EMU*(ESC(M, 3)+EMU*ESC(M, 4)))
      PE=PH*(1.-ESC(M, 9)*(DH/ESC(M,1)-1.)/2.)+DH*ESC(M, 9)*EH
      P=AMAX1(PE,0.)
      SX=0.
      SY=0.
      SZ=0.
      SXY=0.
      IH3=25
      IF (NCALL .GE. 3) GO TO 900
      RETURN
C * * * * *
C
C     PRINT DAMAGE ARRAYS
C
C * * * * *
900   IF (EN .EQ. 0.) GO TO 980
      IF (K .LT. KLAST) REWIND 2
      KLAST=K
      PRINT 8000,K,J,IH3,ROT,EN,TAU,EP
      JN=0
      DO 1000 NG=1,NANG
      TAUZ(NG)=0.
      NS=NSIZE(M,NG)*2
      IF (NS .EQ. 0 .OR. CN(JN+1) .EQ. 0.)GO TO 1000
       II=JN+NS-1
       CNA(II)=CN(II)
      TAUZ(NG)=CN(II)*CN(II+1)**3
       IF (NS .LE. 2) GO TO 975
       DO 970 I=3,NS,2
       II=NS-I+JN
      TAUZ(NG)=TAUZ(NG)+CN(II)*CN(II+1)**3
 970   CNA(II)=CN(II)+CNA(II+2)
 975   CONTINUE
      PRINT 8500,NG
      PRINT 9001,(CNA(JN+I),I=1,NS,2)
      PRINT 9002,(CN(JN+I),I=2,NS,2)
      WRITE (2,1902) J,K,NG
      WRITE (2,9001) (CN(JN+I),I=1,NS,2)
      WRITE (2,9002) (CN(JN+I),I=2,NS,2)
 1902 FORMAT(10H J =      ,I10,10H K=       ,I10,10H NG=      ,I10)
 8000 FORMAT(3H K=I5,3H J=I5,5H IH3=I5.5H ROT=E10.3,4H EN=E10.3,
     1  5H TAU=E10.3,4H EP=E10.3)
 8500 FORMAT(4H NG=I5)
 9001 FORMAT(4H CN=10(E10.3,2X))
 9002  FORMAT(4H CL=10(E10.3,2X))
 1000 JN=JN+NS
      PRINT 1980,(TAUZ(I),I=1,NANG)
 1980 FORMAT (6H TAUZ= 9F10.6)
      NTOT=2*NSIZT(M)
980   PRINT 9003,K,J,EP,(CN(NTOT+I),I=1,NANG)
9003  FORMAT(* K=*I3,* J=*I3,* EP=*  E12.4,16H TOT PL STRAIN =6F10.3)
      RETURN
      END
```

SUBROUTINE SIGMAT

```
      FUNCTION SIGMAT(LS,T)                                             SIGMAT 2
      DIMENSION PS(10),TS(10)                                           SIGMAT 3
      DATA PS/54.4E6,34.0E6,40.8E6,24.5E6,6*0./                         SIGMAT 4
      DATA TS/0.,6.E-4,8.E-4,3.2E-3,6*0./                               SIGMAT 5
      DATA NM/4/                                                        SIGMAT 6
      N=1                                                               SIGMAT 7
   20 N=N+1                                                             SIGMAT 8
      IF ( T .GT. TS(N) .AND. N .LT. NM) GO TO 20                       SIGMAT 9
      SIGMAT=PS(N-1)+(PS(N)-PS(N-1))/(TS(N)-TS(N-1))*(T-TS(N-1))        SIGMAT10
      RETURN                                                            SIGMAT11
      END                                                               SIGMAT12
```

SUBROUTINE SSCALH

```
      FUNCTIDN SSCALH(JS)                                               SSCALH 2
C                                                                       SSCALH 3
C     COMPUTES RADIANT ENERGY FOR DEPOSITION IN EACH CELL AT HALFSTEP   SSCALH 4
C     PDINT AND INITIALIZES ENERGY IN NEW ZONES                         SSCALH 5
C     INPUT - J(=JS), NSPEC, SDURM, TIME, DTNH, SSTOPM, DTN.            SSCALH 6
C     DUTPUT - SSCALH.                                                  SSCALH 7
C                                                                       SSCALH 8
      INTEGER H,POROUS,PRESS,RINTER,SOLIO,SPALL                         PUFCOM 2
      REAL MATL,NEM,NET,NEMH,NETH                                       PUFCOM 3
C             MISCELLANEOUS                                             PUFCOM 4
      COMMON AZERO(1),CEF,CKS,DAVG,DELTIM,OISCPT(10),OOLD,ORHO,OTMAX,   PUFCOM 5
     1 DTMIN,DTN,DTNH,DU,DX,EDLD,F,FAC,FIRST,J,JCYCS,JINIT,             PUFCOM 6
     2 JFIN,JREZON(15),JSMAX,JSTAR,JTS,LSUB(30),M,MAXPR(30),N,NCYCS,    PUFCOM 7
     3 NEDIT,NPERN,NR,NREZON,NSCRB(6),NSEPRAT,NSPALL,NTEDT,             PUFCOM 8
     4 NTEX,NTR(15),POLD,P6(20),R(30),RLAST,SLAST,SMAX,TEOIT(50),       PUFCOM 9
     5 TF,TIME,TJ,TREZON,TS,T6(20),ULAST,UOLO,UZERD,XLAST,XNOW,XOLO     PUFCOM10
     1 ,XJDIT(20)                                                       PUFCOM11
C             HALFSTEP VALUES                                           PUFCOM12
      COMMDN OH,OHLAST,OUH,EH,PH,RH,RHLAST,SH,SHLAST,UH,UHLAST,XH,XHLASTPUFCOM13
     1 ,NEMH,NETH                                                       PUFCOM14
C             CONDITION INDICATORS                                      PUFCOM15
      COMMDN INF,LINTER,MIRRDR,NDRMAL,PDROUS,PRESS,RINTER,SOLID,SPALL   PUFCOM16
C             CELL LAYOUT                                               PUFCOM17
      COMMON DXX(30),JBNO(30),JMAT(30),NAUTO,MATL(6,2),NLAYER,NMTRLS,   PUFCOM18
     1 THK(30)                                                          PUFCOM19
C                                                                       PUFCOM20
      COMMON /RAD/ SSTOP(5),START(5),SDURM,SSTOPM,NSPEC,SSJ,JSS,IPLOT(4)RADCOM 2
     1 ,XMAX(4),XMIN(4),YMAX(4),YMIN(4),IA(7),ITITLE(24),NARZ,TARZ      RADCOM 3
      COMMON/SS/SS(500)                                                 SSCOM  2
C                                                                       SSCALH12
10    SSCALH=0.                                                         SSCALH13
      IF (NSPEC .EQ. 0) RETURN                                          SSCALH14
      IF (JS .GT. JSS) GD TO 50                                         SSCALH15
      IF (SOURM .EQ. 1.) RETURN                                         SSCALH16
      IF (TIME-DTNH-.5*DTN .GT. SSTDPM) RETURN                          SSCALH17
C             ENERGY ADDITION IN ACTIVE ZONES - HALF STEP               SSCALH18
      DO 48 I=1,NSPEC                                                   SSCALH19
      JFINNS=JFIN*(I-1)                                                 SSCALH20
      IF ((TIME-.5*OTNH-START(I))*(TIME-DTNH-.5*OTN-SSTDP(I))) 46,48,48 SSCALH21
 46   SSCALH=SSCALH+SS(JFINNS+JS)*(AMIN1(SSTOP(I),TIME-.5*DTNH)-        SSCALH22
     1 AMAX1(START(I),TIME-DTNH-.5*OTN))                                SSCALH23
48    CONTINUE                                                          SSCALH24
      RETURN                                                            SSCALH25
C             ENERGY ADDITION FDR NEW ZONES                             SSCALH26
50    JSS=JS                                                            SSCALH27
      DO 60 I=1,NSPEC                                                   SSCALH28
      JFINNS=JFIN*(I-1)                                                 SSCALH29
      IF (TIME-.5*DTNH .LT. START(I)) GO TO 60                          SSCALH30
      SSCALH=SSCALH+SS(JFINNS+JS)*(AMIN1(SSTDP(I),TIME-.5*DTNH)-        SSCALH31
     1 START(I))                                                        SSCALH32
60    CONTINUE                                                          SSCALH33
75    RETURN                                                            SSCALH34
      END                                                               SSCALH35
```

SUBROUTINE STORR

```
      SUBROUTINE STORR                                                  STORR      2
      INTEGER H,POROUS,PRESS,RINTER,SOLID,SPALL                         PUFCOM     2
      REAL MATL,NEM,NET,NEMH,NETH                                       PUFCOM     3
C              MISCELLANEOUS                                            PUFCOM     4
      COMMON AZERO(1),CEF,CKS,DAVG,DELTIM,DISCPT(10),DOLD,DRHO,DTMAX,   PUFCOM     5
     1 DTMIN,DTN,DTNH,DU,DX,EOLD,F,FAC,FIRST,J,JCYCS,JINIT,             PUFCOM     6
     2 JFIN,JREZON(15),JSMAX,JSTAR,JTS,LSUB(30),M,MAXPR(30),N,NCYCS,    PUFCOM     7
     3 NEDIT,NPERN,NR,NREZON,NSCRB(6),NSEPRAT,NSPALL,NTEDT,             PUFCOM     8
     4 NTEX,NTR(15),POLD,P6(20),R(30),RLAST,SLAST,SMAX,TEDIT(50),       PUFCOM     9
     5 TF,TIME,TJ,TREZON,TS,T6(20),ULAST,UOLD,UZERO,XLAST,XNOW,XOLD     PUFCOM    10
     1 ,XJDIT(20),MS                                                    PUFCOM    11
C              HALFSTEP VALUES                                          PUFCOM    12
      COMMON DH,DHLAST,DUH,EH,PH,RH,RHLAST,SH,SHLAST,UH,UHLAST,XH,XHLAST PUFCOM    13
     1 ,NEMH,NETH                                                       PUFCOM    14
C              CONDITION INDICATORS                                     PUFCOM    15
      COMMON INF,LINTER,MIRROR,NORMAL,POROUS,PRESS,RINTER,SOLID,SPALL   PUFCOM    16
C              CELL LAYOUT                                              PUFCOM    17
      COMMON DXX(30),JBND(30),JMAT(30),NAUTO,MATL(6,2),NLAYER,NMTRLS,   PUFCOM    18
     1 THK(30)                                                          PUFCOM    19
C                                                                       PUFCOM    20
C              COORDINATE ARRAYS                                        COORDCOM   2
      COMMON/COORD/X(200),X0(200),CHL(200),DHL(200),DPDD(200),DPDE(200),COORDCOM   3
     1 EHL(200),H(200,3),NEM(200),NET(200),PHL(200),RHL(200),SDT(200),  COORDCOM   4
     2 SHL(200),T(200),U(200),YHL(200),ZHL(200)                         COORDCOM   5
      COMMON/NSC/A(5000)                                                NSCCOM     2
      COMMON /IND/ IEOS(6),INDK(20),NALPHA,NCMP(6),NFR(6),NPOR(6),      INDCOM     2
     1 NDS(6),NPR(6),NCON(6),NVAR(6)                                    INDCOM     3
      COMMON /JED/JEDIT(100),JNUM(100),JTYP(100),NAME2(40),JEDSIZ,      JEDCOM     2
     1 MODLUS,NERR,NJEDIT,NTAPE                                         JEDCOM     3
      COMMON /PES/ LVMAX,LVTOT,LVAR(200),COM(4000)                      STORR      8
      DIMENSION RIMP(20),JINT(20)                                       STORR      9
      DIMENSION KB(300)                                                 STORR     10
      EQUIVALENCE (A(2501),KB)                                          STORR     11
      IF (N .GT. 1) GO TO 100                                           STORR     12
C        INITIALIZATION                                                 STORR     13
      NTAPE=3  $  NERR=IBUF=0  $  MODLUS=50                             STORR     14
      IF (NJEDIT .GT. 45) MODLUS=25                                     STORR     15
      IF (NJEDIT .GT. 95) MODLUS=10                                     STORR     16
      IF (NJEDIT .GT. 245) MODLUS=5                                     STORR     17
      JEDSIZ=2500/MODLUS                                                STORR     18
      DO 30 I=1,10                                                      STORR     19
30    A(2500+I)=DISCPT(I)                                               STORR     20
      KB(11)=MODLUS                                                     STORR     21
      KB(12)=JEDSIZ                                                     STORR     22
      KB(13)=JCYCS                                                      STORR     23
      KB(14)=NJEDIT                                                     STORR     24
      DO 40 I=1,100                                                     STORR     25
      KB(14+I)=JTYP(I)                                                  STORR     26
40    KB(114+I)=JNUM(I)                                                 STORR     27
      BUFFER OUT(NTAPE,1) (A(2501),A(2714))                             STORR     28
      DO 50 I=1,20                                                      STORR     29
      JINT(I)=JEDIT(I)                                                  STORR     30
 50   RIMP(I)=0.                                                        STORR     31
C                                                                       STORR     32
C        BEGIN STORAGE                                                  STORR     33
 100  IF (LSUB(7) .NE. 0) GO TO 600                                     STORR     34
      IB=JEDSIZ*IBUF                                                    STORR     35
      A(IB+1)=N                                                         STORR     36
      A(IB+2)=TIME*1.E6                                                 STORR     37
      A(IB+3)=DTNH*1.E9                                                 STORR     38
      A(IB+4)=DELTIM                                                    STORR     39
      A(IB+5)=JTS                                                       STORR     40
      IC=IB+5                                                           STORR     41
      IR=0                                                              STORR     42
      DO 500 JE=1,NJEDIT                                                STORR     43
      JD=JEDIT(JE)                                                      STORR     44
      JNUMB=JNUM(JE)                                                    STORR     45
      IF (JNUMB .GE. 4000 ) GOTO 200                                    STORR     46
C        STORAGE FOR ALL ARRAY VARIABLES                                STORR     47
      A(IC+JE)=X(JNUMB+JD)                                              STORR     48
      GO TO 500                                                         STORR     49
C        STORAGE FOR COM  VARIABLES                                     STORR     50
 200  IF (JNUMB .GE. 5000) GO TO 300                                    STORR     51
      JN=JNUMB-4000                                                     STORR     52
      L=LVAR(JD)                                                        STORR     53
```

SUBROUTINE STORR (Concluded)

```
      A(IC+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(IC+JE)=R(JD+1)
      GO TO 500
C        SECOND PRINCIPAL STRESS
  320 IF (NALPHA .EQ. 2) GO TO 325
      A(IC+JE)=-0.5*SHL(JD)+1.5*PHL(JD)
      GO TO 500
  325 A(IC+JE)=PHL(JD)+SDT(JD)
      GO TO 500
C      THIRD PRINCIPAL STRESS
  330 IF (NALPHA .EQ.2) GO TO 335
      A(IC+JE)=-0.5*SHL(JD)+1.5*PHL(JD)
      GO TO 500
  335 A(IC+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(IC+JE)=RIMP(IR)
      GO TO 500
C        SPECIFIC VOLUME
  350 IF (DHL(JD) .GT. 0.) A(IC+JE)=1./DHL(JD)
      GO TO 500
C        DEVIATOR STRESS - FIRST DIRECTION
  360 A(IC+JE)=SHL(JD)-PHL(JD)
      GO TO 500
C        DEVIATOR STRESS - SECOND DIRECTION
  370 IF (NALPHA .EQ. 2) GO TO 375
      A(IC+JE)=0.5*(PHL(JD)-SHL(JD))
      GO TO 500
  375 A(IC+JE)=SDT(JD)
      GO TO 500
C        DEVIATOR STRESS - THIRD DIRECTION
  380 IF (NALPHA .EQ. 2) GO TO 385
      A(IC+JE)=0.5*(PHL(JD)-SHL(JD))
      GO TO 500
  385 A(IC+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
  505 IF (UNIT(NTAPE)) 520,510,510
  510 NERR=NERR+1
  520 BUFFER OUT(NTAPE,1)(A(1),A(JEDSIZ*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(JEDSIZ*MODLUS+1),A(5000))
      IBUF=0
      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,IH3,M,J,N,D,DOLD,RHOS,SDH,MUM,F,DTNP1,NEMSTRES2  2
     1   ,NET,TSR)                                                      STRES2  3
      REAL MUM,NEM,NET                                                  STRES2  4
      COMMON /S2/ ALF,CO,EEN,EENP1,EPN,KS,TAUEL,TAUI,TAUN,TAUO,VELS,VMU,ALCOM   2
     1  ZAM,ZAMUSV,ZEP,ZEPDSV,ZEPMAXC,ZEPMAXS,ZEPSAVE,ZTAUY,ZTAUYMX     ALCOM   3
      DIMENSION  TAUY(300),TAUYMX(300),EPMAXS(300),EPMAXC(300),         STRES2  6
     1        EPDSV(300),EPSAVE(300),AMUSV(300),EP(300),TAU(300)        STRES2  7
      DIMENSION TSR(6,30)                                               STRES2  8
C                                                                       STRES2  9
C         VALUE OF IND - 0   COMPLETE CALCULATION                       STRES210
C                        1   COMPLETE CALC., EXCEPT FOR RESETTING ARRAYS STRES211
C                        2   ONLY RESET ARRAYS                          STRES212
      IF (LS .GT. 0) GO TO 5                                            STRES213
      KS=0                                                              STRES214
      ZAM= MUM                                                          STRES215
      CO=TSR(M,15)                                                      STRES216
      TAUO=TSR(M,16)                                                    STRES217
      TAUI=TSR(M,17)                                                    STRES218
      ALF=TSR(M,18)                                                     STRES219
      DO 4 I=1,300                                                      STRES220
      TAUY(I)=TAUYMX(I)=EPMAXS(I)=EPMAXC(I)=EPDSV(I)=EPSAVE(I)=EP(I)=   STRES221
     1  TAU(1)=0.                                                       STRES222
      AMUSV(I)=ZAM                                                      STRES223
    4 CONTINUE                                                          STRES224
      LS=1                                                              STRES225
5     IF (IND .EQ. 2) GO TO 100                                         STRES226
      ZTAUY=TAUY(J)        $    ZTAUYMX=TAUYMX(J)   $    ZEPMAXS=EPMAXS(J)  STRES227
      ZEP=EP(J)            $    ZEPMAXC=EPMAXC(J)   $  ZEPDSV=EPDSV(J)  STRES228
      ZEPSAVE=EPSAVE(J)    $    ZAM=ZAMUSV=MUM    $    TAUN=TAU(J)      STRES229
C                                                                       STRES230
C     *** TESTS FOR MATERIAL EXCEEDING MELT ENERGY ***                  STRES231
C                                                                       STRES232
      IF (ZEPMAXS .LT. 0.) GO TO 90                                     STRES233
      IF (F.GT.0.) GO TO 10                                             STRES234
      ZEPMAXS=-1.                                                       STRES235
      SDH=0.                                                            STRES236
      GO TO 90                                                          STRES237
C                                                                       STRES238
C     *** TEST FOR INITIALIZING SHEAR STRESS CALCULATIONS ***           STRES239
C                                                                       STRES240
10    IF (ZEPMAXS .GT. 0.) GO TO 20                                     STRES241
      ENU = D/RHOS                                                      STRES242
      IF (ABS(ENU-1.) .GE. 1.E-6) GO TO 20                              STRES243
      TAUEL=0.                                                          STRES244
      GD TO 70                                                          STRES245
C                                                                       STRES246
C     *** UPDATE STRAIN AND ELASTIC SHEAR STRESS AT TIME(N+1) ***       STRES247
C                                                                       STRES248
   20 KS=0                                                              STRES249
      VELS=-ALOG(D/DOLD)                                                STRES250
      EEN= ALOG(DOLD/RHOS)                                              STRES251
      EENP1=EEN-VELS                                                    STRES252
      EPN=ZEP                                                           STRES253
       ZAMUSV=AMIN1(ZAMUSV,AMAX1(MUM-ALF*ABS(EEN),1.))                  STRES254
      C1=AMIN1(ZAMUSV,AMAX1(ZAM-ALF*ABS(EENP1),1.))                     STRES255
      TAUEL=C1*(EENP1-1.5*EPN)                                          STRES256
C                                                                       STRES257
      IF (ABS(EPN) .LT. 1.E-6 .AND. ABS(TAUN) .LT. TAUO) EPN=0.         STRES258
C                                                                       STRES259
C     *** TEST 1 - TEST FOR EXCEEDING ELASTIC LIMIT AT TIME(N) ***      STRES260
C                                                                       STRES261
      IF (ABS(ZEPMAXS) .GT. 0. .DR. ABS(EPN) .GT. 0.) GO TO 30          STRES262
C                                                                       STRES263
C     *** TEST 2 - TEST FOR EXCEEDING ELASTIC LIMIT AT TIME(N+1) ***    STRES264
C                                                                       STRES265
      IF (ABS(TAUEL) .LT. TAUO) GO TO 70                                STRES266
C                                                                       STRES267
C     *** INITIAL CROSSING OF ELASTIC LIMIT ***                         STRES268
```

SUBROUTINE STRES2 (Concluded)

```
C
      KS=1
      GO TO 60
C
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
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. ABS(ZTAUYMX).GT.1.) GO TO 50
      KS=2
      ZTAUYMX=0.
      GO TO 60
C
C     *** TEST 5 - TEST FOR RELOADING OR REUNLOADING FROM AN ELASTIC
C                  POINT AT TIME(N) ***
C
40    IF (ABS(ZTAUYMX) .GT. ABS(ZTAUY)) KS=3
      GO TO 60
C
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
C
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
C     *** POINT AT TIME(N+1) IS ON ELASTIC CURVE ***
C
70    ZTAUY=TAUEL
      SDH=4.*TAUEL/3.
      ZEPDSV=0.
C
C     *** UPDATE TAUYMX AT TIME(N+1) ***
C
80    ZTAUYMX=AMAX1(ABS(ZTAUYMX),ABS(ZTAUY))*SIGN(1.,ZTAUY)
      IH3=KS
      SDSTORE=SDH
      SDH=SDH*F
      IF (IND .EQ. 0) GO TO 100
   90 RETURN
100   TAUY(J)=ZTAUY        $   TAUYMX(J)=ZTAUYMX    $   EPMAXS(J)=ZEPMAXS
      EPMAXC(J)=ZEPMAXC    $   EPDSV(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,G,H,ES,ROS,EN,E,EQSTVM,EQSTAM,NCYC)  TSQE
C                                                                    C TSQE
C**      CALCULATES MU OR PTH FROM KNOWN PRESSURE AND EOS RELATION. **C  TSQE
C        IP = 0, INVERSE EOS. IP = 1, INVERSE EOS FOR PTH = ALFA*PST.  C TSQE
C                                                                    C TSQE
      NC=0  $  P0=EMU0=P11=0.  $  G2=G/2                                 TSQE
      ESUBC=1.0  $  ENN=0.5                                              TSQE
      IF (EQSTVM .GT. 0. .AND. E .GT. ES) ESUBC=1.+ALOG(E/ES)            TSQE
      IF (EQSTVM .NE. 0.) ENN=ABS(EQSTVM)                                TSQE
      ERAT=E/ES  $  IF (E .GT. ES) ERAT=1.0                              TSQE
      EN2=(EN+ERAT*EQSTAM)/ESUBC  $  ES2=ES*ESUBC                        TSQE
      IF (NC .EQ. 0) IXX=0                                               TSQE
      IF (PP .EQ. 0. .AND. E .LT. ES) GO TO 67                           TSQE
      IND = IP+1                                                         TSQE
      IF (PP .LE. GRE) IND = IND+2                                       TSQE
      EMU1 = (PP-GRE)/C                                                  TSQE
8     NC=NC+1  $  P11=P1                                                 TSQE
      S4=0.                                                              TSQE
      GO TO (10,15,20,25) IND                                            TSQE
C        PATH FOR COMPRESSION - SOLID PRESSURE KNOWN.               ** C TSQE
10    WMU = 1.+EMU1                                                      TSQE
      PH = EMU1*(C+EMU1*(D+EMU1*S))                                      TSQE
      P1 = GRE+PH*(1.-G2*EMU1/WMU)                                       TSQE
      EMU2 = TSQE = EMU1+(PP-P1)*(0.5/(PH*G2/WMU**2+(C+EMU1*(2.*D+EMU1*3 TSQE
     1. *S))*(1.-G2*EMU1/WMU))+0.5*(EMU1-EMU0)/(P1-P0))                  TSQE
      GO TO 30                                                           TSQE
C        PATH FOR EXPANSION - SOLID PRESSURE KNOWN.                 ** C TSQE
20    WMU=1.+EMU1                                                        TSQE
      S1=ROS*WMU                                                         TSQE
      SQ=WMU**ENN                                                        TSQE
      S2=H+(G-H)*SQ                                                      TSQE
      IF (EN2*EMU1/WMU**2 .GT. -30.) S4=EXP(EN2*EMU1/WMU**2)             TSQE
      S3=E-ES2*(1.-S4)                                                   TSQE
      P1=S1*S2*S3                                                        TSQE
      DPDMU=ROS*S2*S3+ROS*S3*ENN*(G-H)*SQ+ROS*S2*ES2*S4*EN2*(1.-EMU1)/   TSQE
     1 WMU**2                                                            TSQE
      EMU2=EMU1+(PP-P1)/DPDMU                                            TSQE
      EMU2=AMAX1(-1.+1.E-8*NC,AMIN1(EMU2,-1.E-8*NC))                     TSQE
      GO TO 30                                                           TSQE
C        PATH FOR COMPRESSION - POROUS PRESSURE KNOWN.              ** C TSQE
15    WMU = 1.+EMU1                                                      TSQE
      ETA = 1.-G2*EMU1/WMU                                               TSQE
      PH = EMU1*(C+EMU1*(D+EMU1*S))                                      TSQE
      P1 = (PH*ETA+GRE)/WMU                                              TSQE
      EMU2 = EMU1+(PP-P1)*(0.5/((ETA*(C+EMU1*(2.*D+EMU1*3*S))-P1-PH*G2/  TSQE
     1 WMU**2)/WMU)+0.5*(EMU1-EMU0)/(P1-P0))                             TSQE
      GO TO 30                                                           TSQE
C        PATH FOR EXPANSION - POROUS PRESSURE KNOWN.                ** C TSQE
25    WMU = 1.+EMU1                                                      TSQE
      SQ=WMU**ENN                                                        TSQE
      S2=H+(G-H)*SQ                                                      TSQE
      IF (EN2*EMU1/WMU**2 .GT. -30.) S4=EXP(EN2*EMU1/WMU**2)             TSQE
      S3=E-ES2*(1.-S4)                                                   TSQE
      P1=ROS*S2*S3                                                       TSQE
      DPDMU=ROS*S2*ES2*S4*EN2*(1.-EMU1)/WMU**3+ROS*S3*(G-H)*ENN*SQ/WMU   TSQE
      EMU2=EMU1+(PP-P1)/DPDMU                                            TSQE
      EMU2=AMAX1(-1.+1.E-8*NC,AMIN1(EMU2,-1.E-8*NC))                     TSQE
30    CONTINUE                                                           TSQE
      IF (NC .GT. 7) PRINT 32,IP,PP,GRE,P1,EMU2,EMU1,EMU0,NC,IXX         TSQE
 32   FORMAT(* IP=*I3,* PP,GRE,P1=*1P3E10.3,* EMU2,EMU1,EMU0=*1P3E12.5,  TSQE
     1 * NC,IXX=*2I3)                                                    TSQE
      IF (NC .EQ. 10) IXX=IXX+1  $  IF (IXX .GT. 10) STOP                TSQE
      IF (ABS(EMU2-EMU1) .GT. 1.E-4*AMAX1(ABS(EMU1),1.E-3)) GO TO 75     TSQE
      TSQE=EMU2                                                          TSQE
65    IF (IP .EQ. 1) TSQE=PTH=PP*(1.+EMU2)                               TSQE
 67   IF (PP .EQ. 0.) TSQE=2*ROS/(1.+SQRT(1.-4./(EN+ERAT*EQSTAM)*ALOG(1. TSQE
     1 -E/ES)))                                                          TSQE
70    RETURN                                                             TSQE
75    CONTINUE                                                           TSQE
      IF (NC .EQ. 13) GO TO 65                                           TSQE
      IF (ABS(P0-PP) .LT. ABS(P1-PP)) GO TO 80                           TSQE
      P0 = P1  $  EMU0 = EMU1                                            TSQE
```

SUBROUTINE TSQE (Concluded)

```
80    IF (PP .GT. GRE) GO TO 90                                          TSQE      74
      IF (P11 .EQ. 0. .OR. (P1-PP)*(PP-P11) .LE. 0.) GO TO 90            TSQE      75
      EMU1=0.5*(EMU1+EMU2)                                               TSQE      76
90    EMU1=EMU2                                                          TSQE      77
95    GO TO 8                                                            TSQE      78
      END                                                                TSQE      79
```

Appendix J

GLOSSARY

Nomenclature of Text

A_w	Atomic weight, g/mole
A_i	Coefficients of the fit between σ_a and $h\nu$
a	Coefficient of Murnaghan equation for pressure, dyn/cm^2
b	Number of cells over which a detonation front is spread, or dimensionless coefficient in Murnaghan equation for pressure
C	Bulk modulus, dyn/cm^2
C_b	10^{-24} cm^2/barn, a conversion factor
C_c	4.186×10^7 erg/cal, a conversion factor
C_e	Effective sound speed, cm/sec
C_F	An effective coefficient of artificial viscosity
C_L	Constant in linear relation between shock velocity and particle velocity, cm/sec
C_P	Specific heat at constant pressure, dyn-cm/g/$^{\circ}$C
C_S	Sound speed, cm/sec
C_0	Coefficient of quadratic artificial viscosity
C_1	Coefficient of linear artificial viscosity
c	Cohesion or shear strength at zero normal stress, dyn/cm^2
D	Density, g/cm^3; or second coefficient in series expansion for Hugoniot pressure, dyn/cm^2
D_x	Detonation velocity, cm/sec
E	Internal energy, erg/g
E_{CJ}	Chapman-Jouguet energy, erg/g
E_e	Effective sublimation energy used in calculation of expanded states, erg/g
E_m	Melt energy, erg/g
E^P	Plastic energy, erg/g

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_B	Fraction of explosive detonated
F_G	Thermal softening factor applied to shear modulus
F_Y	Thermal softening factor applied to yield strength
G	Shear modulus, dyn/cm^2
H	Grüneisen ratio for expanded states
h	Planck's constant, 4.1354×10^{-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'	$1/2\ \sigma'_{ij}\ \sigma'_{ij}$, Second invariant of the deviator stress tensor, dyn^2/cm^4
j	Coordinate or cell numbers
k	Boltzmann's constant, 8.6164×10^{-5} eV/^{o}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	6.02252×10^{23}, Avogadro's number, atom/mole
N_c	Number of cells in a zone or number of constituents in a material
N_ϕ	$\tan^2(45^o + \phi/2)$, 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

Q_x	Energy of an explosive, erg/g
R	Total mechanical stress in direction of propagation, dyn/cm^2
R_x	Geometric ratio between successive cells
r	Radial distance in cylindrical or spherical coordinates, cm
S	Third coefficient in series expansion for Hugoniot pressure, dyn/cm^2
S_L	Coefficient in linear relation between shock velocity and particle velocity
$\bar{S}$	Entropy, erg/g/°C
T	Spall strength, dyn/cm^2; or Kelvin temperature, °K; or time constant for stress relaxation, sec
T_h	Zone thickness, cm
ΔT_n	Duration of the nth source
t	Problem time, sec
t_b	Time of detonation, sec
U	Particle velocity, cm/sec
U_s	Shock velocity, cm/sec
u_{CJ}	Chapman-Jouguet particle velocity, cm/sec
V	Specific volume, cm^3/g
V_{CJ}	Chapman-Jouguet specific volume, cm^3/g
W	$\ln(h\nu)$, with $h\nu$ in keV
X	Coordinate location, cm
X_D	Point of initiation of a detonation, cm
$\bar{X}$	Midcell location, cm
ΔX	Cell size in direction of propagation, cm
ΔX_f	Last cell in a zone, cm
ΔX_1	First cell in a zone, cm
Y	Yield strength, dyn/cm^2
Y_D	Work-hardening coefficient, $dyn/cm^2/(g/cm^3)$
Y_τ	Yield stress in shear, dyn/cm^2
Z	Cell mass (g/cm^2, g/cm, or g for planar, cylindrical, or spherical flow)
α	Volumetric thermal expansion coefficient, 1/°C or coefficient in the Coulomb model of Drucker and Prager

β	Coulomb coefficient for the effect of pressure on yield strength
Γ	Grüneisen's ratio
Γ_e	Effective Grüneisen ratio used in calculation of expanded states
γ	Shear strain, or polytropic gas constant
δ	Added change in thickness between successive cells in an arithmetic layout
δ_{ij}	Kronecker delta: zero for $i \neq j$; one for $i = j$
ε	$1 - \rho_o/\rho$, Lagrangian strain
ε_{ij}	Strain tensor
ε_{ij}^E	Component of the elastic strain tensor
ε_{ij}^P	Component of the plastic strain tensor
ε_{EL}	$Y(2G)$, strain to the Hugoniot elastic limit
$d\bar{\varepsilon}^P$	Equivalent plastic strain, defined in Eq. (4.33)
θ	Temperature, ^{o}C; or angle between the radiation direction and normal incidence on the layer
$d\theta$	Small angle containing the typical cell in cylindrical or spherical geometry
λ	Proportionality factor used in plasticity calculations, cm^2/dyn
μ	$\rho/\rho_o - 1$, a strain
μ_a	Linear absorption coefficient, 1/cm
ν	Vibration frequency, Hz
ξ_i	Dimensionless parabolic interpolation factors
ρ	Density, g/cm^3
ρ_o	Initial density
σ_{ij}	Thermodynamic stress in i direction on j plane, dyn/cm^2
σ'_{ij}	Deviator stress in i direction on j plane, dyn/cm^2
σ'^{N}_{ij}	Deviator stress computed on an elastic basis, dyn/cm^2
$\bar{\sigma}$	Effective stress $\sqrt{\frac{3}{2}\sigma'_{ij}\sigma'_{ij}}$, dyn/cm^2
$\bar{\sigma}^N$	Effective stress based on elastically computed stresses, $\sqrt{\frac{3}{2}\sigma'^{N}_{ij}\sigma'^{N}_{ij}}$, dyn/cm^2

σ_a	Mass absorption coefficient, barns/atom
σ_N	Normal stress, dyn/cm^2
σ_r	Radial stress, dyn/cm^2
σ_θ	Circumferential stress, dyn/cm^2
τ	Shear stress, dyn/cm^2
τ_c	Shear yield stress, dyn/cm^2
ϕ	Angle of internal friction, or yield function. Yield occurs for $\phi = 0$. For negative values of ϕ, behavior is elastic; positive values are not permitted.
ω	$h\nu/KT$, 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, $\lvert$DTMAX$\rvert$ is the number of cells desired in the layer numbered $\lvert$NREZON$\rvert$ (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/cm^2. 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/cm^2. 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/cm^2. 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/cm^2 (input)
MUP(M)	Initial shear modulus in porous material, dyn/cm^2 (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/cm^3
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)

NJEDIT	Number of lines of data in the request for historical listings (input)
NHNU	Number of energy values in an arbitrary spectrum (input)
NLAYER	Number of layers, counting blank layers or gaps. A hollow cylinder or sphere is represented with a gap as the first layer (input)
NMTRLS	Number of materials for which data are supplied in the problem input (input)
NPOINT	Number of points in a depth-dose profile (input)
NPOR(M)	Indicator for a porous material: zero for no model, 1 for POREQST, 2 for PORHOLT, 3 for PEST, and 4 for CAP (input)
NPR(M)	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)
NREZON	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)
NSCRB	An array of indicators for controlling radiation deposition plots from DEPOS (input)
NSPEC	Number of spectra (input)
NTEDT	Number of time edits requested at specified times (input)
NTR	Number of the TEDIT for which a rezone is requested (input)
NVAR(M)	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)
NYAM	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) C (COSQ) or V (VISC): artificial viscosity E (EMELT) or M (MELT): thermal strength reduction factor GM (GMELT): thermal reduction for shear modulus T (TENS): spall strength Y (YIELD): yield strength, shear modulus, work-hardening modulus
NZONES	Number of zones in a layer (input)
PCJ(M)	Chapman-Jouguet pressure dyn/cm^2
PHL(J)	Pressure, dyn/cm^2 (positive in compression)

U(J)	Particle velocity, cm/sec
UZERO	Flyer velocity, cm/sec (input)
VCJ(M)	Chapman-Jouguet specific volume, cm^3/g
VISC	Indicator used with NYAM (see NYAM)
X(J)	Eulerian location of coordinate J, cm
YADD(M)	Work-hardening coefficient, dyn/cm^2 during input and $dyn/cm^2/(g/cm^3)$ after resetting in GENRAT (input)
YHL(J)	Yield strength
YIELD	Indicator used with NYAM (see NYAM)
YOS	Initial yield strength, dyn/cm^2 (input)
ZHL(J)	Cell mass; g/cm^2, g/cm, or g for planar, cylindrical, or spherical geometry

P6	Pressure coefficient in prescribed exponential pressure boundary. P6(1) is for left boundary, P6(2) for right (input), dyn/cm^2
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/cm^2 (positive in compression)
RHO(M)	Initial cell density, g/cm^3 (input)
RHOS(M)	Initial solid density, g/cm^3 (input)
SDH	Deviator stress in the direction of propagation, dyn/cm^2 (positive in compression)
SDT(J)	Deviator stress in the transverse (circumferential) direction in cylindrical geometry, dyn/cm^2
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/cm^2 (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/cm^2 (negative)
TARZ	Problem time when automatic rezoning is terminated, sec (input)
TBL	hν, 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/cm^2 (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

1. J. von Neumann and R. D. Richtmyer, "A Method for the Numerical Calculation of Hydrodynamic Shocks," J. Appl. Phys., 21, p. 232 (1950).
2. R. D. Richtmyer and K. W. Morton, Difference methods for Initial-Value Problems, 2d ed. (interscience Publishers, New York, 1967).
3. J. K. Crosby and S. P. Gill, "Feasibility Study of an Explosive Gun," Final Report No. 2, Stanford Research Institute, Contract NAS2-1361, for NASA-Ames Research Center (May 1966).
4. O. Nance and H. A. Smith, "An X-Ray Effects Code (U)," UCRL-5616-T, University of California, Lawrence Radiation Laboratory, Livermore, California (July 1959) (Secret).
5. O. Nance, M. Wilkins, I. Blandford, and H. Smith, "SHARP--A One-Dimensional Hydrodynamics Code for the IBM 704," UCRL-5206-T, University of California, Lawrence Radiation Laboratory, Livermore, California (April 1958).
6. L. E. Bothell et al., "Equation of State for the PUFF Code (U)," AFSWC-TDR-62-147, Research Directorate, Air Force Special Weapons Center, Kirtland Air Force Base, New Mexico (1963) (Confidential).
7. L. E. Bothell et al., "Predictions of the Response of Materials to X-Rays (U)," AFSWC-TDR-62-152, Research Directorate, Air Force Special Weapons Center, Kirtland Air Force Base, New Mexico (1963) (Secret-RD).
8. L. E. Bothell, "Theoretical Calculations of Shock Propagation in Ablative Materials (U)," AFSWC-TDR-61-63, Research Directorate Air Force Special Weapons Center, Kirtland Air Force Base, New Mexico (March 1962)(Secret-RD).
9. L. E. Bothell, C. E. Archuleta, and R. E. Franke, "PUFF III, Variable Time Step, and Mass absorption Coefficient Codes (U)," AFSWC-TDR-62-151, Research Directorate, Air Force Special Weapons Center, Kirtland Air Force Base, New Mexico (1963) (Secret-RD).
10. L. E. Bothell, C. E. Archuleta, and R. E. Franke, "PUFF IV Hydrocode, Part I(U)," Kaman Nuclear Report KN-664-65-1, Contract DA-01-021-AMC-90023(Y) (November 1965) (Secret-RD).
11. L. E. Bothell and C. E. Archuleta, "PUFF IV-EP Hydrocode, Part I (U)," Kaman Nuclear Final Report KN-664-67-1, Contract DA-01-021-AMC-90023(Y) (January 1967)(Secret-RD).

12. L. E. Bothell and C. E. Archuleta, "PUFF V-EP CODE (U)," Kaman Nuclear Final Report, KN-703-67-101(R), Contract DA-01-021-AMC-90029(Y), for Sentinel Systems Command, SENSC-DRS, Redstone Arsenal, Alabama (December 15, 1967) (Secret-RD).

13. R. A. Grandey, "PUFF V-TS Computer Program (U)," AFSWC-TDR-62-76, Research Directorate, Air Force Special Weapons Center, Kirtland Air Force Base, New Mexico (1963) (Confidential).

14. R. G. Allen, L. K. Goodwin, et al., "A Study of X-Ray Countermeasure Methods (U)," Technical Report No. AFWL-TDR-64-42, Air Force Weapons Laboratory, Kirtland Air Force Base, New Mexico (June 1964) (Secret-RD).

15. R. N. Brodie and J. E. Hormuth, "The PUFF 66 and P PUFF 66 Computer Programs," Research and Technology Division, Air Force Weapons Laboratory, Kirtland Air Force Base, New Mexico (May 1966).

16. W. Herrmann, P. Holzhauser, and R. J. Thompson, "WONDY, A Computer Program for Calculating Problems of Motion in One Dimension," SC-RR-66-601, Sandia Corporation, Albuquerque, New Mexico (February 1967).

17. Lynn Seaman, "SRI PUFF 3 Computer Code for Stress Wave Propagation," Technical Report No. AFWL-TR-70-51, Air Force Weapons Laboratory, Kirtland Air Force Base, New Mexico (September 1970).

18. R. H. Fisher, G. A. Lane, and R. A. Cecil, "RIP, A One-Dimensional Material Response Code," Report No. 3SR-751-I, by Systems, Science and Software for Defense Nuclear Agency, Washington, D.C. (September 1972).

19. L. Seaman, R. F. Williams, J. T. Rosenberg, D. C. Erlich, and R. K. Linde, "Classification of Materials by Shock Properties," Final Report No. AFWL-TR-69-96 by SRI for Air Force Weapons Laboratory, Kirtland Air Force Base, New Mexico (November 1969).

20. L. Seaman and J. T. Rosenberg, "Characterization of Homogeneous Materials in Compression and Tension," Final Report No. AFWL-TR-73-15 by SRI for Air Force Weapons Laboratory, Kirtland Air Force Base, New Mexico (January 1973).

21. D. A. Shockey, L. Seaman, and D. R. Curran, "Dynamic Fracture of Beryllium under Plate Impact and Correlation with Electron Beam and Underground Test Results," Final Report No. AFWL-TR-73-12 by SRI for Air Force Weapons Laboratory, Kirtland Air Force Base, New Mexico (January 1973).

22. L. Seaman, T. W. Barbee, Jr., and D. R. Curran, "Dynamic Fracture Criteria of Homogeneous Materials," Final Report No. AFWL-TR-71-156 by SRI for Air Force Weapons Laboratory, Kirtland Air Force Base, New Mexico (December 1971).

23. L. Seaman and D. A. Shockey, "Models for Ductile and Brittle Fracture for Two-Dimensional Wave Propagation Calculations," Final Report AMMRC-CTR-75-2 by SRI for Army Materials and Mechanics Research Center, Watertown, Massachusetts (February 1975).

24. D. A. Shockey, L. Seaman, D. R. Curran, P. S. De Carli, M. Austin, and J. P. Wilhelm, "A Computational Model for Fragmentation of Armor Under Ballistic Impact," Final Report by SRI for U.S. Army Ballistic Research Laboratories, Aberdeen Proving Ground, Maryland (December 1973).

25. L. Seaman, D. R. Curran, and D. A. Shockey, "Computational Models for Ductile and Brittle Fracture,: J. Appl. Phys., Vol. 47, No. 11, pp. 4814-4826 (November 1976).

26. T. Barbee, L. Seaman, and R. C. Crewdson, "Dynamic Fracture Criteria of Homogeneous Materials," Technical Report No. AFWL-TR-70-99 by SRI for Air Force Weapons Laboratory, Kirtland Air Force Base, New Mexico (November 1970).

27. L. K. Goodwin, L. A. Johnson, and R. S. Wright, "An Equation of State of Metals," DASA-2286, Aeronutronic Division of Philco-Ford Corporation, Newport Beach, California (April 1969).

28. L. Seaman, R. E. Tokheim, and D. R. Curran, "Computational Representation of Constitutive Relations for Porous Materials," Final Report No. DNA 3412F by SRI for Defense Nuclear Agency, Washington, D.C. (May 1974).

29. E. B. Royce, "GRAY, a Three-Phase Equation of State for Metals," UCRL-51121, Lawrence Livermore Laboratory, Livermore, California (September 3, 1971).

30. D. E. Grady, C. W. Smith, G. M. Muller, K. D. Mahrer, and C. F. Petersen, "In Situ Constitutive Relations of Soils and Rocks," Final Report DNA-3671Z by SRI for Defense Nuclear Agency, Washington, D.C. (March 1974).

31. L. Seaman and R. K. Linde, "Distended Material Model Development, Vol. 1: Experiments and Theory for the Model," AFWL-TR-68-143, Stanford Research Institute, Menlo Park, California (May 1969).

32. A. Holt, A. Kusubov, M. Carroll, and B. Hord, "Stress-Wave Propagation in Distended Asbestos Phenolic," UCRL-51120, Lawrence Livermore Laboratory, Livermore, California (September 3, 1971).

33. Y. M. Gupta and L. Seaman, "Local Response of Reinforced Concrete to Missile Impacts," Final Report by SRI for Electric Power Research Institute, Palo Alto, California (October 1977).

34. L. Seaman, D. A. Shockey, D. R. Curran, and R. E. Tokheim, "Development of a Shear Band Model for Fragmentation in Exploding Cylinders," Final Report by SRI for Naval Surface Weapons Laboratory, Dahlgren, Virginia (August 1975).

35. D. C. Erlich, L. Seaman, D. A. Shockey, and D. R. Curran, "Development and Application of a Computational Shear Band Model," Final Report by SRI for U.S. Army Ballistic Research Laboratory, Aberdeen Proving Ground, Maryland (June 1977).

36. H. E. Read and R. A. Cecil, "A Rate-Dependent Constitutive Model for Shock-Loaded S-200 Beryllium," DNA-2845F, Systems, Science and Software, La Jolla, California (February 1972).

37. M. H. Rice, R. G. McQueen, and J. M. Walsh, "Compression of Solids by Strong Shock Waves," Solid State Physics, Vol. 6, F. Seitz and D. Turnbull, Eds. (Academic Press, New York, 1958).

38. D. R. Stull et al., JANAF Thermochemical Tables, Dow Chemical Company, Contract No. AF 04(611)-7554, sponsored by Project Principia of the Advanced Research Projects Agency (August 1965), with Addenda published in August 1966 and August 1967.

39. K. Terzaghi, Theoretical Soil Mechanics (John Wiley and Sons, Inc., New York, 1943).

40. D. C. Drucker and W. Prager, Soil Mechanics and Plastic Analysis or Limit Design," Quarterly of Applied Mathematics, Vol. 10, p. 157 (1952).

41. W. H. McMasters et al., "Compilation of X-ray Cross Sections," University of California, Livermore, Report No. UCRL-50174, Section II Revision 1 (May 1969).

42. R. H. Fisher and J. W. Wiehe, "A User's Guide to the FSCATT Code," Final Report No. DASA 2618 by Systems, Science and Software for Defense Atomic Support Agency, Washington, D.C. (November 1970).

43. F. W. Sears, Optics (Addison-Wesley Press, Inc., Cambridge, Massachusetts, 1949).

44. J. Taylor, Detonation in Condensed Explosives (Clarendon Press, Oxford, 1952).

45. R. Hill, The Mathematical Theory of Plasticity (Clarendon Press, Oxford, 1950).

46. M. L. Wilkins, "Calculation of Elastic-Plastic Flow," University of California, Lawrence Radiation Laboratory, Livermore, California, in Fundamental Methods in Hydrodynamics, Vol. 3 (Academic Press, New York and London, 1964).

DISTRIBUTION LIST

No. of Copies	Organization
12	Commander Defense Technical Info Center ATTN: DDC-DDA Cameron Station Alexandria, VA 22314
4	Director Defense Advanced Research Projects Agency ATTN: Tech Info Dr. Ernest F. Blase Dr. Bement Dr. Ray Gogolewski 1400 Wilson Boulevard Arlington, VA 22209
1	Deputy Assistant Secretary of the Army (R&D) Department of the Army Washington, DC 20310
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCDMD-ST 5001 Eisenhower Avenue Alexandria, VA 22333
6	Commander US Army Armament Research and Development Command ATTN: DRDAR-TSS (2 Cys) J. D. Corrie R. J. Weimer J. Beetle E. Bloore Dover, NJ 07801
1	Commander US Army Armament Materiel Readiness Command ATTN: DRSAR-LEP-L, Tech Lib Rock Island, IL 61299
5	Commander US Army Watervliet Arsenal ATTN: Dr. T. Davidson Dr. M. A. Hussain Dr. S. L. Pu Mr. D. P. Kendall Dr. John Underwood Watervliet, NY 12189
1	Commander US Army Aviation Research and Development Command ATTN: DRSAV-E P. O. Box 209 St. Louis, MO 63166
1	Director US Army Air Mobility Research and Development Laboratory Ames Research Center Moffett Field, CA 94035
1	Commander US Army Communications Research and Development Command ATTN: DRDCO-PPA-SA Fort Monmouth, NJ 07703
1	Commander US Army Electronics Research and Development Command Technical Support Activity ATTN: DELSD-L Fort Monmouth, NJ 07703
1	Commander US Army Harry Diamond Labs ATTN: DRXDO-TI 2800 Powder Mill Road Adelphi, MD 20783

DISTRIBUTION LIST

No. of Copies — Organization

3
Commander
US Army Missile Research
and Development Command
ATTN: DRDMI-R
DRDMI-YDL
Dr. Raymond Conrad
Redstone Arsenal, AL 35809

3
Commander
US Army Mobility Equipment
Research & Development Cmd
ATTN: DRDME-WC
DRSME-RZT
STSFB-MW, Dr. J. Bond
Fort Belvoir, VA 22060

1
Commander
US Army Tank Automotive
Research & Development Cmd
ATTN: DRDTA-UL
Warren, MI 48090

1
Director
US Army TRADOC Systems
Analysis Activity
ATTN: ATAA-SL, Tech Lib
White Sands Missile Range
NM 88002

5
Commander
US Army Materials and
Mechanics Research Center
ATTN: DRXMR-ATL
Dr. E. Wright
Mr. R. Shea
DRXMRR-H, Dr. D. Dandekar
DRXMR-T, Mr. J. Mescall
Watertown, MA 02172

No. of Copies — Organization

4
Commander
US Army Research Office
ATTN: Dr. Hermann Robl
Dr. E. Saibel
Dr. George Mayer
Dr. James Murray
P. O. Box 12211
Research Triangle Park
NC 27709

1
Commander
US Army Research and
Standardization Group (Europe)
Box 65
ATTN: Dr. B. Steverding
Box 65
FBO NY 09510

1
Commander
US Army War College
ATTN: Lib
Carlisle Barracks, PA 17013

1
Commander
US Army Command and General
Staff College
ATTN: Archives
Fort Leavenworth, KS 66027

1
Commander
US Military Academy
ATTN: Library
West Point, NY 10996

1
Office of Naval Research
ATTN: Code 402
Department of the Navy
Washington, DC 20360

1
Commander
Naval Research Laboratory
ATTN: Code 2020, Tech Lib
Washington, DC 20375

DISTRIBUTION LIST

No. of Copies | Organization

7 Commander
Naval Research Laboratory
Engineering Materials Division
ATTN: E. A. Lange
G. R. Yoder
C. A. Griffis
R. J. Goode
R. W. Judy, Jr.
A. M. Sullivan
R. W. Crooker
Washington, DC 20375

1 Commander
Naval Research Laboratory
Metallurgy Division
ATTN: W. S. Pellini
Washington, DC 20375

2 Commander
Naval Surface Weapons Ctr
ATTN: Mr. W. H. Holt
Tech Lib
Dahlgren, VA 22445

2 Commander
Naval Surface Weapons Ctr
ATTN: Dr. Robert Crowe
Tech Lib
Silver Spring, MD 20910

1 AFOSR(Dr. Alan H. Rosenstein)
Bolling AFB, DC 20332

1 AFWL(Tech Lib)
Kirtland AFB, NM 87117

1 Director
Lawrence Livermore Laboratory
ATTN: Dr. M. L. Wilkins
P. O. Box 808
Livermore, CA 94550

No. of Copies | Organization

6 Sandia Laboratories
ATTN: Tech Lib
Dr. A. L. Stevens
Dr. Lee Davison
Dr. W. E. Warren
Dr. L. D. Bertholf
Dr. Marlin Kipp
Albuquerque, NM 87115

7 SRI International
ATTN: Dr. George R. Abrahamson
Dr. Donald R. Curran
Dr. Donald A. Shockey
Dr. Lynn Seaman
Dr. R. Caligiuri
Mr. D. Erlich
Mr. R. Burback
333 Ravenswood Avenue
Menlo Park, CA 94025

<u>Aberdeen Proving Ground</u>

Dir, USAMSAA
ATTN: DRXSY-D
DRXSY-MP, H. Cohen
Cdr, USATECOM
ATTN: DRSTE-TO-F
Dir, Wpns Sys Concepts Team,
Bldg. E3516, EA
ATTN: DRDAR-ACW

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, ARRADCOM, 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:____

____

____