FINITE-DIFFERENCE MODEL OF TWO-DIMENSIONAL, SINGLE-, AND TWO-PHASE HEAT TRANSPORT IN A POROUS MEDIUM—VERSION I

By CHARLES R. FAUST and JAMES W. MERCER

U.S. GEOLOGICAL SURVEY
Open-File Report 77—234

1977

UNITED STATES DEPARTMENT OF THE INTERIOR

CECIL D. ANDRUS, Secretary

GEOLOGICAL SURVEY

V.E. McKelvey, Director

For additional information write to:

Chief Hydrologist
U.S. Geological Survey, WRD
410 National Center
Reston, Virginia 22092

CONTENTS

FIGURES

TABLES

Finite-Difference Model of Two-Dimensional, Single-, and Two-Phase Heat Transport in a Porous Medium——Version I

by

Charles R. Faust and James W. Mercer

ABSTRACT

Model documentation is presented for a two-dimensional (areal) heat-transport model capable of simulating both water- and vapor-dominated geothermal reservoirs that conform with the assumptions of the model. Finite-difference techniques are used to solve for the dependent variables pressure and enthalpy. The program is designed to simulate time-dependent problems such as those associated with geothermal reservoirs undergoing exploitation, and can treat the transition from compressed water to two-phase flow. In order to simulate more complicated field problems the present program is being extended, and therefore the model described in this report is referred to as VERSION I. A listing of the computer code is included.

INTRODUCTION

The continuity equations for steam and water are reduced to two nonlinear partial differential equations in which the dependent variables are fluid pressure and enthalpy. These equations are approximated using finite-difference techniques and are solved using a direct matrix technique. The nonlinear coefficients are calculated using Newton-Raphson iteration on the accumulation terms, and an option is provided for using either upstream or midpoint weighting on the mobility terms. The model can simulate flow of compressed water, two-phase mixtures, and super-heated steam over a temperature range of 10° to 300°C. In addition, it can handle the conversion from single-phase flow to two-phase flow.

The model described in this report is referred to as VERSION I and is considered to be a research tool; that is, this version is kept as simple as possible so that the program can be easily understood and modified. For this reason, many sophisticated changes and additions which are generally required to simulate complicated field problems are not included. Such changes are the subject of current research and will be described in subsequent reports.

MODEL DESCRIPTION

Theoretical Development

The computer program presented in this report is based on a theoretical development originally outlined by Mercer, Faust, and Pinder (1974) and presented in detail by Faust (1976). Only the final equations and major assumptions are given here.

Major assumptions include the following:

1) capillary pressure is negligible;
2) local thermal equilibrium exists among all phases;
3) Darcy's equations for two-phase flow are valid;
4) the thermodynamic properties of the geothermal fluid are those of a pure-water system;
5) thermal dispersion/conduction is treated as a property of the medium;
6) flow in the reservoir can be represented by using a two-dimensional (areal) model; and
7) the reservoir is a confined porous medium.

Final Equations

Based on these assumptions the equations describing flow and heat transport in a geothermal reservoir are (Faust, 1976):

$$\nabla\cdot\left(\frac{b\rho_s kk_{rs}}{\mu_s}\nabla p\right)+\nabla\cdot\left(\frac{b\rho_w kk_{rw}}{\mu_w}\nabla p\right) \quad + bq_s'+bq_w' = b\frac{\partial(\phi\rho)}{\partial t} \tag{1}$$

and

$$\nabla\cdot\left(\frac{b\rho_s h_s kk_{rs}}{\mu_s}\nabla p\right) + \nabla\cdot\left(\frac{b\rho_w h_w kk_{rw}}{\mu_w}\nabla p\right) \quad + \nabla\cdot\{K_m b[(\frac{\partial T}{\partial p})_h\nabla p+(\frac{\partial T}{\partial h})_p\nabla h]\} \tag{2}$$

$$+ bq_s'h_s'+bq_w'h_w'+q'' \quad = b\frac{\partial}{\partial t}[\phi\rho h+(1-\phi)\rho_r h_r]$$

where the term, q", represents the conductive-heat gain (or loss) to the confining beds (overburden and underburden), and may be obtained from,

$$q''=K_r\frac{\partial T}{\partial z}\Big|_{\substack{\text{overburden}\\\text{contact}}} - K_r\frac{\partial T}{\partial z}\Big|_{\substack{\text{underburden}\\\text{contact}}} \tag{3}$$

Definitions of the parameters are included in the section on notation.

Auxiliary Relationships

Additional assumptions and relationships include the following:

1) The fluid enthalpy, h, of the mixture is defined in the two-phase region by:

$$h = \frac{S_s\rho_s h_s + S_s\rho_w h_w}{\rho} \quad . \tag{4}$$

2) The density, ρ, of the mixture is defined by

$$\rho = S_s\rho_s + S_w\rho_w \quad . \tag{5}$$

3) Phase saturations sum to one:

$$S_s + S_w = 1 \quad . \tag{6}$$

4) Porosity is a function of pressure, and can be expanded about an initial porosity distribution by a truncated Taylor series with first order pressure terms:

$$\phi = \phi_i[1+\beta(p-p_i)], \tag{7}$$

where the subscript i indicates initial values.

5) Phase viscositites are functions of temperature (Meyer and others, 1967; modified for the cgs system),

$$\mu_s = 10^{-6}(0.407 \cdot T + 80.4), \tag{8}$$

and,

$$\mu_w = 10^{-6} \left\{241.4 \cdot 10^{[247.8/(T+133.15)]}\right\}, \qquad (9)$$

where,

μ_s = dynamic viscosity of steam, g/cm-sec,

μ_w = dynamic viscosity of water, g/cm-sec,

T = temperature, °C.

Equation 8 is valid for superheated steam at 10^6 dynes per square centimeter pressure in the temperature range of 100 to 300 degrees Celsius, and is approximately valid for steam viscosity along the saturation line in that range. Equation 9 is valid for liquid water along the saturation line from 0 to 300 degrees Celsius.

6) The relative permeability expressions are functions of saturation and are a variation of those given by Corey (1954) for a drainage displacement process, that is, vaporization dominates condensation:

$$k_{rw} = \frac{(S_w - S_{wr} - S_{sr})^4}{(1 - S_{wr} - S_{sr})^4}, \qquad (10)$$

and

$$k_{rs} = \left[1 - \frac{(S_w - S_{wr} - S_{sr})}{(1 - S_{wr} - S_{sr})}\right]^2 \cdot \left[1 - \frac{(S_w - S_{wr} - S_{sr})^2}{(1 - S_{wr} - S_{sr})^2}\right], \quad (11)$$

where S_{wr} and S_{sr} are specified residual water and steam saturations.

7) Reservoir thickness, rock density, rock specific heat and intrinsic permeability are functions of the spatial coordinates.

8) Rock enthalpy may be determined by the expression,

$$h_r = c_r T \quad (12)$$

where the rock enthalpy is in ergs per gram, the temperature is in degrees Celsius, and the formation heat capacity, c_r, is in ergs per gram per degree Celsius.

9) In the two-phase region, the amount of heat lost to the well is defined as

$$q_h = q'_s h'_s + q'_w h'_w , \quad (13)$$

and the total mass lost to the well as,

$$q_m = q'_s + q'_w . \quad (14)$$

The steam production rate is determined by the fractional flow of the steam phase as follows,

$$q'_s = \sigma_s q_m \quad , \tag{15}$$

where

$$\sigma_s = k_{rs}/(k_{rs} + \frac{\rho_w \mu_s}{\rho_s \mu_w} k_{rw}).$$

Since h'_s, h'_w and q_m are known, q'_s is computed using equation 15, q'_w is computed using equation 14, and q_h is computed using equation 13. Finally, equations relating the thermodynamic properties of pure water and steam to enthalpy and pressure were determined by least square regressions of published experimental data. Figure 1 shows a pressure-enthalpy plot for pure water, this diagram may be divided into the subregions: compressed water, two-phase (steam-water), and superheated steam. Data were obtained from Meyer and others (1967) and Keenan and others (1969) for an enthalpy range of 2.09×10^9 to 3.175×10^{10} ergs per gram, a pressure range of 1.0×10^6 to 1.75×10^8 dynes per square centimeter, and a temperature range of 10 to 300 degrees Celsius. The following expressions, having a maximum error of 0.5 percent within the observed data range, were determined:

10) Steam enthalpy, h_s, and water enthalpy, h_w, are treated as functions of pressure:

$$h_s = 2.82282 \cdot 10^{10} - 3.91952 \cdot 10^{15}/p + 2.54342 \cdot 10^{21}/p^2 - 9.38879 \cdot 10^{-8} \cdot p^2 \tag{16}$$

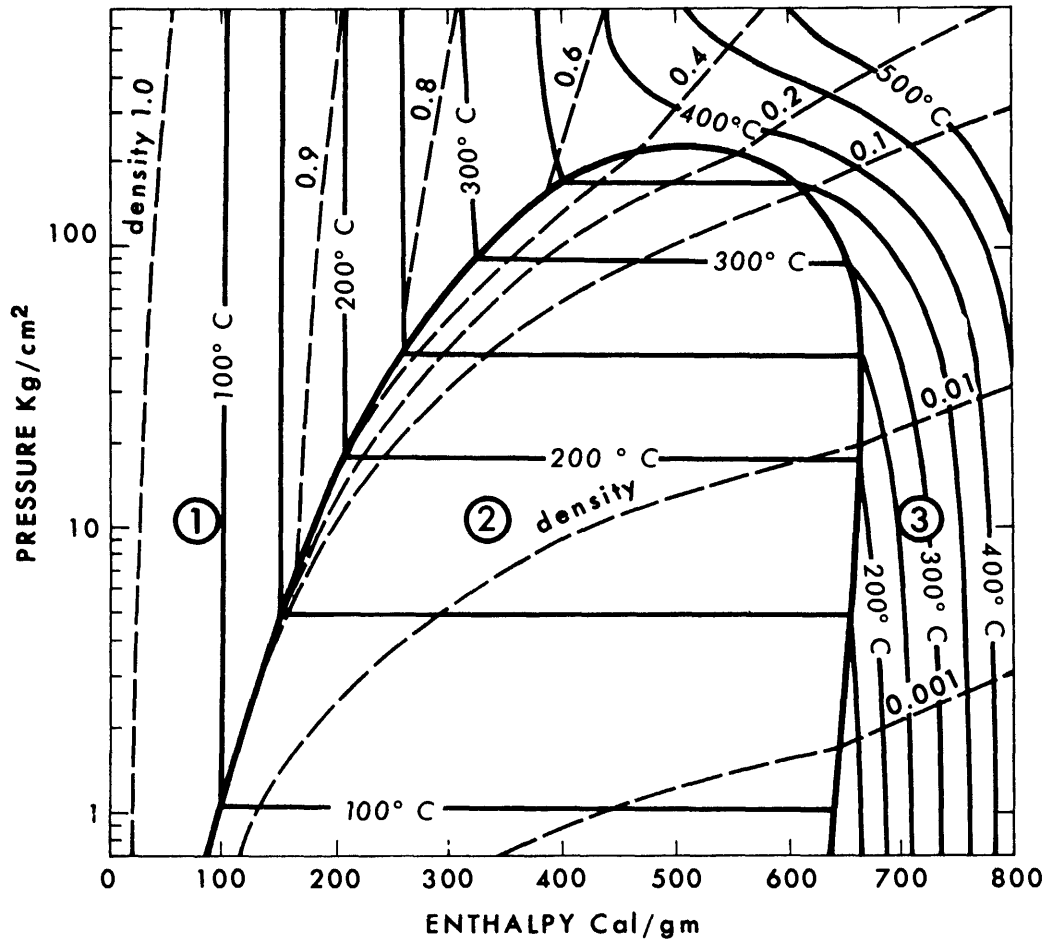


Figure 1. Pressure-enthalpy diagram for pure water and vapor showing three thermodynamic regions below the critical point: 1) compressed water, 2) two-phase steam and water and 3) superheated steam (modified from White, Muffler, and Truesdell, 1971).

and

$$h_w = 7.30984 \cdot 10^9 + 1.29239 \cdot 10^2\, p - 1.00333$$
$$10^{-6} \cdot p^2 + 3.9881 \cdot 10^{-15} \cdot p^3$$
$$- 9.90697 \cdot 10^{15}/p + 1.29267 \cdot 10^{22}/p^2$$
$$- 6.28359 \cdot 10^{27}/p^3 \,, \qquad (17)$$

where

p = pressure, dynes/cm^2

h = enthalpy, ergs/g

h_s = enthalpy of saturated steam, ergs/g

h_w = enthalpy of saturated water, ergs/g.

11) Temperature is treated as a function of pressure and enthalpy for the compressed-water and superheated-steam regions. For the compressed-water region,

$$T = - 2.41231 + 2.56222 \cdot 10^{-8} \cdot h$$
$$- 9.31415 \cdot 10^{-17} \cdot p^2 \qquad (18)$$
$$- 2.2568 \cdot 10^{-19} \cdot h^2 \,,$$

and for the superheated-steam region,

$$
\begin{aligned}
T = &- 374.669 + 4.79921 \cdot 10^{-6} \cdot p \\
&- 6.33606 \cdot 10^{-15} \cdot p^2 \\
&+ 7.39386 \cdot 10^{-19} \cdot h^2 \\
&- 3.3372 \cdot 10^{34}/h^2p^2 \\
&+ 3.57154 \cdot 10^{19}/p^3 \\
&- 1.1725 \cdot 10^{-37} \cdot h^3p \\
&- 2.26861 \cdot 10^{43}/h^4 ,
\end{aligned}
\tag{19}
$$

where the temperature, T, is in degrees Celsius. For the two-phase (steam-water) region h_w is used in place of h in equation 18.

12) Total density, ρ, steam and water densities, ρ_s and ρ_w are considered functions of pressure and enthalpy. For the compressed-water region,

$$
\begin{aligned}
\rho = \rho_w = &\ 1.00207 + 4.42607 \cdot 10^{-11} \cdot p \\
&- 5.47456 \cdot 10^{-12} \cdot h \\
&+ 5.02875 \cdot 10^{-21} \cdot hp \\
&- 1.24791 \cdot 10^{-21} \cdot h^2 ,
\end{aligned}
\tag{20}
$$

and for the superheated-steam region,

$$\rho = \rho_s = -2.26162 \cdot 10^{-5} + 4.38441 \cdot 10^{-9} \cdot p - 1.79088 \cdot 10^{-19} \cdot ph + 3.69276 \cdot 10^{-36} \cdot p^4 + 5.17644 \cdot 10^{-41} \cdot ph^3 , \quad (21)$$

where density is in grams per cubic centimeter. For the steam-water region, saturation pressures and enthalpies are used in equation 20 and 21 to obtain ρ_s and ρ_w .

Numerical Development

The technique used to solve equations 1 and 2 is based on the finite-difference method. For this method the areal extent of the reservoir is subdivided into rectangular grid blocks (see figure 2) in which the fluid and reservoir properties are assumed uniform. The continuous derivatives in equations 1 and 2 are approximated by finite-difference expressions at points (nodes) in the centers of the blocks. This results in a nonlinear system of 2n equations with 2n unknowns (the values of pressure and enthalpy at the nodes) where n is the number of nodes. The general finite-difference representation and solution procedure for this system of nonlinear equations are outlined below.

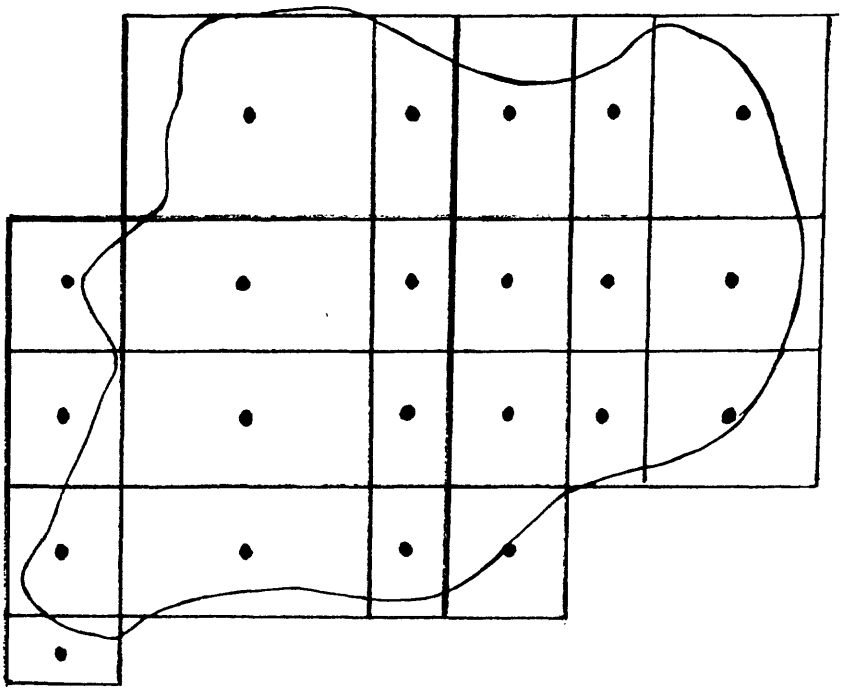


Figure 2. Finite-difference grid showing the locations of the reservoir, the grid blocks and nodes.

Finite-Difference Representation

Equations 1 and 2 may be written in compact, implicit finite-difference form as:

$$\Delta_x[(T_{wx}+T_{sx})\Delta_x p^{n+1}] + \Delta_y[(T_{wy}+T_{sy})\Delta_y p^{n+1}] + Vq'_m = \frac{1}{\Delta t}(M^{n+1}-M^n) \quad , \tag{22}$$

and

$$\Delta_x(T_{hx}\Delta_x p^{n+1}) + \Delta_y(T_{hy}\Delta_y p^{n+1}) + \Delta_x(T_{cx}\Delta_x h^{n+1}) + \Delta_y(T_{cy}\Delta_y h^{n+1}) + Vq'_h + \frac{V}{b}q'' = \frac{1}{\Delta t}(E^{n+1}-E^n) \quad , \tag{23}$$

where the right sides of equations 22 and 23 are the accumulation terms for mass and energy, respectively, and q'_m and q'_h are the total mass and heat lost to wells, respectively.

The transmissibility terms T_w , T_s , T_h , and T_c , are given by:

$$T_w = (kA/\ell)\rho_w k_{rw}/\mu_w \ , \tag{24a}$$

$$T_s = (kA/\ell)\rho_s k_{rs}/\mu_s \ , \tag{24b}$$

$$T_h = T_w h_w + T_s h_s + (AK_m/\ell)\left(\frac{\partial T}{\partial p}\right)_h \ , \tag{24c}$$

$$T_c = (AK_m/\ell)\left(\frac{\partial T}{\partial h}\right)_p \ , \tag{24d}$$

and the mass and energy terms M and E are:

$$M = V\phi\rho \ , \tag{25a}$$

$$E = V[\phi\rho h + (1-\phi)\rho_r h_r] \ , \tag{25b}$$

where V, A and ℓ are the grid block volume, cross-sectional area perpendicular to the flow direction, and the length increment in the flow direction, respectively. The difference operator acts as follows:

$$\Delta_x(T_{wx}\Delta_x p^{t+\Delta t}) = T_{wx_{i+\frac{1}{2},j}}\left(p^{t+\Delta t}_{i+1,j} - p^{t+\Delta t}_{i,j}\right)$$

$$- T_{wx_{i-\frac{1}{2},j}}\left(p^{t+\Delta t}_{i,j} - p^{t+\Delta t}_{i-1,j}\right) \ , \tag{26}$$

where i and j are indices in the x- and y- directions, and t is the index for the time level.

The interblock transmissibility terms (values at $i + \frac{1}{2}$, $i - \frac{1}{2}$, $j + \frac{1}{2}$ and $j - \frac{1}{2}$) are composed of two parts: that which is a function of space only (for example, kA/ℓ) and that which is a nonlinear function of pressure and/or enthalpy (for example $\rho k_r/\mu$). To approximate these terms requires averaging or weighting of the various components over each grid block. For the space dependent part, this is accomplished by using a harmonic mean, for example,

$$(kA/\ell)_{i+\frac{1}{2}} = \frac{2k_{i+1}k_iA_{i+1}A_i}{k_{i+1}A_{i+1}\ell_i + k_iA_i\ell_{i+1}} \tag{27}$$

The nonlinear part of the transmissibility terms are generally assigned the upstream value. The upstream node is determined by comparing the pressures at (i) and (i+1), and using the larger pressure to compute the nonlinear part. Alternatively this part may be determined by a length weighted arithmetic average, for example,

$$k_{rw_{i+\frac{1}{2}}} = \frac{k_{rw_{i+1}}\ell_{i+1} + k_{rw_i}\ell_i}{\ell_{i+1} + \ell_i} \tag{28}$$

Of the two procedures, upstream weighting yields a lower order approximation of the spatial derivative but exhibits a more stable solution.

Solution Procedure

The difference equations 22 and 23 are solved simultaneously for the unknown pressure and enthalpy in each grid block for each time step. Since equations 22 and 23 are nonlinear, a provision is included to iterate on nonlinear coefficients. Newton-Raphson iteration is used on the accumulation terms and Picard iteration is used on the coefficients of the spatial derivatives. Two difference equations are obtained for each grid block, and the resulting system of 2n equation has the form,

$$[B]\ \{X^{t+\Delta t}\} - \{f(X^{t+\Delta t})\} + \{f(X^{t})\} + \{q\} = 0\ , \qquad (29)$$

where the superscript indicates the time level, the matrix [B] incorporates the transmissibility terms, the vector {X} contains the unknown pressure and enthalpy values, that is

$$\{X\} = \begin{Bmatrix} p_1 \\ h_1 \\ p_2 \\ h_2 \\ \vdots \\ p_n \\ h_n \end{Bmatrix}$$

and the vector {f(X)} is a non-linear function describing the accumulation terms.

To linearize 29, an iterative technique called the Newton-Raphson procedure is applied. The iteration level is indicated by a subscript, and in particular the first iteration (initial guess) is indicated by the subscript (o). Substitution of the initial guess into equation 29 yields a residual:

$$\left[B_o\right]\{X_o^{t+\Delta t}\} - \{f(X_o^{t+\Delta t})\} + \{f(X^t)\} + \{q_o\} = \{R(X_o^{t+\Delta t})\} \ . \tag{30}$$

For the first time step, the initial conditions are used as the initial guess; for subsequent time steps, the results from the previous time step are used as the initial guess.

If the residual is expanded in a truncated first order Taylor series, one obtains,

$$\{R(X^{t+\Delta t})\} = \{R(X_o^{t+\Delta t})\} + \{\Delta X\}\left[\frac{\partial\{R(X_o^{t+\Delta t})\}}{\partial X_o^{t+\Delta t}}\right] \doteq 0 \ ,$$

or rearranging,

$$\left[\frac{\partial\{R(X_o^{t+\Delta t})\}}{\partial X_o^{t+\Delta t}}\right]\{\Delta X\} = - \{R(X_o^{t+\Delta t})\} \ . \tag{31}$$

Taking the derivatives with respect to the unknown vector {X} equation 35 yields,

$$\left[\frac{\partial\{R(X_o^{t+\Delta t})\}}{\partial X_o^{t+\Delta t}}\right] = \left[B_o\right] - \left[\frac{\partial\{f(X_o^{t+\Delta t})\}}{\partial X_o^{t+\Delta t}}\right] \ , \tag{32}$$

where $\left[\frac{\partial\{f(X_o^{t+\Delta t}\}}{\partial X_o^{t+\Delta t}}\right]$ is obtained by taking the partial derivatives of the accumulation terms with respect to pressure and enthalpy. Specifically, by use of the chain rule:

$$b\,\frac{\partial(\phi\rho)}{\partial p} = b\left[\rho\frac{d\phi}{dp} + \left(\frac{\partial\rho}{\partial p}\right)_h\right]$$

$$b\,\frac{\partial(\phi\rho)}{\partial h} = b\left(\frac{\partial\rho}{\partial h}\right)_p ,$$

$$b\,\frac{\partial}{\partial p}\left[\phi\rho h + (1-\phi)\rho_r h_r\right] = b\left[(\rho h - \rho_r h_r)\,\frac{d\phi}{dp}\right.$$

$$\left. + h\left(\frac{\partial\rho}{\partial p}\right)_h + (1-\phi)\rho_r c_r\left(\frac{\partial T}{\partial p}\right)_h\right] ,$$

and

$$b\,\frac{\partial}{\partial h}\left[\phi\rho h + (1-\phi)\rho_r h_r\right] = b\left[\phi h\left(\frac{\partial\rho}{\partial h}\right)_p + (1-\phi)\rho_r c_r\left(\frac{\partial T}{\partial h}\right)_p + \phi\rho\right] .$$

Substituting 32 into equation 31 gives,

$$\left[\left[B_o\right] - \left[\frac{\partial\{f(X_o^{t+\Delta t})\}}{\partial X_o^{t+\Delta t}}\right]\right]\{\Delta X\} = -\{R(X_o^{t+\Delta t})\} , \qquad (33)$$

which is the matrix equation to be solved.

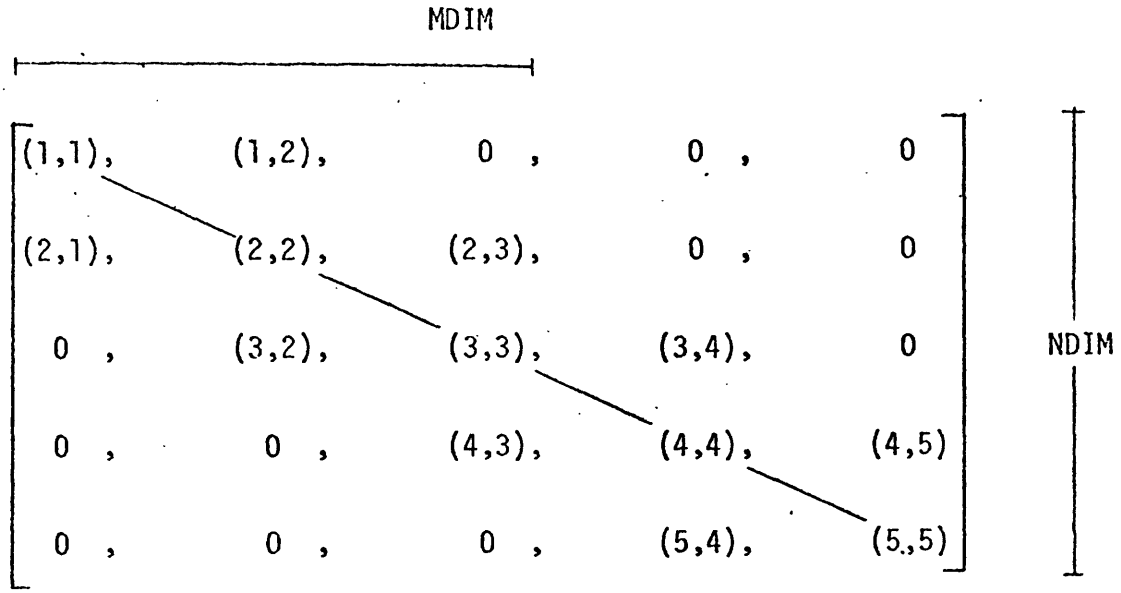
The new values of $X^{t+\Delta t}$ are determined from,

$$\{X_i^{t+\Delta t}\} = \{X_{i-1}^{t+\Delta t}\} + \{\Delta X\} \quad , \tag{34}$$

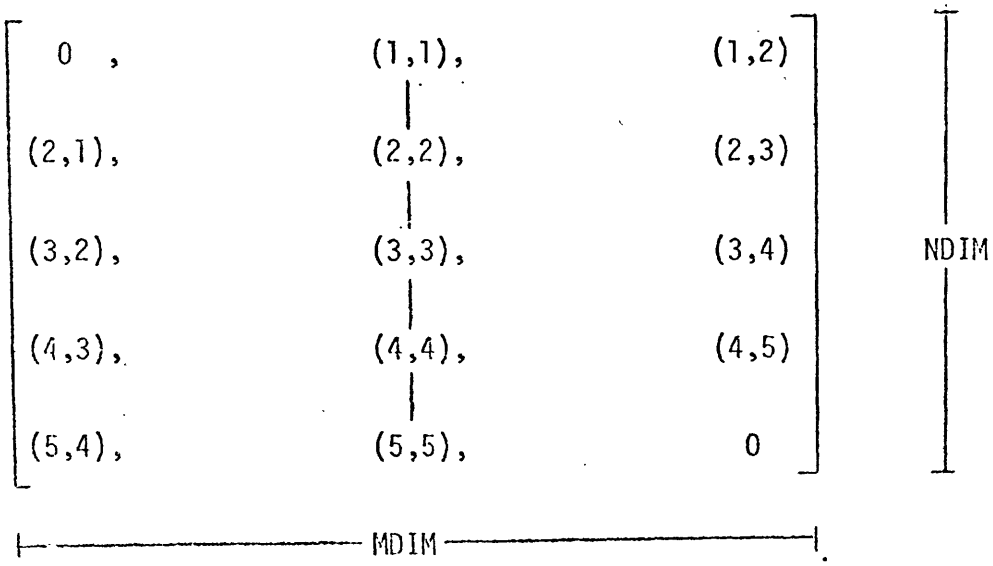
where i = 1,2, . . . , m, m being the specified number of iterations. If m = 1, this procedure is equivalent to the residual formulation outlined by Weinstein, Stone, and Kwan (1969).

In most of the two-dimensional, areal simulations studied it was not necessary to use Newton-Raphson iteration on the nonlinear coefficients in [B] and {q} to obtain satisfactory solutions. For radial problems and vertical cross-sections this is not the case as pointed out by Toronyi and Farouq Ali (1975). As stated earlier, Picard iteration is used for [B] and {q}. In this process the nonlinear coefficients are simply updated on each iteration.

The solution of the linearized matrix equation 33 is solved using the Gauss-Doolittle method for banded, nonsymmetric matrices. The coefficient matrix is decomposed into the product of upper and lower triangular matrices, from which the solution may be determined by back substitution. Because the coefficient matrix is nonlinear, this solution procedure is required for each iteration of each time step. The general form of the coefficient matrix and the form in which it is stored in the program is shown in figure 3, where MDIM is the matrix bandwidth and NDIM is the number of rows.



a. Normal storage of matrix where bold marks indicate the main diagonal.



b. Banded storage of matrix showing new location of the main diagonal.

Figure 3. Normal storage of matrix and banded storage of matrix.

Heat Loss Calculations

Geothermal reservoirs occur in areas of anomalously high heat flow. The effects of the high geothermal gradient and the loss and gain of heat to the base and cap rock are important factors that are incorporated into the reservoir model. In a typical hydrothermal system, heat flows into the reservoir at the base and out at the top.

In order to obtain the heat flux through the confining beds three simplifying assumptions are made. First, the permeability in the confining bed is assumed to be low and convective flow may be neglected; thus, only the heat conduction equation needs to be solved. Further, the horizontal conduction terms in the confining beds are assumed to be small relative to the vertical terms, and may be neglected. For petroleum reservoirs undergoing thermal recovery, it has been demonstrated that the effect of horizontal conduction in the confining beds is small (Chase and O'Dell, 1973). This assumption leads to the one-dimensional, heat-conduction equation:

$$K_r \frac{\partial^2 T'}{\partial z^2} = (\rho c)_r \frac{\partial T'}{\partial t} \quad ,$$

where the vertical thermal conductivity, density, and heat capacity of the confining bed are considered constant. The final assumption is that the geothermal reservoir, prior to exploitation, is at steady state; that is, the heat entering at the base equals that leaving at the top and the net heat gain of the reservoir is zero. This assumption allows consideration of only the "transient" heat flow caused by temporal temperature changes in the reservoir.

Therefore, the one-dimensional, conduction equation may be reformulated as follows:

$$K_r \frac{\partial^2 T^*}{\partial z^2} = (\rho c)_r \frac{\partial T^*}{\partial t} \quad , \tag{35}$$

where $T^* = T' - T'_0$ (T' is the temperature in the confining bed and T'_0 is the initial temperature in the confining bed.)

Equation 35 is subject to initial and boundary conditions. The initial conditions are simply $T^*(z,0) = 0$. For the boundary condition at the top of the cap rock, the temperature change is assumed to be zero for all time. At the cap rock-reservoir boundary, a step function in temperature is used. It is determined using the difference between the initial reservoir temperature and the reservoir temperature at the last time step (that is, the step function is lagged by one time step). Once the T^* distribution through the cap rock is determined, Fourier's equation is used to compute the heat flux.

A similar set of boundary and initial conditions apply to the base rock; however, instead of solving equation 35 for both the top and bottom, it is assumed that the two fluxes are approximately equal and therefore the total heat leakage is obtained by multiplying the flux computed at the top by two. This assumption is valid since only the "transient" heat leakage is considered; however, this portion of the program could be modified if needed (for example, if steady state thermal gradients above and below the reservoir are significantly different).

In the numerical model, equation 35 is actually solved at each grid block for each iteration. It is solved using a Galerkin, finite-element approximation for the space derivative combined with an implicit difference approximation for the time derivative. Linear elements are used with a variable mesh generator that divides the confining bed thickness into ten elements that double in size with distance from the reservoir boundary. Therefore, the element adjacent to the reservoir is relatively small, and it is the temperature difference across this element that is used in Fourier's equation to compute the heat flux.

Notes on Use of Program

(1) This version of the finite-difference model is restricted to problems involving confined, horizontal reservoirs exhibiting two-dimensional flow. Futhermore, the reservoir is overlain and underlain by impermeable layers that allow only conduction of heat.

(2) To minimize core requirements, the dimensions of the arrays A(NBB,MBE) and R(NBB) must be specified for each problem, where

$$NBB = 2*NB$$

(NB - number of nonzero blocks, that is, blocks that have nonzero permeability) and MBE is the estimated matrix bandwidth given for two simultaneous unknowns at each grid block by

$$MBE = 2 * (2 * MM + 1) + 1$$

where,

$$MM = NY \text{ IF } NX > NY$$
$$MM = NX \text{ IF } NY > NX$$

and NX and NY are the number of columns and rows, respectively.

(3) The actual matrix bandwidth is computed internally and printed. If it differs from the estimated bandwidth, change MBE so that it is equal to the actual matrix bandwidth. Also, change the dimensions of array A.

(4) The regression equations used in this program are based on steam table data for a temperature range of 10° to 300°C.

(5) The units of the input data must be in the cgs system.

(6) At present most arrays are dimensioned to solve problems with a maximum of 20 columns and 10 rows.

(7) Relative permeability functions described in equations 10 and 11 are programmed in SUBROUTINE PRPTY statement numbers PRP1230 - PRP1300 where the residual water saturation, S_{wr} = 0.3 and the residual steam saturation, S_{sr} = 0.05. Other equations may be substituted for these; the only restriction is that relative permeability must be a smooth function of saturation. A non-smooth relationship (such as linear interpolation between data points) can result in an oscillatory, unstable solution.

(8) The two-dimensional treatment in this model assumes that the fluid properties are uniform with depth. This assumption is probably valid only for very thin reservoirs, but may be a suitable approximation for some applications.

(9) The program should be in double precision, except when using a computer having single precision accuracy to 10 significant digits, in which case remove the REAL*8 IMPLICIT cards.

(10) Although the unknown dependent variables are pressure and enthalpy, the user is given the option to read in either initial pressures and temperatures (KØD9 = 1) or initial pressures and enthalpies (KØD9 = 0). This option is provided since field temperatures are more readily available than enthalpies. If temperatures are read, they are converted in the program to enthalpies, and subsequent calculations are made using the enthalpy values. If the initial conditions of the reservoir are two phase, the user must read in enthalpies.

(11) To reduce the number of lines in the output, only the computed pressures and enthalpies are printed for each time step. The user may, however, also have the computed water saturations, temperatures, and fluid densities printed as often as desired by specifying the proper value for IPRT. This parameter allows the additional data to be printed every IPRT time step. For example, if IPRT = 10, saturations, temperatures and densities will be printed on time step 1, 10, 20, . . . until the end of the simulation.

(12) Minimum number of blocks required for a successful run are two in the x-direction and two in the y-direction.

Description of the Subroutines

The FØRTRAN IV code contains a main program and 12 subroutines, which are shown diagrammatically in Figure 4. The purpose of each subroutine is listed below.

MAIN Driving program for the subroutines. In addition, the large arrays, A and R, are dimensioned to minimize core requirements.

GDATA Reads and writes problem information according to the formats listed in the INPUT section of this report.

READ Reads the two dimensional arrays containing data for each finite-difference block.

TCALC Computes the interblock transmissibility terms. Intrinsic permeability is determined as a harmonic mean of the values in the two blocks.

MATR Computes and prints matrix bandwidth. The estimated bandwidth should equal the computed bandwidth. The bandwidth is used in dimensioning arrays, and if the estimated and computed bandwidths are not equal, computational errors could result. See the section on 'Notes on Use of Program' for details on how to calculate the estimated bandwidth.

PRPTY Computes thermodynamic properties based on regression equations determined using data from steam tables.

VERTCD Computes vertical conductive heat leakage through a confining bed. This is accomplished by solving the one-dimensional, heat-conduction equation at each finite-difference grid block. The numerical method used in this subroutine is the finite-element method.

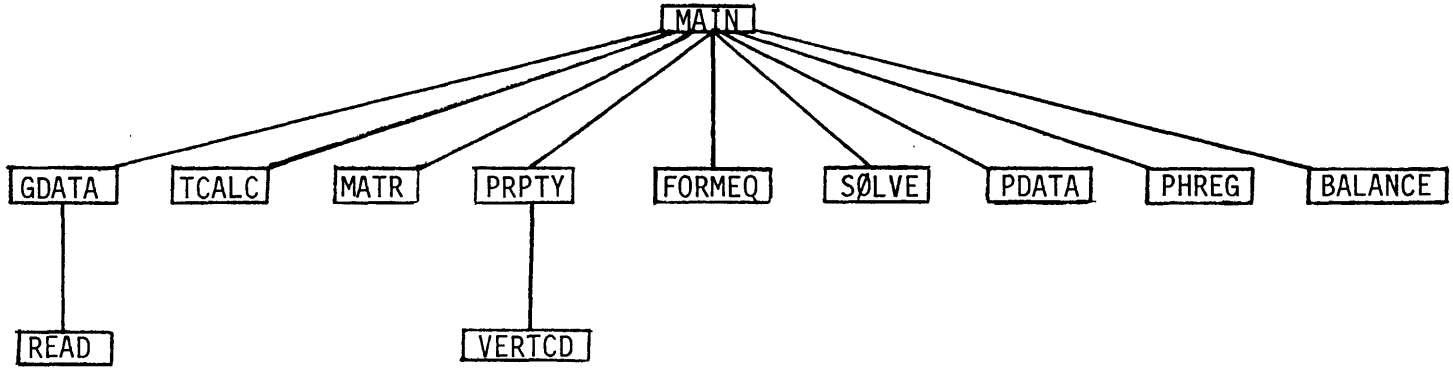


Figure 4. Program subroutines showing order and link of calling.

FØRMEQ Forms the final matrix equation. Note that the matrix is nonsymmetric and that the dependent variables, pressure and enthalpy are solved simultaneously.

SØLVE Solves the matrix equation using the Gauss-Doolittle method. It triangularizes a banded nonsymmetric matrix and then back substitutes.

PDATA Prints the computed pressures and enthalpies for each time step.

PHREG Determines the thermodynamic region for each finite-difference grid block.

BALNCE Computes a mass and energy balance for each time step.

A generalized flow chart showing the approximate order that the subroutines are used is shown in Figure 5.

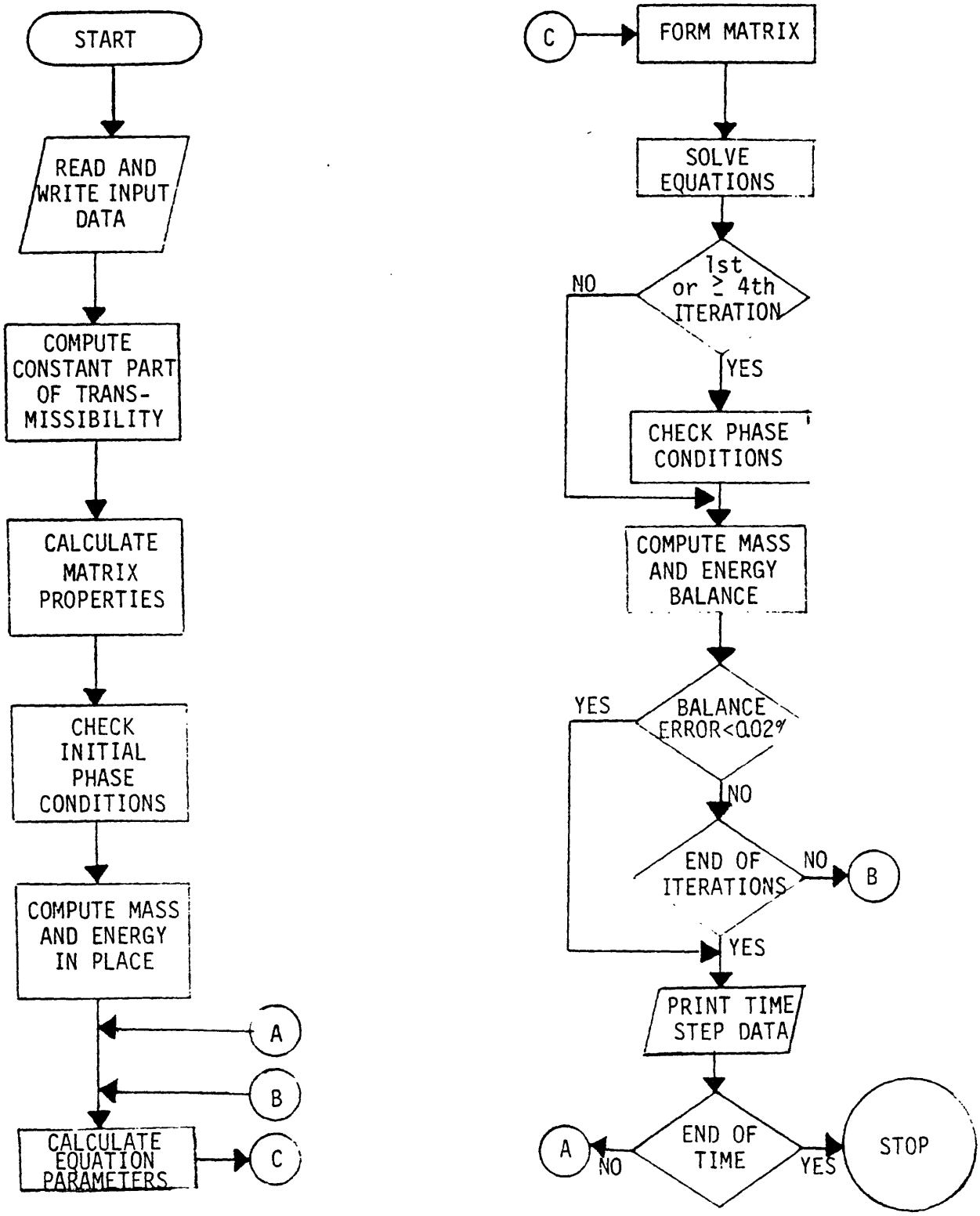


Figure 5. Generalized flow chart.

Input

Columns	Format	Name	Description
Card 1			
1-80	20A4	TITLE	Problem title
Card 2			
1-5	I5	NX	Number of columns (x-direction)
6-10	I5	NY	Number of rows (y-direction)
11-15	I5	NB	Number of non-zero blocks (those with non-zero permeability)
16-20	I5	NK	Maximum number of Newton iterations
21-25	I5	NT	Maximum number of time steps
26-30	I5	NS	Number of sources
31-35	I5	MBE	Estimated bandwidth (see text)
36-40	I5	IØPT	Read 1 for upstream weighting; 2 for midpoint weighting (See section on Finite-Difference Representation)
41-45	I5	PPRT	Number of time steps between printing thermodynamic data
Card 3			
1-10	G10.0	DELT	Time step (in seconds)

Columns	Format	Name	Description
Card 4			
1-5	I5	KØD1	Read 1 if x-spacing is constant (otherwise, leave blank)
6-10	I5	KØD2	Read 1 if y-spacing is constant
11-15	I5	KØD3	Read 1 if initial pressure is constant
16-20	I5	KØD4	Read 1 if initial enthalpy is constant
21-25	I5	KØD5	Read 1 if x-permeability is constant
26-30	I5	KØD6	Read 1 if y-permeability is constant
31-35	I5	KØD7	Read 1 if initial porosity is constant
36-40	I5	KØD8	Read 1 if reservoir thickness is constant
41-45	I5	KØD9	Read 1 if temperature is read in place of enthalpy
		Data Set 1 - X-spacing	
1-80	8G10.0	DX(I)	Spacing in the x-direction (NX values); if constant, KØD1 = 1 and only read one value (cm)
		Data Set 2 - Y-spacing	
1-80	8G10.0	DY(J)	Spacing in the y-direction (NY values); if constant, KØD2 = 1 and only read one value (cm)
		*Data Set 3 - Initial pressure**	
1-80	8G10.0	P(I,J)	Initial pressure distribution in the reservoir; if constant, KOD3 = 1 and only read one value (dynes/cm^2)

*Start new card for the beginning of each new row (may start with the top row or bottom row, but be consistent) leaving blanks for missing blocks.

Columns	Format	Name	Description
		*Data Set 4 - Initial enthalpy *+*	
1-80	8G10.0	H(I,J)	Initial enthalpy distribution in the reservoir; if constant, KØD4 = 1 and only read one value (ergs/g)
		*Data Set 5 - X-permeability**	
1-80	8G10.0	XK(I,J)	Reservoir permeability in x-direction; if constant, KØD5 = 1 and only read one value (cm^2)
		*Data Set 6 - Y-permeability**	
1-80	8G10.0	YK(I,J)	Reservoir permeability in y-direction; if constant, KØD6 = 1 and only read one value (cm^2)
		*Data Set 7 - Porosity**	
1-80	8G10.0	PHI(I,J)	Reservoir porosity; if constant, KØD7 = 1 and only read one value (dimensionless)
		*Data Set 8 - Thickness**	
1-80	8G10.0	DZ(I,J)	Reservoir thickness; if constant, KØD8 = 1 and only read one value (cm)

* Start new card for the beginning of each new row (may start with the top row or bottom row, but be consistent) leaving blanks for missing blocks.

+ If KØD9 = 1, read initial temperature distribution (°C) instead.

Columns	Format	Name	Description
		Data Set 9 - Sources	
1-5	I5	I	Column number of well
6-10	I5	J	Row number of well
11-25	G15.0	Q(I,J)	Strength (g/s) of source/sink at block i,j

Note: 1-NS cards; if NS = 0 this data set is omitted.

Card 5

1-10	G10.0	XKC	Medium thermal conductivity of the reservoir (ergs/s-cm°C)
11-20	G10.0	CØND	Confining bed thermal conductivity (ergs/s-cm°C)
21-30	G10.0	PHFWT	Rock enthalpy derivative with respect to temperature (specific heat) (ergs/g°C)
31-40	G10.0	DF	Rock density (g/cm^3)
41-50	G10.0	BETA	Compressibility of reservoir (cm^2/dyne)

Output

To aid the user in detecting errors associated with data input, data that is read in is immediately printed; thus output appears in the following order:

Title of Problem

Finite-Difference Data

Grid Numbers

Time Parameters

Codes

Spacing in X-Direction

Spacing in Y-Direction

Initial Pressure

Initial Enthalpy

X-Permeability

Y-Permeability

Initial Porosity

Reservoir Thickness

Sources

Rock Properties

Maximum Bandwidth

In addition, every IPRT time step the following is also printed:

Water Saturations
Temperatures
Density

} Printed at beginning of time step, but are based on pressure and enthalpy from previous time step.

Finally, on a successful run, the following is printed every time step:

Step Number

Time

Pressure Values

Enthalpy Values

Mass and Energy Balance

APPLICATIONS

Example 1

The linear flow of hot incompressible fluid through a confined aquifer may be described by the following equations:

$$K_r \frac{\partial^2 u}{\partial z^2} = \rho_r c_r \frac{\partial u}{\partial t}, \quad z > 0, \; t > 0 \tag{36a}$$

$$K_t \frac{\partial^2 u}{\partial x^2} - v_w \rho_w c_w \frac{\partial u}{\partial x} + \frac{2}{b} K_r \frac{\partial u}{\partial z} = \rho_t c_t \frac{\partial u}{\partial t}, \quad z = 0, \; x > 0, \; t > 0 \tag{36b}$$

subject to:

$$u(x,0) = 0, \; x > 0$$

$$u(0,t) = 1, \; z = 0, \; t \geq 0$$

$$\lim_{x^2 + z^2 \to \infty} u = 0$$

where $u = \frac{T-T_o}{T_1-T_o}$ is the normalized temperature; T_o is the overburden (underburden) temperature and the initial aquifer temperature; and T_1 is the temperature of the injection fluid. The subscripts refer to the rock, r, water, w, and total (rock and water), t.

Avdonin (1964) presents an analytical solution for equation (36):

$$u(\chi,\tau) = \frac{\chi}{\sqrt{\pi\tau}} \int_0^1 \left\{ \exp\left[- (s\gamma\sqrt{\tau} - \frac{\chi}{2s\sqrt{\tau}})^2 \right] \operatorname{erfc} \left(\frac{\alpha s^2\sqrt{\tau}}{2\sqrt{1-s^2}} \right) \right\} \frac{ds}{s^2} \cdots$$

where $\chi = \frac{2x}{b}$; $\tau = \frac{4K_t t}{c_t \rho_t b^2}$; $\gamma = \frac{Qc_w\rho_w}{4K_t}$;

$\alpha = \sqrt{\frac{K_r c_r \rho_r}{K_t c_t \rho_t}}$; and Q is the injection flow rate. (In the original reference, γ is defined with an 8 in the denominator; this appears to be a typographical error.)

To simulate this problem, the following assumptions were made:

1) Temperature is a function of enthalpy only, according to:

$$T = - 0.0208 + 2.39 \times 10^{-8} h \qquad (38)$$

2) Density is a function of pressure only, according to:

$$\rho = 0.989875 + 4.00894 \times 10^{11} p \qquad (39)$$

3) Porosity is constant.

These changes must be made in the program by appropriately changing the single-phase (water) statement functions for temperature and density, and by appropriately changing their respective pressure- and enthalpy-derivatives. For constant porosity, beta is read as zero. Other parameters and the initial conditions used in this example are given in table 1. A block-centered grid consisting of 20 blocks was used, and the time step was 6.25×10^6 sec. The total simulation time was 3.75×10^8 sec.

The results are presented in figure 6. This problem was chosen because it exhibits pronounced truncation error in the approximation of the spatial derivatives. Time steps were chosen to reduce time truncation errors. Both mid-point weighting and upstream weighting were used. It is clear that mid-point weighting approximates the temperature front better than upstream weighting, but exhibits oscillations at the base of the front. Upstream weighting smears the front out by numerical diffusion and does not exhibit oscillations.

TABLE 1 - DATA FOR AVDONIN LINEAR EXAMPLE

PARAMETER	SYMBOL	VALUE
velocity	v	1.28×10^{4} cm/sec
porosity	ϕ	0.20
reservoir thermal conductivity	K_m	3.20×10^{6} ergs/sec-cm ^{o}C
confining bed thermal conductivity	K'	3.20×10^{5} ergs/sec-cm ^{o}C
rock density	ρ_r	2.50 g/cm^{3}
rock specific heat	c_r	1.01×10^{7} ergs/g ^{o}C
aquifer thickness	b	2.00×10^{4} cm
initial pressure	p_i	1.38×10^{7} dynes/cm2
initial enthalpy	h_i	3.35×10^{9} ergs/g
initial fluid viscosity	μ_i	3.58×10^{-3} g/cm-sec
initial temperature	T_i	80.19 ^{o}C
injection temperature	T'	40.01 ^{o}C
fluid density	ρ	0.99 g/cm^{3}
fluid specific heat	c_w	4.18×10^{7} ergs/g^{o}C

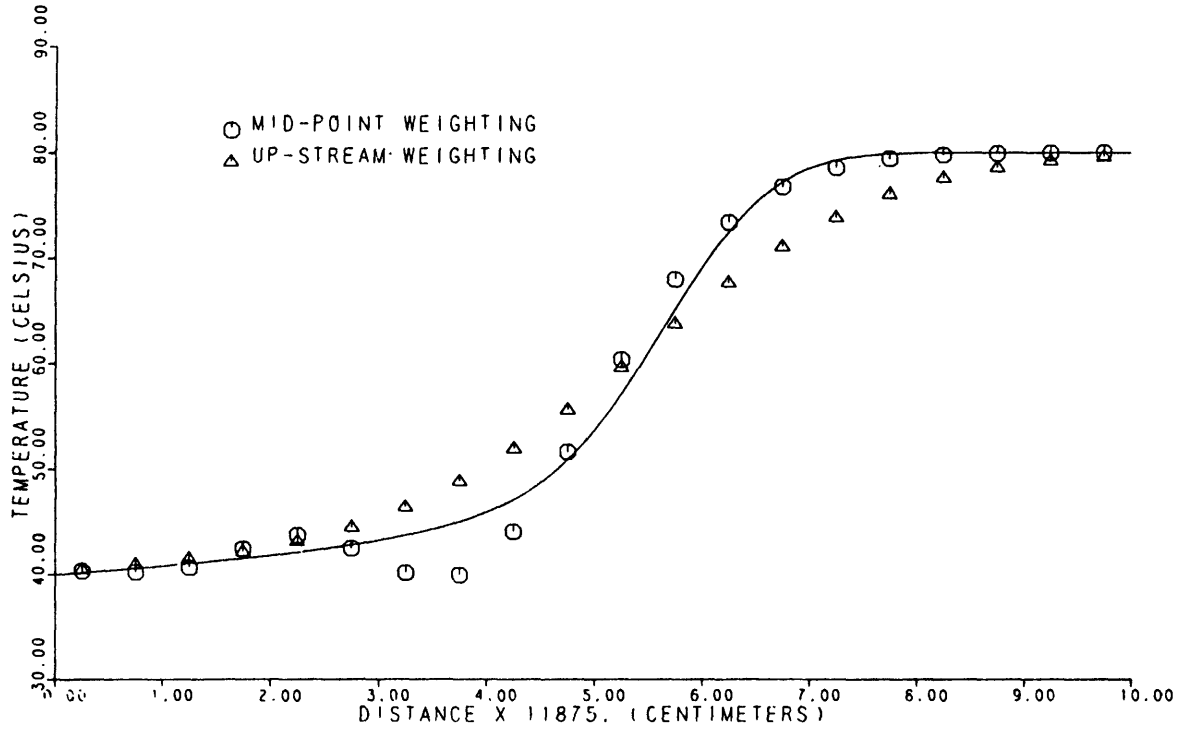


Figure 6. Finite-difference solution for the Avdonin linear example at time = 3.75×10^8 sec; analytical solution indicated by solid line.

Example 2

This example was designed to demonstrate the behavior of an initially hot-water geothermal reservoir that develops a two-phase zone under the influence of production. This problem was originally solved using a finite-element model and a five year simulation period (Faust and Mercer, 1975) and was rerun later under the same conditions using a finite-difference model (Faust and Mercer, 1976). The finite-difference simulation in this report is the same as that presented in the above references, however this simulation stops after 104 days.

Data for this example is given in table 2. This is a one-dimensional problem and therefore only one row is used in conjunction with ten columns, giving ten non-zero blocks. It is assumed that the reservoir is closed to the flow of heat and mass at all boundaries. The time step is 5.0×10^5 seconds and the maximum number of time steps is 18, with a maximum of twelve iterations. The estimated bandwidth is seven, and the spacings in the x- and y-directions are 2.0×10^4 cm and 5.0×10^4 cm respectively. Upstream weighting is used and there is only one sink located (at the left end) in the first block (1,1). A listing of the formated model data is given in figure 7 Finally, the relative permeabilities used for this problem are:

$$k_{rw} = [(S_w - 0.35)/0.65]^4 \tag{40}$$

and

$$k_{rs} = \{1.0 - [(S_w - 0.35)/0.65]^2\}* \{1.0 - [(S_w - 0.35)/0.65]\}^2 \ . \tag{41}$$

TABLE 2 - DATA FOR EXAMPLE WITH CONVERSION

porosity	ϕ	0.10
reservoir thermal conductivity	K_m	3.20×10^5 ergs/sec-cm°C
rock density	ρ_r	2.50 g/cm^3
rock specific heat	c_r	1.01×10^7 ergs/g°C
aquifer thickness	b	5.00×10^4 cm
initial pressure	p_i	4.38×10^7 dynes/cm^2
initial enthalpy	h_i	1.02×10^{10} ergs/g
initial temperature	T_i	236.12°C
initial density	ρ_i	0.82 g/cm^3
permeability	k	1.00×10^{-10} cm^2
discharge rate	Q	2.00×10^4 g/sec

Figure 7. Formated Data for Example 2.

```
 EXAMPLE RUN WITH CONVERSION
  10     1    10    12    18     1     7     1     1
500000.
  1     1     1     1     1     1     1     1
20000.
50000.
  .438D8
1.024D10
  0.1D-9
  0.1D-9
0.1
50000.
  1     1   -20000.
   3.205    0.00000     1.0107    2.5           0.0D0
```

These relationships may be found in the program at statement numbers PRP1230 - PRP1300. For problems with different functions, the user must change these statements. Note that experience has shown that these functions must be smooth and continuous or numerical oscillations may result.

At the end of 17 time steps (8.5×10^6 sec) the finite-difference model predicts that the water saturation in the first block is 0.96994; all other blocks are at a saturation of 1.0. For illustrative purposes results from the finite-element and finite-difference models (Faust and Mercer, 1976) for the five year simulation are compared in figures 8 and 9. In figure 8, pressure is plotted versus distance (origin is at the left end of the reservoir), for various simulation times. The specified times represent the duration of exploitation. In this example, the pressure drops rapidly in the early stages of production, and at the end of one month the first block has become two phase. When this occurs, the pressure continues to drop rapidly in the rest of the reservoir, but changes only slightly in the block containing the sink. This is expected, because once a block becomes two-phase, the pressure is maintained by the formation of steam. Also, note that once the system becomes two phase, the time step may be increased without causing any numerical problems. After 3 years of exploitation, the pressures in the entire reservoir have lowered to a point just slightly above the saturated-vapor pressure. Subsequent mass extraction results in reduced water saturations in elements near the sink, and pressures drop very slowly.

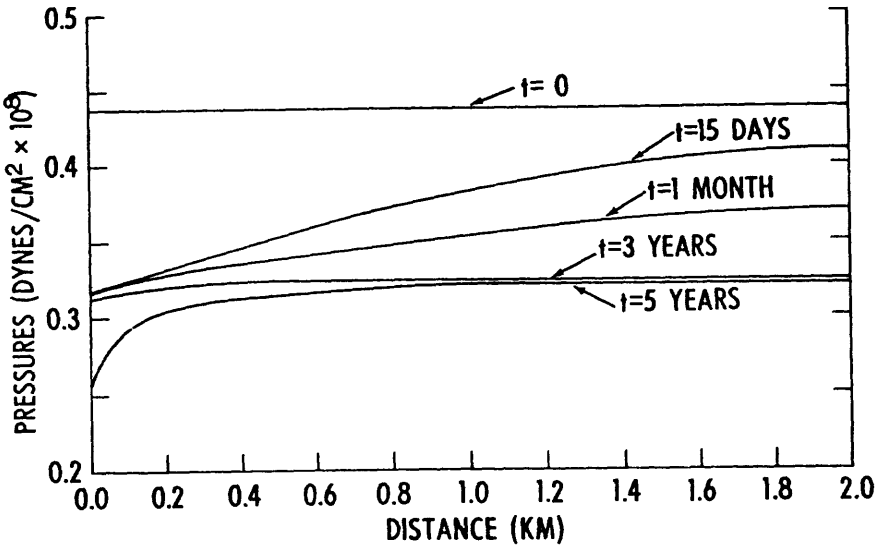


Figure 8.- Computed pressure distribution at various times for example 2. (After Faust and Mercer, 1976.)

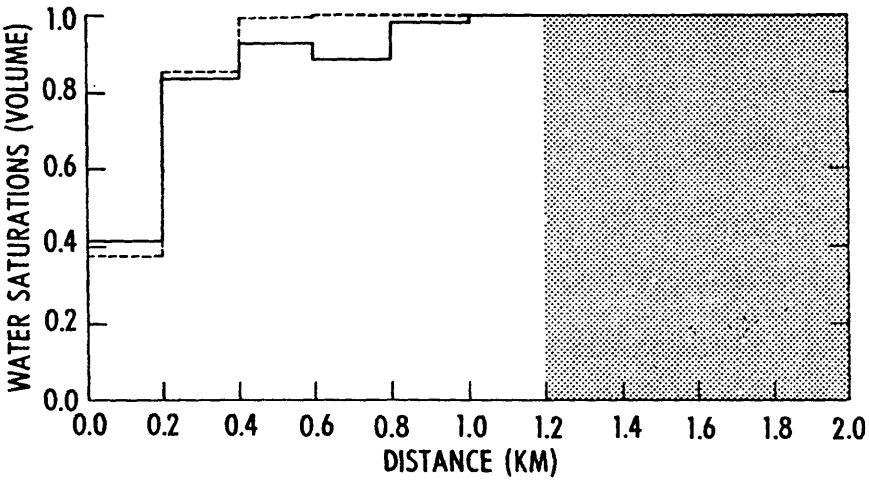


Figure 9. Computed water saturation distribution for example 2 after 5 years of exploitation. (Single-phase water indicated by shading; solid line: Galerkin finite-element solution; dashed line: finite-difference solution.) After Faust and Mercer, 1976.

Water saturation distributions at the end of 5 years of exploitation predicted by both the Galerkin- and finite-difference models are shown in figure 9. Note that the finite-element solution exhibits an oscillation in the saturation distribution. The upstream-weighting scheme used in the finite-difference model eliminates this oscillation.

NOTATION

Parameter	Coded Name	Description
b	DZ	reservoir thickness
c_r	PHFWT	rock specific heat
h	H	enthalpy
h_r	HROCK	rock enthalpy
h_s	HS	steam enthalpy
h_w	HW	water enthalpy
K_m	XKC	medium thermal conductivity
K_r	CØND	confining bed thermal conductivity
k	XK	x- permeability
	YK	y- permeability
k_r	XKW	water relative permeability
	XKS	steam relative permeability
p	P	pressure
q_m^*	Q	mass source term
q_h^*	QH	energy source term
q^{ll}	CQ	vertical conductive energy source term
S_w	SW1	water volume saturation
T	TEMP2	temperature
t	TIME	simulation time
β	BETA	rock compressibility
μ_s	VS	steam viscosity

Parameter	Coded Name	Description
μ_w	VW	water viscosity
ρ	DEN2	density
ρ_r	DF	rock density
ρ_s	DS	steam density
ρ_w	DW	water density
ϕ	PHI	porosity

REFERENCES CITED

Avdonin, N. A., 1964, Some formulas for calculating the temperature field of a stratum subject to thermal injection: Neft' i Gaz, v. 3, p. 37-41.

Chase, C. A., and O'Dell, P. M., 1973, Application of variational principles to cap and base rock heat losses: Soc. Petroleum Engineers Jour., v. 13. p. 200-210.

Corey, A. T., 1954, The interrelation between gas and oil relative permeabilities: Producers Monthly, v. 19, p. 38-41.

Faust, C. R., 1976, Numerical simulation of fluid flow and energy transport in liquid- and vapor-dominated hydrothermal systems: University Park, Pa., Pennsylvania State Univ., Ph.D. thesis, 163 p.

Faust, C. R., and Mercer, J. W., 1975, Mathematical modeling of geothermal systems, in Proceedings of the Second United Nations Symposium on the development and use of geothermal resources, San Francisco, Calif., May 20-29, 1975: United Nations, v. 3, p. 1635-1642.

Faust, C. R., and Mercer, J. W., 1976, An analysis of finite-difference and finite-element techniques for geothermal reservoir simulation, in Numerical simulation of reservoir performance, Society of Petroleum Engineers of AIME, 4th Symposium, Los Angeles, Calif., Feb. 19-20, 1976, Proc.: Soc. Petroleum Engineers, Paper SPE-5742, p. 337-354.

Keenan, J. H., Keys, F. G., Hill, P. G., and Moore, J. G., 1969, Steam tables: London, John Wiley & Son, Inc., 162 p.

Mercer, J. W., Faust, C. R., and Pinder, G. F., 1974, Geothermal reservoir simulation, in Conference on research for the development of geothermal energy resources, sponsored by National Science Foundation, Pasadena, Calif., Sept. 23-25, 1974, Proc.: Natl. Sci. Found. Rept. RA-N-74-159, p. 256-267.

Meyer, C. A., McClintock, R. B., Silvestri, G. J., and Spencer, R. C., 1967, ASME Steam Tables: New York, American Society of Mechanical Engineers, 2d ed., 328 p.

Toronyi, R. M., and Farouq Ali, S. M., 1975, Two-phase, two-dimensional simulation of a geothermal reservoir and the wellbore system, in Society of Petroleum Engineers of AIME, 50th Ann. Fall Mtg., Dallas, Tex., Sept. 28-Oct. 1, 1975, Proc.: Soc. Petroleum Engineers, Paper SPE-5521, 26 p.

Weinstein, H. G., Stone, J. L., and Kwan, T. V., 1969, Iterative procedure for solution of systems of parabolic and elliptic equations in three dimensions: I and EC Fund, v. 2, no. 2, p. 281-287.

White, D. E., Muffler, L. P. J., and Truesdell, A. H., 1971, Vapor-dominated hydrothermal systems compared with hot-water systems: Econ. Geology, v. 66, p. 75-97.

(page 55 follows)

PROGRAM LISTING

(page 57 follows)

```
C     MAIN PROGRAM                                                      MNP  10
C     ************                                                      MNP  20
      IMPLICIT REAL*8(A-H,O-Z)                                          MNP  30
C                                                                       MNP  40
C     PURPOSE:  TO SIMULATE VAPOR- AND LIQUID-DOMINATED GEOTHERMAL      MNP  50
C               RESERVOIRS USING FINITE DIFFERENCE TECHNIQUES           MNP  60
C     PROGRAMED BY CHARLES R. FAUST AND JAMES W. MERCER                 MNP  70
C                  U. S. GEOLOGICAL SURVEY                              MNP  80
C                  1975-1976                                            MNP  90
C     ----------------------------------------------------------------MNP 100
C                                                                       MNP 110
C     TO MINIMIZE CORE, DIMENSION A AND R FOR EACH PROBLEM:             MNP 120
C     A(NBB,MBE), R(NBB)                                                MNP 130
      DIMENSION A(300,43), R(300)                                       MNP 140
C                                                                       MNP 150
      COMMON /INPUT/ PHI(20,10),XK(20,10),YK(20,10),P(20,10),H(20,10),X(MNP 160
     1400),X1(400),NP(20,10),NPP(200,2),DX(20),DY(10),DZ(20,10),Q(20,10)MNP 170
      COMMON /CONTRO/ NK,NX,NXX,NY,NYY,NB,NBB,NT,DELT,TIME,PHFAT,DF,XKC,MNP 180
     1COND,COEF,BETA,IPRT                                               MNP 190
      COMMON /WORK/ AX(21,10),BX(21,10),EX(21,10),AY(20,11),BY(20,11),EYMNP 200
     1(20,11),TX(21,10),TXK(21,10),TY(20,11),TYK(20,11),XM(20,10),XMASS(MNP 210
     220,10),EN(20,10),ENERGY(20,10),C(20,10),D(20,10),F(20,10),G(20,10)MNP 220
     3,QH(20,10),XN(20,10),IM(20,10),DTP(20,10),DTH(20,10),PHIC(20,10),PMNP 230
     40LD(20,10)                                                        MNP 240
      COMMON /EXTRA/ MBW,MBE                                            MNP 250
      COMMON /CHECK/ IND(200),INDOLD(200)                               MNP 260
C     ----------------------------------------------------------------MNP 270
C                                                                       MNP 280
C     GET DATA -                                                        MNP 290
      CALL GDATA(IOPT)                                                  MNP 300
C     ****************                                                  MNP 310
      NKK=NK                                                            MNP 320
C                                                                       MNP 330
C     FOR DIMENSIONING -                                                MNP 340
      NBBD=NBB                                                          MNP 350
      MBED=MBE                                                          MNP 360
C                                                                       MNP 370
C     CALCULATE TRANSMISSIBILITY TERMS -                                MNP 380
      CALL TCALC(DY,DX,DZ,XK,YK,TY,TYK,TX,TXK,XKC,NY,NX,NYY,NXX)        MNP 390
C     ****************************************************************MNP 400
C                                                                       MNP 410
C     CALCULATE BANDWIDTH -                                             MNP 420
      CALL MATR(NP,NY,NX,MBE,MBW)                                       MNP 430
C     ***************************                                       MNP 440
C                                                                       MNP 450
C     CHECK INITIAL PHASE CONDITIONS                                    MNP 460
C     COMPUTE THERMODYNAMIC REGION -                                    MNP 470
      CALL PHREG(IND)                                                   MNP 480
C     ***************                                                   MNP 490
      DO 10 K=1,NB                                                      MNP 500
   10 INDOLD(K)=IND(K)                                                  MNP 510
C                                                                       MNP 520
```

```
C     CALCULATE INITIAL BALANCE -                                        MNP 530
      CALL BALNCE(0,NKK,PCEE)                                            MNP 540
C     **********************                                             MNP 550
C                                                                        MNP 560
C                                                                        MNP 570
C  TIME LOOP -                                                           MNP 580
      DO 180 L=1,NT                                                      MNP 590
C                                                                        MNP 600
C  ITERATION LOOP -                                                      MNP 610
      DO 130 KKK=1,NK                                                    MNP 620
C                                                                        MNP 630
C     CALCULATE PROPERTY COEFFICIENTS -                                  MNP 640
      CALL PRPTY(L,KKK)                                                  MNP 650
C     *****************                                                  MNP 660
C                                                                        MNP 670
C     INITIALIZE BOUNDARY ARRAYS -                                       MNP 680
      DO 20 I=1,NX                                                       MNP 690
      AY(I,1)=0.D0                                                       MNP 700
      BY(I,1)=0.D0                                                       MNP 710
      EY(I,1)=0.D0                                                       MNP 720
      AY(I,NYY)=0.D0                                                     MNP 730
      BY(I,NYY)=0.D0                                                     MNP 740
   20 EY(I,NYY)=0.D0                                                     MNP 750
C                                                                        MNP 760
      DO 30 J=1,NY                                                       MNP 770
      AX(1,J)=0.D0                                                       MNP 780
      BX(1,J)=0.D0                                                       MNP 790
      EX(1,J)=0.D0                                                       MNP 800
      AX(NXX,J)=0.D0                                                     MNP 810
      BX(NXX,J)=0.D0                                                     MNP 820
   30 EX(NXX,J)=0.D0                                                     MNP 830
C                                                                        MNP 840
      DO 50 J=1,NY                                                       MNP 850
      DO 50 I=2,NX                                                       MNP 860
C     FOR UPSTREAM WEIGHTING -                                           MNP 870
      ALPHA=1.D0                                                         MNP 880
      IF (P(I-1,J).GT.P(I,J)) ALPHA=0.D0                                 MNP 890
      IF (IOPT.EQ.1) GO TO 40                                            MNP 900
C     FOR MIDPOINT WEIGHTING -                                           MNP 910
      ALPHA=DX(I)/(DX(I)+DX(I-1))                                        MNP 920
   40 EX(I,J)=(ALPHA*XN(I,J)+(1.0-ALPHA)*XN(I-1,J))*TX(I,J)              MNP 930
      AX(I,J)=(ALPHA*DTP(I,J)+(1.0-ALPHA)*DTP(I-1,J))*TXK(I,J)+(ALPHA*TMMNP 940
     1(I,J)+(1.0-ALPHA)*TM(I-1,J))*TX(I,J)                               MNP 950
   50 BX(I,J)=(ALPHA*DTH(I,J)+(1.0-ALPHA)*DTH(I-1,J))*TXK(I,J)           MNP 960
C                                                                        MNP 970
      DO 70 I=1,NX                                                       MNP 980
      DO 70 J=2,NY                                                       MNP 990
      ALPHA=1.D0                                                         MNP1000
      IF (P(I,J-1).GT.P(I,J)) ALPHA=0.D0                                 MNP1010
      IF (IOPT.EQ.1) GO TO 60                                            MNP1020
      ALPHA=DY(J)/(DY(J)+DY(J-1))                                        MNP1030
   60 EY(I,J)=(ALPHA*XN(I,J)+(1.0-ALPHA)*XN(I,J-1))*TY(I,J)              MNP1040
```

```
      AY(I,J)=(ALPHA*DTP(I,J)+(1.0-ALPHA)*DTP(I,J-1))*TYK(I,J)+(ALPHA*TMMNP1050
     1(I,J)+(1.0-ALPHA)*TM(I,J-1))*TY(I,J)                              MNP1060
   70 BY(I,J)=(ALPHA*DTH(I,J)+(1.0-ALPHA)*DTH(I,J-1))*TYK(I,J)          MNP1070
C                                                                       MNP1080
C     FORM FINAL MATRIX EQUATION -                                      MNP1090
      CALL FORMEQ(A,R,NBBD,MBED)                                        MNP1100
C     ***************************                                       MNP1110
      IHALF=(MBW-1)/2                                                   MNP1120
C                                                                       MNP1130
C     TRIANGULARIZE -                                                   MNP1140
      CALL SOLVE(1,A,R,NBB,IHALF,NBBD,MBED)                             MNP1150
C     ************************************                              MNP1160
C                                                                       MNP1170
C     SOLVE -                                                           MNP1180
      CALL SOLVE(2,A,R,NBB,IHALF,NBBD,MBED)                             MNP1190
C     ************************************                              MNP1200
      DO 80 K=1,NB                                                      MNP1210
      X1(2*K-1)=X1(2*K-1)+R(2*K-1)                                      MNP1220
   80 X1(2*K)=X1(2*K)+R(2*K)                                            MNP1230
      IF (KKK.NE.1) GO TO 110                                           MNP1240
   85 CONTINUE                                                          MNP1245
      CALL PHREG(IND)                                                   MNP1250
C     ***************                                                   MNP1260
C                                                                       MNP1270
C     DID A CONVERSION OCCURE? -                                        MNP1280
      DO 100 K=1,NB                                                     MNP1290
```

```
  100 INDOLD(K)=IND(K)                                                  MNP1450
      GO TO 105                                                         MNP1455
  110 CONTINUE                                                          MNP1460
      IF (KKK.GE.4) GO TO 85                                            MNP1465
  105 CONTINUE                                                          MNP1467
      DO 120 K=1,NB                                                     MNP1470
      I=NPP(K,1)                                                        MNP1480
      J=NPP(K,2)                                                        MNP1490
      P(I,J)=X1(2*K-1)                                                  MNP1500
  120 H(I,J)=X1(2*K)                                                    MNP1510
      WRITE (6,190) KKK                                                 MNP1520
```

```
C                                                                       MNP1530
C     CALCULATE MASS AND HEAT BALANCE -                                 MNP1540
      CALL BALNCE(1,KKK,PCEE)                                           MNP1550
C     ************************                                          MNP1560
      ERR=DABS(PCEE)                                                    MNP1570
C     CHECK ERROR OF ENERGY BALANCE TO DETERMINE NUMBER OF ITERATIONS - MNP1580
      IF (ERR.LT.0.02) GO TO 140                                        MNP1590
  130 CONTINUE                                                          MNP1600
      GO TO 160                                                         MNP1610
  140 CONTINUE                                                          MNP1620
      DO 150 I=1,NX                                                     MNP1630
      DO 150 J=1,NY                                                     MNP1640
      XMASS(I,J)=XM(I,J)                                                MNP1650
  150 ENERGY(I,J)=EN(I,J)                                               MNP1660
  160 CONTINUE                                                          MNP1670
C                                                                       MNP1680
C  END ITERATION LOOP                                                   MNP1690
      DO 170 K=1,NBB                                                    MNP1700
  170 X(K)=X1(K)                                                        MNP1710
      TIME=TIME+DELT                                                    MNP1720
C                                                                       MNP1730
C     PRINT RESULTS -                                                   MNP1740
      CALL PDATA(P,H,TIME,L,NY,NX)                                      MNP1750
C     ***************************                                       MNP1760
  180 CONTINUE                                                          MNP1770
C                                                                       MNP1780
C  END TIME LOOP                                                        MNP1790
      STOP                                                              MNP1800
C                                                                       MNP1810
  190 FORMAT (//15X,16HITERATION NUMBER,I5)                             MNP1820
      END                                                               MNP1830-
```

```
      SUBROUTINE GDATA(IOPT)                                                DAT  10
C     **********************                                                DAT  20
      IMPLICIT REAL*8(A-H,O-Z)                                              DAT  30
C                                                                           DAT  40
C     CALLED FORM MAIN                                                      DAT  50
C     PURPOSE:  TO READ AND WRITE PROBLEM INFORMATION                       DAT  60
C     ------------------------------------------------------------------DAT  70
      DIMENSION TITLE(20)                                                   DAT  80
C                                                                           DAT  90
      COMMON /INPUT/ PHI(20,10),XK(20,10),YK(20,10),P(20,10),H(20,10),X(DAT 100
     1400),X1(400),NP(20,10),NPP(200,2),DX(20),DY(10),DZ(20,10),Q(20,10)DAT 110
      COMMON /CONTRO/ NK,NX,NXX,NY,NYY,NB,NBB,NT,DELT,TIME,PHFWT,DF,XKC,DAT 120
     1COND,COEF,BETA,IPRT                                                 DAT 130
      COMMON /WORK/ AX(21,10),BX(21,10),EX(21,10),AY(20,11),BY(20,11),EYDAT 140
     1(20,11),TX(21,10),TXK(21,10),TY(20,11),TYK(20,11),XM(20,10),XMASS(DAT 150
     220,10),EN(20,10),ENERGY(20,10),C(20,10),D(20,10),F(20,10),G(20,10)DAT 160
     3,QH(20,10),XN(20,10),TM(20,10),DTP(20,10),DTH(20,10),PHIC(20,10),PDAT 170
     4OLD(20,10)                                                          DAT 180
      COMMON /EXTRA/ MBW,MBE                                                DAT 190
C     ------------------------------------------------------------------DAT 200
C                                                                           DAT 210
C     READ AND WRITE UNITS -                                                DAT 220
      NR=5                                                                  DAT 230
      NW=6                                                                  DAT 240
C                                                                           DAT 250
C     INITIALIZE SIMULATION TIME -                                          DAT 260
      TIME=0.D0                                                             DAT 270
C                                                                           DAT 280
C     READ AND WRITE TITLE OF PROBLEM -                                     DAT 290
      READ (NR,280) TITLE                                                   DAT 300
      WRITE (NW,290)                                                        DAT 310
      WRITE (NW,300) TITLE                                                  DAT 320
C                                                                           DAT 330
C     READ AND WRITE FINITE DIFFERENCE INFORMATION -                        DAT 340
      READ (NR,310) NX,NY,NB,NK,NT,NS,MBE,IOPT,IPRT                        DAT 350
C     NX - NUMBER OF COLUMNS (X-DIRECTION)                                  DAT 360
C     NY - NUMBER OF ROWS (Y-DIRECTION)                                     DAT 370
C     NB - NUMBER OF NON-ZERO BLOCKS                                        DAT 380
C     NK - MAXIMUM NUMBER OF NEWTON ITERATIONS                              DAT 390
C     NT - MAXIMUM NUMBER OF TIME STEPS                                     DAT 400
C     NS - NUMBER OF SOURCES                                                DAT 410
C     MBE - ESTIMATED BANDWIDTH                                             DAT 420
C     IOPT - READ 1 FOR UPSTREAM WEIGHTING ON PRESSURE;                     DAT 430
C                 2 FOR MIDPOINT WEIGHTING                                  DAT 440
C     IPRT - NUMBER OF TIME STEPS BETWEEN PRINTING THERMO DATA              DAT 450
      WRITE (NW,320)                                                        DAT 460
      WRITE (NW,330) NX,NY,NB,NK,NT,NS,MBE,IOPT,IPRT                        DAT 470
C                                                                           DAT 480
      NXX=NX+1                                                              DAT 490
      NYY=NY+1                                                              DAT 500
      NBB=2*NB                                                              DAT 510
C     NBB - TOTAL NUMBER OF EQUATIONS                                       DAT 520
```

```
C                                                                       DAT 530
C     TIME PARAMETERS -                                                 DAT 540
      READ (NR,340) DELT                                                DAT 550
C     DELT - TIME STEP IN SECONDS                                       DAT 560
      WRITE (NW,370) DELT                                               DAT 570
C                                                                       DAT 580
C     DATA CODES -                                                      DAT 590
      READ (NR,310) KOD1,KOD2,KOD3,KOD4,KOD5,KOD6,KOD7,KOD8,KOD9       DAT 600
C     KOD1 - READ 1 IF X-SPACING IS CONSTANT                            DAT 610
C     KOD2 - READ 1 IF Y-SPACING IS CONSTANT                            DAT 620
C     KOD3 - READ 1 IF INITIAL PRESSURE IS CONSTANT                     DAT 630
C     KOD4 - READ 1 IF INITIAL ENTHALPY IS CONSTANT                     DAT 640
C     KOD5 - READ 1 IF X-PERMEABILITY IS CONSTANT                       DAT 650
C     KOD6 - READ 1 IF Y-PERMEABILITY IS CONSTANT                       DAT 660
C     KOD7 - READ 1 IF INITIAL POROSITY IS CONSTANT                     DAT 670
C     KOD8 - READ 1 IF RESERVOIR THICKNESS IS CONSTANT                  DAT 680
C     KOD9 - READ 1 IF TEMPERATURE IS READ IN PLACE OF ENTHALPY         DAT 690
      WRITE (NW,260) KOD1,KOD2,KOD3,KOD4,KOD5,KOD6,KOD7,KOD8,KOD9      DAT 700
C                                                                       DAT 710
C     SPACING -                                                         DAT 720
      IF (KOD1.EQ.1) GO TO 10                                           DAT 730
      READ (NR,340) (DX(I),I=1,NX)                                      DAT 740
      GO TO 30                                                          DAT 750
   10 READ (NR,340) DX1                                                 DAT 760
      DO 20 I=1,NX                                                      DAT 770
   20 DX(I)=DX1                                                         DAT 780
   30 WRITE (NW,350)                                                    DAT 790
      WRITE (NW,390) (DX(I),I=1,NX)                                     DAT 800
C                                                                       DAT 810
      IF (KOD2.EQ.1) GO TO 40                                           DAT 820
      READ (NR,340) (DY(J),J=1,NY)                                      DAT 830
      GO TO 60                                                          DAT 840
   40 READ (NR,340) DY1                                                 DAT 850
      DO 50 J=1,NY                                                      DAT 860
   50 DY(J)=DY1                                                         DAT 870
   60 WRITE (NW,360)                                                    DAT 880
      WRITE (NW,390) (DY(J),J=1,NY)                                     DAT 890
C                                                                       DAT 900
C     PRESSURE -                                                        DAT 910
      WRITE (NW,380)                                                    DAT 920
      CALL READ(P,KOD3,NY,NX)                                           DAT 930
C     ***********************                                           DAT 940
C                                                                       DAT 950
C     ENTHALPY -                                                        DAT 960
      IF (KOD9.EQ.1) WRITE (NW,270)                                     DAT 970
      IF (KOD9.NE.1) WRITE (NW,400)                                     DAT 980
      CALL READ(H,KOD4,NY,NX)                                           DAT 990
C     ***********************                                           DAT1000
      IF (KOD9.NE.1) GO TO 100                                          DAT1010
C                                                                       DAT1020
C     TEMPERATURES READ (COMPRESSED WATER REGION ASSUMED)               DAT1030
C     COMPUTE ENTHALPY BY NEWTON-RAPHSON METHOD                         DAT1040
```

```
      DO 80 J=1,NY                                                      DAT1050
      DO 80 I=1,NX                                                      DAT1060
      IF (H(I,J).EQ.0.) GO TO 80                                        DAT1070
      T=H(I,J)                                                          DAT1080
      H(I,J)=T*41841004.18                                              DAT1090
      PP=P(I,J)*0.1**7                                                  DAT1100
      HH=H(I,J)*0.1**7                                                  DAT1110
      DO 70 JJ=1,4                                                      DAT1120
      T1=-2.41231+2.56222D-01*HH-9.31415D-03*PP*PP-2.2568D-05*HH*HH     DAT1130
      DT=2.56222D-01-4.51360D-05*HH                                     DAT1140
      DH=-(T1-T)/DT                                                     DAT1150
      HH=HH+DH                                                          DAT1160
   70 CONTINUE                                                          DAT1170
      H(I,J)=HH*10.**7                                                  DAT1180
   80 CONTINUE                                                          DAT1190
      WRITE (NW,400)                                                    DAT1200
      DO 90 J=1,NY                                                      DAT1210
   90 WRITE (NW,390) (H(I,J),I=1,NX)                                    DAT1220
  100 CONTINUE                                                          DAT1230
C                                                                       DAT1240
C     X- AND Y-PERMEABILITY -                                           DAT1250
      WRITE (NW,410)                                                    DAT1260
      CALL READ(XK,KOD5,NY,NX)                                          DAT1270
C     ************************                                          DAT1280
C                                                                       DAT1290
      WRITE (NW,420)                                                    DAT1300
      CALL READ(YK,KOD6,NY,NX)                                          DAT1310
C     ************************                                          DAT1320
C                                                                       DAT1330
C     POROSITY -                                                        DAT1340
      WRITE (NW,430)                                                    DAT1350
      CALL READ(PHI,KOD7,NY,NX)                                         DAT1360
C     *************************                                         DAT1370
C     SAVE PRESSURE AND INITIAL POROSITY -                              DAT1380
      DO 110 I=1,NX                                                     DAT1390
      DO 110 J=1,NY                                                     DAT1400
      POLD(I,J)=P(I,J)                                                  DAT1410
  110 PHIO(I,J)=PHI(I,J)                                                DAT1420
C                                                                       DAT1430
C     RESERVOIR THICKNESS (BY BLOCKS) -                                 DAT1440
      WRITE (NW,440)                                                    DAT1450
      CALL READ(DZ,KOD8,NY,NX)                                          DAT1460
C     ************************                                          DAT1470
C                                                                       DAT1480
C     SOURCE/SINK (G/SEC) -                                             DAT1490
      WRITE (NW,450)                                                    DAT1500
      DO 120 J=1,NY                                                     DAT1510
      DO 120 I=1,NX                                                     DAT1520
  120 Q(I,J)=0.D0                                                       DAT1530
      IF (NS.EQ.0) GO TO 140                                            DAT1540
      DO 130 K=1,NS                                                     DAT1550
      READ (NR,460) I,J,Q(I,J)                                          DAT1560
```

```
  130 WRITE (NW,470) I,J,Q(I,J)                                          DAT1570
C                                                                        DAT1580
C     ROCK PROPERTIES -                                                  DAT1590
  140 READ (NR,340) XKC,COND,PHFWT,DF,BETA                               DAT1600
C     XKC - MEDIUM THERMAL CONDUCTIVITY                                  DAT1610
C     COND - CONFINING BED THERMAL CONDUCTIVITY                          DAT1620
C     PHFWT - ROCK ENTHALPY DERIVATIVE W.R.T. TEMPERATURE                DAT1630
C     DF - ROCK DENSITY                                                  DAT1640
C     BETA - ROCK COMPRESSIBILITY                                        DAT1650
      WRITE (NW,480) XKC,COND,PHFWT,DF,BETA                              DAT1660
C                                                                        DAT1670
C     COMPUTE COEFFICIENT FOR CONDUCTIVE LEAKAGE -                       DAT1680
      IF (COND.LT.0.1) GO TO 150                                         DAT1690
      COEF=DF*PHFWT/COND                                                 DAT1700
      GO TO 160                                                          DAT1710
  150 COEF=0.D0                                                          DAT1720
  160 CONTINUE                                                           DAT1730
C                                                                        DAT1740
C     NUMBER GRID BLOCKS -                                               DAT1750
C     NP(I,J) - SEQUENCE NUMBERING OF BLOCKS I,J                         DAT1760
      IBN=0                                                              DAT1770
      DO 190 I=1,NX                                                      DAT1780
      DO 180 J=1,NY                                                      DAT1790
      IF (DZ(I,J).LT.0.001) GO TO 170                                    DAT1800
      IBN=IBN+1                                                          DAT1810
      NP(I,J)=IBN                                                        DAT1820
      GO TO 180                                                          DAT1830
  170 NP(I,J)=0                                                          DAT1840
  180 CONTINUE                                                           DAT1850
  190 CONTINUE                                                           DAT1860
      WRITE (NW,240)                                                     DAT1870
      DO 200 J=1,NY                                                      DAT1880
  200 WRITE (NW,250) (NP(I,J),I=1,NX)                                    DAT1890
C                                                                        DAT1900
C     COMPUTE I,J FOR EACH SEQUENTIAL BLOCK NUMBER -                     DAT1910
      DO 220 I=1,NX                                                      DAT1920
      DO 220 J=1,NY                                                      DAT1930
      IJ=NP(I,J)                                                         DAT1940
      IF (IJ.EQ.0) GO TO 220                                             DAT1950
      NPP(IJ,1)=I                                                        DAT1960
      NPP(IJ,2)=J                                                        DAT1970
      GO TO 220                                                          DAT1980
  220 CONTINUE                                                           DAT2000
C                                                                        DAT2010
C     INITIALIZE SOLUTION VECTOR -                                       DAT2020
      DO 230 K=1,NB                                                      DAT2030
      II=NPP(K,1)                                                        DAT2040
      JJ=NPP(K,2)                                                        DAT2050
      X(2*K-1)=P(II,JJ)                                                  DAT2060
      X1(2*K-1)=P(II,JJ)                                                 DAT2070
      X(2*K)=H(II,JJ)                                                    DAT2080
      X1(2*K)=H(II,JJ)                                                   DAT2090
```

```
  230 CONTINUE                                                          DAT2100
      RETURN                                                            DAT2110
C                                                                       DAT2120
  240 FORMAT (11X,12HGRID NUMBERS/11X,12(1H-))                          DAT2130
  250 FORMAT (11X,16I5)                                                 DAT2140
  260 FORMAT (/////11X,5HCODES/11X,5(1H-)/11X,9I5)                      DAT2150
  270 FORMAT (/////,11X,19HINITIAL TEMPERATURE/11X,19(1H-)/)            DAT2160
  280 FORMAT (20A4)                                                     DAT2170
  290 FORMAT (1H1,///35X,25HSTEAM-WATER FLOW ANALYSIS//)                DAT2180
  300 FORMAT (11X,70(1H*)//11X,20A4//11X,70(1H*)///)                    DAT2190
  310 FORMAT (16I5)                                                     DAT2200
  320 FORMAT (/11X,22HFINITE DIFFERENCE DATA/11X,22(1H-)/)              DAT2210
  330 FORMAT (1H ,10X,11HNUMBER OF -,2X,7HCOLUMNS,I23/21X,1H-,2X,4HROWS,DAT2220
     1I26/21X,1H-,2X,15HNON-ZERO BLOCKS,I15/21X,1H-,2X,18HMAXIMUM ITERATDAT2230
     2IONS,I12/21X,1H-,2X,18HMAXIMUM TIME STEPS,I12/21X,1H-,2X,7HSOURCESDAT2240
     3,I23/21X,1H-,2X,19HESTIMATED BANDWIDTH,I11/21X,1H-,2X,16HWEIGHTINGDAT2250
     4 OPTION,I14/21X,1H-,2X,15HPRINTING OPTION,I15///)                 DAT2260
  340 FORMAT (8G10.0)                                                   DAT2270
  350 FORMAT (/////,11X,22HSPACING IN X-DIRECTION/11X,22(1H-)/)         DAT2280
  360 FORMAT (/////,11X,22HSPACING IN Y-DIRECTION/11X,22(1H-)/)         DAT2290
  370 FORMAT (///,11X,15HTIME PARAMETERS/11X,15(1H-)//11X,28HINITIAL TIMDAT2300
     1E STEP IN SECONDS,G22.8/)                                         DAT2310
  380 FORMAT (/////,11X,16HINITIAL PRESSURE/11X,16(1H-)/)               DAT2320
  390 FORMAT (/(11X,8(G12.5,2X)))                                       DAT2330
  400 FORMAT (/////,11X,16HINITIAL ENTHALPY/11X,16(1H-)/)               DAT2340
  410 FORMAT (/////,11X,14HX-PERMEABILITY/11X,14(1H-)/)                 DAT2350
  420 FORMAT (/////,11X,14HY-PERMEABILITY/11X,14(1H-)/)                 DAT2360
  430 FORMAT (/////,11X,16HINITIAL POROSITY/11X,16(1H-)/)               DAT2370
  440 FORMAT (/////,11X,19HRESERVOIR THICKNESS/11X,19(1H-)/)            DAT2380
  450 FORMAT (/////,11X,7HSOURCES/11X,7(1H-)/)                          DAT2390
  460 FORMAT (2I5,G15.0)                                                DAT2400
  470 FORMAT (11X,2I5,G15.8)                                            DAT2410
  480 FORMAT (/11X,15HROCK PROPERTIES/11X,15(1H-)/11X,28HMEDIUM THERMAL DAT2420
     1CONDUCTIVITY-G17.5/11X,35HCONFINING BED THERMAL CONDUCTIVITY-G10.3DAT2430
     2/11X,25HROCK ENTHALPY DERIVATIVE-G20.5/11X,13HROCK DENSITY-G32.5/1DAT2440
     31X,'ROCK COMPRESSIBILITY-',G24.5/)                                DAT2450
      END                                                               DAT2460-
```

```
      SUBROUTINE READ(DUM,KODE,NY,NX)                                   RED  10
C     ********************************                                  RED  20
      IMPLICIT REAL*8(A-H,O-Z)                                          RED  30
C                                                                       RED  40
C     CALLED FROM GDATA                                                 RED  50
C     PURPOSE:  TO READ TWO-DIMENSIONAL ARRAYS                          RED  60
C     ---------------------------------------------------------------   RED  70
      DIMENSION DUM(20,10)                                              RED  80
C     ---------------------------------------------------------------   RED  90
      NR=5                                                              RED 100
      NW=6                                                              RED 110
      IF (KODE.EQ.1) GO TO 20                                           RED 120
      DO 10 J=1,NY                                                      RED 130
   10 READ (NR,70) (DUM(I,J),I=1,NX)                                    RED 140
      GO TO 40                                                          RED 150
   20 READ (NR,70) DUM1                                                 RED 160
      DO 30 J=1,NY                                                      RED 170
      DO 30 I=1,NX                                                      RED 180
   30 DUM(I,J)=DUM1                                                     RED 190
   40 DO 50 J=1,NY                                                      RED 200
   50 WRITE (NW,60) (DUM(I,J),I=1,NX)                                   RED 210
      RETURN                                                            RED 220
C                                                                       RED 230
   60 FORMAT (/(11X,8(G12.5,2X)))                                       RED 240
   70 FORMAT (8G10.0)                                                   RED 250
      END                                                               RED 260-
```

```
      SUBROUTINE TCALC(DY,DX,DZ,XK,YK,TY,TYK,TX,TXK,XKC,NY,NX,NYY,NXX)  TCA  10
C     ****************************************************************  TCA  20
      IMPLICIT REAL*8(A-H,O-Z)                                          TCA  30
C                                                                       TCA  40
C     CALLED FROM MAIN                                                  TCA  50
C     PURPOSE:  TO COMPUTE TRANSMISSIBILITY TERMS -                     TCA  60
C     ----------------------------------------------------------------TCA  70
      DIMENSION DY(10), DX(20), DZ(20,10), XK(20,10), YK(20,10), TY(20,1TCA  80
     11), TYK(20,11), TX(21,10), TXK(21,10)                             TCA  90
C     ----------------------------------------------------------------TCA 100
C                                                                       TCA 110
C     COMPUTE TRANSMISSIBILITY TERMS IN THE X-DIRECTION -               TCA 120
      DO 10 J=1,NY                                                      TCA 130
C     IF ONLY ONE COLUMN, SKIP CALCULATIONS -                           TCA 140
      IF (NX.EQ.1) GO TO 20                                             TCA 150
      DO 10 I=2,NX                                                      TCA 160
      TXC=DY(J)*DZ(I-1,J)/DX(I-1)                                       TCA 170
      TXD=DY(J)*DZ(I,J)/DX(I)                                           TCA 180
      TXA=TXC*XK(I-1,J)                                                 TCA 190
      TXB=TXD*XK(I,J)                                                   TCA 200
      TTT=TXA+TXB                                                       TCA 210
      IF (TTT.EQ.0.0) GO TO 10                                          TCA 220
C     PERMEABILITY TERM -                                               TCA 230
      TX(I,J)=2.0*TXA*TXB/TTT                                           TCA 240
C     HEAT CONDUCTION TERM -                                            TCA 250
      TXK(I,J)=2.0*TXC*TXD/(TXC+TXD)*XKC                                TCA 260
   10 CONTINUE                                                          TCA 270
   20 CONTINUE                                                          TCA 280
C                                                                       TCA 290
C     COMPUTE TRANSMISSIBILITY TERMS IN THE Y-DIRECTION -               TCA 300
      DO 30 I=1,NX                                                      TCA 310
      IF (NY.EQ.1) GO TO 40                                             TCA 320
      DO 30 J=2,NY                                                      TCA 330
      TYC=DX(I)*DZ(I,J-1)/DY(J-1)                                       TCA 340
      TYD=DX(I)*DZ(I,J)/DY(J)                                           TCA 350
      TYA=TYC*YK(I,J-1)                                                 TCA 360
      TYB=TYD*YK(I,J)                                                   TCA 370
      TTT=TYA+TYB                                                       TCA 380
      IF (TTT.EQ.0.0) GO TO 30                                          TCA 390
      TY(I,J)=2.0*TYA*TYB/TTT                                           TCA 400
      TYK(I,J)=2.0*TYC*TYD/(TYC+TYD)*XKC                                TCA 410
   30 CONTINUE                                                          TCA 420
   40 CONTINUE                                                          TCA 430
C                                                                       TCA 440
C     SET TRANSMISSIBILITY OF BOUNDARY BLOCKS (REGULAR RECTANGULAR MESH)TCA 450
C     TO ZERO (NO-FLOW) -                                               TCA 460
      DO 50 I=1,NX                                                      TCA 470
      TY(I,1)=0.D0                                                      TCA 480
      TYK(I,1)=0.D0                                                     TCA 490
      TY(I,NYY)=0.D0                                                    TCA 500
   50 TYK(I,NYY)=0.D0                                                   TCA 510
C                                                                       TCA 520
```

```
      DO 60 J=1,NY                                                      TCA 530
      TX(1,J)=0.D0                                                      TCA 540
      TXK(1,J)=0.D0                                                     TCA 550
      TX(NXX,J)=0.D0                                                    TCA 560
   60 TXK(NXX,J)=0.D0                                                   TCA 570
C                                                                       TCA 580
C     SET TRANSMISSIBILITY OF BOUNDARY BLOCKS (EXTERNAL AND INTERNAL,   TCA 590
C     IRREGULAR MESH) TO ZERO -                                         TCA 600
      DO 80 I=1,NX                                                      TCA 610
      DO 70 J=1,NY                                                      TCA 620
      IF (DZ(I,J).GT.0.D0) GO TO 70                                     TCA 630
      TX(I+1,J)=0.D0                                                    TCA 640
      TXK(I+1,J)=0.D0                                                   TCA 650
      TY(I,J+1)=0.D0                                                    TCA 660
      TYK(I,J+1)=0.D0                                                   TCA 670
      TXK(I,J)=0.D0                                                     TCA 680
      TX(I,J)=0.D0                                                      TCA 690
      TYK(I,J)=0.D0                                                     TCA 700
      TY(I,J)=0.D0                                                      TCA 710
   70 CONTINUE                                                          TCA 720
   80 CONTINUE                                                          TCA 730
      RETURN                                                            TCA 740
      END                                                               TCA 750-
```

```
      SUBROUTINE MATR(NP,NY,NX,MBE,MBW)                                  MAT  10
C     **********************************                                 MAT  20
      IMPLICIT REAL*8(A-H,O-Z)                                           MAT  30
C                                                                        MAT  40
C     CALLED FROM MAIN                                                   MAT  50
C     PURPOSE:  COMPUTE MATRIX BANDWIDTH                                 MAT  60
C     -------------------------------------------------------------------MAT  70
      DIMENSION N(5), NP(20,10)                                          MAT  80
C     -------------------------------------------------------------------MAT  90
C                                                                        MAT 100
C     COMPUTE MAXIMUM DIFFERENCE BETWEEN SEQUENTIAL BLOCK NUMBERS        MAT 110
C     OF ADJACENT BLOCKS -                                               MAT 120
      MM=0                                                               MAT 130
      DO 80 I=1,NX                                                       MAT 140
      DO 70 J=1,NY                                                       MAT 150
      DO 10 KK=1,4                                                       MAT 160
   10 N(KK)=0                                                            MAT 170
      I1=I-1                                                             MAT 180
      I2=I+1                                                             MAT 190
      J1=J-1                                                             MAT 200
      J2=J+1                                                             MAT 210
      IF (I1.EQ.0) GO TO 20                                              MAT 220
      N(1)=NP(I1,J)                                                      MAT 230
   20 CONTINUE                                                           MAT 240
      IF (I2.GT.NX) GO TO 30                                             MAT 250
      N(3)=NP(I2,J)                                                      MAT 260
   30 CONTINUE                                                           MAT 270
      IF (J1.EQ.0) GO TO 40                                              MAT 280
      N(2)=NP(I,J1)                                                      MAT 290
   40 CONTINUE                                                           MAT 300
      IF (J2.GT.NY) GO TO 50                                             MAT 310
      N(4)=NP(I,J2)                                                      MAT 320
   50 N(5)=NP(I,J)                                                       MAT 330
      IF (N(5).EQ.0) GO TO 70                                            MAT 340
      DO 60 K=1,4                                                        MAT 350
      IF (N(K).EQ.0) GO TO 60                                            MAT 360
C     COMPUTE DIFFERENCE -                                               MAT 370
      NN=N(5)-N(K)                                                       MAT 380
      IF (NN.LT.0) NN=-NN                                                MAT 390
C     COMPUTE MAXIMUM DIFFERENCE -                                       MAT 400
      IF (NN.GT.MM) MM=NN                                                MAT 410
   60 CONTINUE                                                           MAT 420
   70 CONTINUE                                                           MAT 430
   80 CONTINUE                                                           MAT 440
C     COMPUTE BANDWIDTH -                                                MAT 450
      MBW=2*(2*MM+1)+1                                                   MAT 460
      PRINT 90, MBW,MBE                                                  MAT 470
      RETURN                                                             MAT 480
C                                                                        MAT 490
   90 FORMAT (//11X,'MAXIMUM BANDWIDTH IS ',I4,2X,'COMPARED TO ESTIMATED'MAT 500
     1 BANDWIDTH OF ',I4///)                                             MAT 510
      END                                                                MAT 520-
```

```
      SUBROUTINE PRPTY(L,KKK)                                          PRP  10
C     **********************                                           PRP  20
      IMPLICIT REAL*8(A-H,O-Z)                                         PRP  30
C                                                                      PRP  40
C     CALLED FROM MAIN                                                 PRP  50
C     PURPOSE:  TO COMPUTE THERMODYNAMIC PROPERTIES                    PRP  60
C     ---------------------------------------------------------------PRP  70
      DIMENSION SW1(20,10), TEMP2(20,10), DEN2(20,10)                  PRP  80
      DIMENSION CQ(20,10)                                              PRP  90
C                                                                      PRP 100
      COMMON /INPUT/ PHI(20,10),XK(20,10),YK(20,10),P(20,10),H(20,10),X(PRP 110
     1400),X1(400),NP(20,10),NPP(200,2),DX(20),DY(10),DZ(20,10),G(20,10)PRP 120
      COMMON /CONTRO/ NK,NX,NXX,NY,NYY,NB,NBB,NT,DELT,TIME,PHFWT,DF,XKC,PRP 130
     1COND,COEF,BETA,IPRT                                             PRP 140
      COMMON /WORK/ AX(21,10),BX(21,10),EX(21,10),AY(20,11),BY(20,11),EYPRP 150
     1(20,11),TX(21,10),TXK(21,10),TY(20,11),TYK(20,11),XM(20,10),XMASS(PRP 160
     220,10),EN(20,10),ENERGY(20,10),C(20,10),D(20,10),F(20,10),G(20,10)PRP 170
     3,QH(20,10),XN(20,10),TM(20,10),DTP(20,10),DTH(20,10),PHIC(20,10),PPRP 180
     40LD(20,10)                                                      PRP 190
      COMMON /CHECK/ IND(200),INDOLD(200)                              PRP 200
C     ---------------------------------------------------------------PRP 210
C                                                                      PRP 220
C     STATEMENT FUNCTIONS FOR THERMODYNAMIC PROPERTIES                 PRP 230
C                                                                      PRP 240
      F1(PX,HX)=1.00207+4.42607D-4*PX-5.47456D-5*HX+5.02875D-7*HX*PX-1.2PRP 250
     147910-7*HX*HX                                                   PRP 260
      F2(PX,HX)=-2.41231+2.56222D-1*HX-9.31415D-3*PX*PX-2.2568D-5*HX*HX PRP 270
      F3(PX,HX)=-2.26162D-5+0.043844100*PX-1.79088D-5*PX*HX+3.69276D-8*PPRP 280
     1X*PX*PX*PX+5.17644D-13*HX*HX*HX*PX                              PRP 290
      F4(PX,HX)=-374.66900+47.9921D0*PX-0.633606D0*PX*PX+7.39386D-5*PX*HPRP 300
     1X-3.33720D6/HX/HX/PX/PX+0.0357154D0/PX/PX/PX-1.1725D-9*HX*HX*HX*PX-PRP 310
     22.26861D15/HX/HX/HX/HX                                          PRP 320
C                                                                      PRP 330
C     F1 - DENSITY, COMPRESSED WATER REGION                            PRP 340
C     F2 - TEMPERATURE, COMPRESSED WATER REGION                        PRP 350
C     F3 - DENSITY,  SUPER HEATED STEAM REGION                         PRP 360
C     F4 - TEMPERATURE, SUPER HEATED STEAM REGION                      PRP 370
      NKK=NK                                                           PRP 380
      IF (L.NE.1) GO TO 20                                             PRP 390
      IF (KKK.NE.1) GO TO 20                                           PRP 400
C                                                                      PRP 410
C     INITIALIZE ARRAYS -                                              PRP 420
      DO 10 I=1,NX                                                     PRP 430
      DO 10 J=1,NY                                                     PRP 440
      XN(I,J)=0.D0                                                     PRP 450
      TM(I,J)=0.D0                                                     PRP 460
      C(I,J)=0.D0                                                      PRP 470
      D(I,J)=0.D0                                                      PRP 480
      F(I,J)=0.D0                                                      PRP 490
      G(I,J)=0.D0                                                      PRP 500
      QH(I,J)=0.D0                                                     PRP 510
      DTP(I,J)=0.D0                                                    PRP 520
```

```
      DTH(I,J)=0.D0                                                     PRP 530
      SW1(I,J)=0.D0                                                     PRP 540
      DEN2(I,J)=0.D0                                                    PRP 550
      TEMP2(I,J)=0.D0                                                   PRP 560
   10 CONTINUE                                                          PRP 570
   20 CONTINUE                                                          PRP 580
      DO 80 I=1,NX                                                      PRP 590
      DO 80 J=1,NY                                                      PRP 600
      IF (DZ(I,J).EQ.0.D0) GO TO 80                                     PRP 610
      PP=P(I,J)                                                         PRP 620
      HH=H(I,J)                                                         PRP 630
      PP=PP*0.1**7                                                      PRP 640
      HH=HH*0.1**7                                                      PRP 650
C                                                                       PRP 660
      K=NP(I,J)                                                         PRP 670
      IF (IND(K)) 30,40,50                                              PRP 700
   30 CONTINUE                                                          PRP 710
C                                                                       PRP 720
C                                                                       PRP 730
C  COMPRESSED WATER REGION -                                            PRP 740
C                                                                       PRP 750
      DW=F1(PP,HH)                                                      PRP 760
      TEMP=F2(PP,HH)                                                    PRP 770
      PDWP=4.42607D-11+5.02875D-14*HH                                   PRP 780
      PDWH=-5.47456D-12+5.02875D-14*PP-2.49582D-14*HH                   PRP 790
      PTWP=-1.86283D-9*PP                                               PRP 800
      PTWH=2.56222D-8-4.51360D-12*HH                                    PRP 810
C     WATER VISCOSITY -                                                 PRP 820
C     1967 ASME STEAM TABLES FORMULA (P. 74) -                          PRP 830
      VW=1.0D-6*(241.4*10.**(247.8/(TEMP+133.15)))                      PRP 840
C                                                                       PRP 850
C     COMPUTE EQUATION COEFFICIENTS -                                   PRP 860
C     SPACE -                                                           PRP 870
      XN(I,J)=DW/VW                                                     PRP 880
      TM(I,J)=DW*HH/VW*10.**7                                           PRP 890
      DTH(I,J)=PTWH                                                     PRP 900
      DTP(I,J)=PTWP                                                     PRP 910
C     TIME -                                                            PRP 920
      C(I,J)=PDWP*PHI(I,J)*HH*10.**7+PHFWT*PTWP*DF*(1.0-PHI(I,J))       PRP 930
      C(I,J)=C(I,J)+DW*HH*1.D7*BETA*PHI0(I,J)-PH10(I,J)*BETA*DF*PHFWT*TEPRP 940
     1MP                                                                PRP 950
      D(I,J)=PDWH*PHI(I,J)*HH*10.**7+PHFWT*PTWH*DF*(1.0-PHI(I,J))+DW*PH1PRP 960
     1(I,J)                                                             PRP 970
      F(I,J)=PHI(I,J)*PDWP+DW*BETA*PHI0(I,J)                            PRP 980
      G(I,J)=PHI(I,J)*PDWH                                              PRP 990
C                                                                       PRP1000
      SW1(I,J)=1.D0                                                     PRP1010
      TEMP2(I,J)=TEMP                                                   PRP1020
      DEN2(I,J)=DW                                                      PRP1030
C     HEAT DISCHARGE -                                                  PRP1040
      QH(I,J)=Q(I,J)*HH*10.**7                                          PRP1050
      GO TO 60                                                          PRP1060
```

```
   40 CONTINUE                                                          PRP1070
C                                                                       PRP1080
C                                                                       PRP1090
C  TWO-PHASE REGION -                                                   PRP1100
      HW=730.984+129.239*PP-10.0333*PP*PP+0.39881*PP*PP*PP
     1   -99.0697/PP+12.9267/PP/PP-0.628359/PP/PP/PP
      HS=2822.82-39.952/PP+2.54342/PP/PP-0.938879*PP*PP                 PRP1110
      TEMP=F2(PP,HW)                                                    PRP1120
      DW=F1(PP,HW)                                                      PRP1130
      DS=F3(PP,HS)                                                      PRP1140
C     WATER SATURATION -                                                PRP1150
      SW=DS*(HS-HH)/(HH*(DW-DS)-(HW*DW-HS*DS))                          PRP1160
      SST=1.0-SW                                                        PRP1170
      SWO=SW                                                            PRP1180
C     VISCOSITIES -                                                     PRP1190
C     1967 ASME STEAM TABLES FORMULA (P. 74) -                          PRP1200
      VW=1.0E-06*(241.4*10.**(247.8/(TEMP+133.15)))                     PRP1210
      VS=1.0E-06*(.407*TEMP+80.4)                                       PRP1220
C     RELATIVE PERMEABILITY (WATER) -                                   PRP1230
      XKW=((SW-0.35)/0.65)**4                                           PRP1240
C     RELATIVE PERMEABILITY (STEAM) -                                   PRP1250
      XKS=(1.0-((SW-0.35)/0.65)**2)*(1.0-((SW-0.35)/0.65))**2           PRP1260
      IF (SW.LT.0.3) XKW=0.0                                            PRP1270
      IF (SW.LT.0.3) XKS=1.0                                            PRP1280
      IF (SW.GT.0.95) XKW=1.0                                           PRP1290
      IF (SW.GT.0.95) XKS=0.0                                           PRP1300
      DEN=DW*SW+SST*DS                                                  PRP1310
      HHH=HH*1.001D0                                                    PRP1320
      SW=DS*(HS-HHH)/(HHH*(DW-DS)-(HW*DW-HS*DS))                        PRP1330
      SST=1.0-SW                                                        PRP1340
      DEN1=DW*SW+DS*SST                                                 PRP1350
      PDWH=(DEN1-DEN)/(HHH-HH)*1.D-7                                    PRP1360
      PPP=PP*.999D0                                                     PRP1370
      HHW=730.984+129.239*PPP-10.0333*PPP*PPP+0.39881*PPP*PPP*PPP
     1   -99.0697/PPP+12.9267/PPP/PPP-0.628359/PPP/PPP/PPP
      HHS=2822.82-39.952/PPP+2.54342/PPP/PPP-0.938879*PPP*PPP           PRP1400
      T1=F2(PPP,HHW)                                                    PRP1410
      DDW=F1(PPP,HHW)                                                   PRP1420
      DDS=F3(PPP,HHS)                                                   PRP1430
      SW=DDS*(HHS-HH)/(HH*(DDW-DDS)-(HHW*DDW-HHS*DDS))                  PRP1440
      SST=1.0-SW                                                        PRP1450
      DEN1=DDW*SW+DDS*SST                                               PRP1460
      PDWP=(DEN1-DEN)/(PPP-PP)*1.D-7                                    PRP1470
      TEMP1=T1                                                          PRP1480
      PTWP=(TEMP1-TEMP)/(PPP-PP)*1.D-7                                  PRP1490
      SW=SWO                                                            PRP1500
C                                                                       PRP1510
C     SPACE -                                                           PRP1520
      XN(I,J)=XKW*DW/VW+XKS*DS/VS                                       PRP1530
      TM(I,J)=(XKW*DW*HW/VW+XKS*DS*HS/VS)*10.**7                        PRP1540
      DTP(I,J)=PTWP                                                     PRP1550
      DTH(I,J)=0.0                                                      PRP1560
```

```
C     TIME -                                                                    PRP1570
      C(I,J)=PDWP*PHI(I,J)*HH*10.**7+PHFWT*PTWP*DF*(1.0-PHI(I,J))               PRP1580
      C(I,J)=C(I,J)+DEN*HH*1.D7*BETA*PHIO(I,J)-PHIO(I,J)*BETA*DF*PHFWT*TPRP1590
     1EMP                                                                       PRP1600
      D(I,J)=PDWH*PHI(I,J)*HH*10.**7+DEN*PHI(I,J)                               PRP1610
      F(I,J)=PHI(I,J)*PDWP+DEN*BETA*PHIO(I,J)                                   PRP1620
      G(I,J)=PHI(I,J)*PDWH                                                      PRP1630
C                                                                               PRP1640
      SW1(I,J)=SW                                                               PRP1650
      TEMP2(I,J)=TEMP                                                           PRP1660
      DEN2(I,J)=DEN                                                             PRP1670
      QH(I,J)=Q(I,J)*(HW+(HS-HW)*XKS/(XKS+VS*XKW/VW*DW/DS))*10.**7              PRP1680
      GO TO 60                                                                  PRP1690
   50 CONTINUE                                                                  PRP1700
C                                                                               PRP1710
C                                                                               PRP1720
C  SUPER-HEATED STEAM REGION -                                                  PRP1730
      DS=F3(PP,HH)                                                              PRP1740
      PDWH=-1.79088D-12*PP+1.552932D-19*HH*HH*PP                                PRP1800
      PDWP=4.38441D-9-1.79088D-12*HH+1.477104D-14*PP*PP*PP+5.17644D-20*HPRP1810
     1H*HH*HH                                                                   PRP1820
      HW=730.984+129.239*PP-10.0333*PP*PP+0.39881*PP*PP*PP
     1   -99.0697/PP+12.9267/PP/PP-0.628359/PP/PP/PP
      HS=2822.82-39.952/PP+2.54342/PP/PP-0.938879*PP*PP
      TT1=F4(PP,HH)
      TT2=F4(PP,HS)
      TT3=F2(PP,HW)
      TEMP=TT1-TT2+TT3
      HHH=HH*1.001D0
      TT1=F4(PP,HHH)
      TEMPH=TT1-TT2+TT3
      PPP=PP*0.999D0
      HW =730.984+129.239*PPP-10.0333*PPP*PPP+0.39881*PPP*PPP*PPP
     1   -99.0697/PPP+12.9267/PPP/PPP-0.628359/PPP/PPP/PPP
      HS=2822.82-39.952/PPP+2.54342/PPP/PPP-0.938879*PPP*PPP
      TT1=F4(PPP,HH)
      TT2=F4(PPP,HS)
      TT3=F2(PPP,HW)
      TEMPP=TT1-TT2+TT3
      PTWH=(TEMPH-TEMP)/(HHH-HH)*1.D-7
      PTWP=(TEMPP-TEMP)/(PPP-PP)*1.D-7
C     STEAM VISCOSITY -                                                         PRP1830
C     1967 ASME STEAM TABLES FORMULA (P. 74) -                                  PRP1840
      VS=1.0E-06*(.407*TEMP+80.4)                                               PRP1850
C                                                                               PRP1860
C     SPACE -                                                                   PRP1870
      XN(I,J)=DS/VS                                                             PRP1880
      TM(I,J)=DS*HH/VS*10.**7                                                   PRP1890
      DTH(I,J)=PTWH                                                             PRP1900
      DTP(I,J)=PTWP                                                             PRP1910
C     TIME -                                                                    PRP1920
      C(I,J)=PDWP*PHI(I,J)*HH*10.**7+PHFWT*PTWP*DF*(1.0-PHI(I,J))               PRP1930
```

```
      C(I,J)=C(I,J)+DS*HH*1.07*BETA*PHIO(I,J)-PHIO(I,J)*BETA*DF*PHFWT*TEPRP1940
     1MP                                                                 PRP1950
      D(I,J)=PDWH*PHI(I,J)*HH*10.**7+PHFWT*PDWH*DF*(1.0-PHI(I,J))+DS*PHIPRP1960
     1(I,J)                                                              PRP1970
      F(I,J)=PHI(I,J)*PDWP+DS*BETA*PHIO(I,J)                             PRP1980
      G(I,J)=PHI(I,J)*PDWH                                               PRP1990
C                                                                        PRP2000
      SW1(I,J)=0.D0                                                      PRP2010
      TEMP2(I,J)=TEMP                                                    PRP2020
      DEN2(I,J)=DS                                                       PRP2030
      QH(I,J)=Q(I,J)*HH*10.**7                                           PRP2040
   60 CONTINUE                                                           PRP2050
      IF (COEF.LT.0.1) GO TO 80                                          PRP2060
C                                                                        PRP2070
C     TRANSIENT CONDUCTIVE HEAT LEAKAGE ONLY -                           PRP2080
      IF (KKK.NE.1) GO TO 70                                             PRP2090
      IC=L+KKK                                                           PRP2100
      II=NP(I,J)                                                         PRP2110
      CALL VERTCD(DELT,TEMP,COEF,COND,CQQ,II,IC)                         PRP2120
C     ******************************************                         PRP2130
      CQQ=CQQ*DX(I)*DY(J)                                                PRP2140
      CQ(I,J)=CQQ                                                        PRP2150
   70 CONTINUE                                                           PRP2160
      QH(I,J)=QH(I,J)+CQ(I,J)                                            PRP2170
   80 CONTINUE                                                           PRP2180
      IF (KKK.NE.1) GO TO 120                                            PRP2190
C     PRINT THERMO DATA EVERY IPRT TIME STEP -                           PRP2200
      IF (MOD(L,IPRT).NE.0) GO TO 120                                    PRP2210
      PRINT 130                                                          PRP2220
      DO 90 J=1,NY                                                       PRP2230
      PRINT 140, (SW1(I,J),I=1,NX)                                       PRP2240
   90 CONTINUE                                                           PRP2250
      PRINT 150                                                          PRP2260
      DO 100 J=1,NY                                                      PRP2270
      PRINT 140, (TEMP2(I,J),I=1,NX)                                     PRP2280
  100 CONTINUE                                                           PRP2290
      PRINT 160                                                          PRP2300
      DO 110 J=1,NY                                                      PRP2310
      PRINT 140, (DEN2(I,J),I=1,NX)                                      PRP2320
  110 CONTINUE                                                           PRP2330
  120 CONTINUE                                                           PRP2340
      RETURN                                                             PRP2350
C                                                                        PRP2360
  130 FORMAT (//11X,'WATER SATURATIONS'/11X,17(1H-)//)                   PRP2370
  140 FORMAT (/(11X,8(G12.5,2X)))                                        PRP2380
  150 FORMAT (//11X,'TEMPERATURES'/11X,12(1H-)//)                        PRP2390
  160 FORMAT (//11X,'DENSITY'/11X,7(1H-)//)                              PRP2400
      END                                                                PRP2410-
```

```
      SUBROUTINE VERTCD(DELT,TT,COEF,COND,CW,II,IC)                     VRT  10
C     ***********************************************                   VRT  20
      IMPLICIT REAL*8(A-H,O-Z)                                          VRT  30
C                                                                       VRT  40
C     CALLED FROM PRPTY                                                 VRT  50
C     PURPOSE:  TO COMPUTE VERTICAL CONDUCTIVE LEAKAGE THROUGH THE      VRT  60
C               CONFINING BED BY THE FINITE ELEMENT METHOD              VRT  70
C     ------------------------------------------------------------------VRT  80
      DIMENSION A(9), B(9), C(9), D(9), F(9), DTEMP(200,11), DELZ(200,10VRT  90
     1), TI(200), DDTEMP(200,11)                                        VRT 100
C     ------------------------------------------------------------------VRT 110
C                                                                       VRT 120
C     COMPUTE DELTA Z ( ON FIRST TIME STEP) -                           VRT 130
      CTHK1=102300.D0                                                   VRT 140
      IF (IC.NE.2) GO TO 30                                             VRT 150
      NNZ=11                                                            VRT 160
C     NNZ - NUMBER OF NODES IN Z-DIRECTION                              VRT 170
C     COMPUTE DELZ (BASED ON TEN ELEMENTS AND DOUBLING THE SIZE OF EACH VRT 180
C     ELEMENT) -                                                        VRT 190
      NNL=NNZ-1                                                         VRT 200
C     NNL - NUMBER OF ELEMENTS IN Z-DIRECTION                           VRT 210
      DELZ(II,NNL)=CTHK1/1023.D0                                        VRT 220
      SUM=DELZ(II,NNL)                                                  VRT 230
      JSTOP=NNL-2                                                       VRT 240
      DO 10 JJ=1,JSTOP                                                  VRT 250
      DELZ(II,NNL-JJ)=2.*DELZ(II,NNL-JJ+1)                              VRT 260
   10 SUM=SUM+DELZ(II,NNL-JJ)                                           VRT 270
      DELZ(II,1)=CTHK1-SUM                                              VRT 280
C                                                                       VRT 290
C     INITIALIZE TI AND DDTEMP -                                        VRT 300
      TI(II)=TT                                                         VRT 310
      DO 20 J=1,NNZ                                                     VRT 320
   20 DDTEMP(II,J)=0.D0                                                 VRT 330
   30 CONTINUE                                                          VRT 340
C                                                                       VRT 350
C     INITIALIZE DTEMP -                                                VRT 360
      DO 40 J=1,NNZ                                                     VRT 370
   40 DTEMP(II,J)=DDTEMP(II,J)                                          VRT 380
C                                                                       VRT 390
C     FACTOR FOR TIME-DERIVATIVE -                                      VRT 400
      THETA=1.D0                                                        VRT 410
C                                                                       VRT 420
C     SEE COATS,ET.AL., SPE J (DEC 1974) 590 FOR 1-DIM EQUATION.        VRT 430
C     BOUNDARY CONDITIONS -                                             VRT 440
C     UPPER -                                                           VRT 450
      DTEMP(II,1)=0.D0                                                  VRT 460
C     LOWER -                                                           VRT 470
      DTEMP(II,NNZ)=TT-TI(II)                                           VRT 480
C                                                                       VRT 490
C                                                                       VRT 500
C     CALCULATE FINITE ELEMENT COEFFICIENTS FOR MATRIX EQUATION -       VRT 510
      J=0                                                               VRT 520
```

```
      DO 50 I=2,NNL                                                     VRT 530
      J=J+1                                                             VRT 540
      A1=DELZ(II,I-1)*COEF/(6.D0*DELT)                                  VRT 550
      C1=DELZ(II,I)*COEF/(6.D0*DELT)                                    VRT 560
C                                                                       VRT 570
      A(J)=A1-THETA/DELZ(II,I-1)                                        VRT 580
      B(J)=2.D0*A1+THETA/DELZ(II,I-1)+2.D0*C1+THETA/DELZ(II,I)          VRT 590
      C(J)=C1-THETA/DELZ(II,I)                                          VRT 600
C     RIGHT-HAND SIDE -                                                 VRT 610
C     USE DDTEMP FROM LAST SOLUTION AS INITIAL CONDITIONS               VRT 620
      D(J)=(A1+(1.D0-THETA)/DELZ(II,I-1))*DDTEMP(II,I-1)+(2.D0*A1-(1.D0-VRT 630
     1THETA)/DELZ(II,I-1)+2.D0*C1-(1.D0-THETA)/DELZ(II,I))*DDTEMP(II,I)+VRT 640
     2(C1+(1.D0-THETA)/DELZ(II,I))*DDTEMP(II,I+1)                       VRT 650
   50 CONTINUE                                                          VRT 660
C                                                                       VRT 670
C     SOLVE BY THOMAS METHOD (SEE CALIF BULL NO 63-4  SEPT, 1971 P 521) VRT 680
C     NOTE THAT THERE ARE NNZ-2=J EQUATIONS AND J UNKNOWNS              VRT 690
C                                                                       VRT 700
C     UPPER TRIANGULARIZE -                                             VRT 710
      F(1)=B(1)                                                         VRT 720
      D(1)=D(1)-A(1)*DTEMP(II,1)                                        VRT 730
      D(J)=D(J)-C(J)*DTEMP(II,NNZ)                                      VRT 740
      DO 60 I=2,J                                                       VRT 750
      F(I)=B(I)-C(I-1)*A(I)/F(I-1)                                      VRT 760
   60 D(I)=D(I)-A(I)*D(I-1)/F(I-1)                                      VRT 770
C                                                                       VRT 780
C     BACK SUBSTITUTE -                                                 VRT 790
      DTEMP(II,J+1)=D(J)/F(J)                                           VRT 800
      J1=J-1                                                            VRT 810
      DO 70 I=1,J1                                                      VRT 820
   70 DTEMP(II,J+1-I)=(D(J-I)-C(J-I)*DTEMP(II,J+2-I))/F(J-I)            VRT 830
C                                                                       VRT 840
C     DTEMP NOW CONTAINS NEW VALUES FOR TEMPERATURE DIFFERENCE          VRT 850
C                                                                       VRT 860
C     COMPUTE TRANSIENT CONDUCTIVE LEAKAGE                              VRT 870
      CQ=-2.D0*COND*(DTEMP(II,NNZ)-DTEMP(II,NNZ-1))/DELZ(II,NNL)        VRT 880
C                                                                       VRT 890
C     SAVE DTEMP                                                        VRT 900
      DO 80 J=1,NNZ                                                     VRT 910
   80 DDTEMP(II,J)=DTEMP(II,J)                                          VRT 920
      RETURN                                                            VRT 930
      END                                                               VRT 940-
```

```
      SUBROUTINE FORMEQ(A,R,NBBD,MBED)                                  FEQ  10
C     ********************************                                  FEQ  20
      IMPLICIT REAL*8(A-H,O-Z)                                          FEQ  30
C     CALLED FROM MAIN                                                  FEQ  40
C     PURPOSE:  TO FORM THE FINAL MATRIX EQUATION                       FEQ  50
C     ------------------------------------------------------------------FEQ  60
      DIMENSION A(NBBD,MBED), R(NBBD)                                   FEQ  70
C                                                                       FEQ  80
      COMMON /INPUT/ PHI(20,10),XK(20,10),YK(20,10),P(20,10),H(20,10),X(FEQ  90
     1400),X1(400),NP(20,10),NPP(200,2),DX(20),DY(10),DZ(20,10),G(20,10)FEQ 100
      COMMON /CONTRO/ NK,NX,NXX,NY,NYY,NB,NBB,NT,DELT,TIME,PHFWT,DF,XKC,FEQ 110
     1COND,COEF,BETA,IPRT                                               FEQ 120
      COMMON /WORK/ AX(21,10),BX(21,10),EX(21,10),AY(20,11),BY(20,11),EYFEQ 130
     1(20,11),TX(21,10),TXK(21,10),TY(20,11),TYK(20,11),XM(20,10),XMASS(FEQ 140
     220,10),EN(20,10),ENERGY(20,10),C(20,10),D(20,10),F(20,10),G(20,10)FEQ 150
     3,QH(20,10),XN(20,10),TM(20,10),DTP(20,10),DTH(20,10),PHIC(20,10),PFEQ 160
     4OLD(20,10)                                                        FEQ 170
      COMMON /EXTRA/ MBW,MBE                                            FEQ 180
C     ------------------------------------------------------------------FEQ 190
      MP=(MBW+1)/2                                                      FEQ 200
C     MP - MAIN DIAGONAL (STORED VERTICALLY)                            FEQ 210
C                                                                       FEQ 220
C     INITIALIZE ARRAYS -                                               FEQ 230
      DO 10 K=1,NBB                                                     FEQ 240
      DO 10 KK=1,MBW                                                    FEQ 250
   10 A(K,KK)=0.D0                                                      FEQ 260
C                                                                       FEQ 270
C     COMPUTE MATRIX -                                                  FEQ 280
      DO 50 K=1,NB                                                      FEQ 290
      I=NPP(K,1)                                                        FEQ 300
      J=NPP(K,2)                                                        FEQ 310
C                                                                       FEQ 320
C     MAIN DIAGONAL TERMS -                                             FEQ 330
C     PRESSURE -                                                        FEQ 340
      A(2*K-1,MP)=-EX(I,J)-EY(I,J)-EX(I+1,J)-EY(I,J+1)                  FEQ 350
C     ENTHALPY -                                                        FEQ 360
      A(2*K,MP-1)=-AX(I,J)-AY(I,J)-AX(I+1,J)-AY(I,J+1)                  FEQ 370
      A(2*K,MP)=-BX(I,J)-BY(I,J)-BX(I+1,J)-BY(I,J+1)                    FEQ 380
C                                                                       FEQ 390
C     OFF DIAGONAL TERMS -                                              FEQ 400
      II=I                                                              FEQ 410
      IF (J.EQ.1) GO TO 20                                              FEQ 420
      JJ=J-1                                                            FEQ 430
      NN=NP(II,JJ)                                                      FEQ 440
      IF (NN.EQ.0) GO TO 20                                             FEQ 450
      NC=MP+(NN-K)*2                                                    FEQ 460
      A(2*K-1,NC)=EY(I,J)                                               FEQ 470
      A(2*K,NC-1)=AY(I,J)                                               FEQ 480
      A(2*K,NC)=BY(I,J)                                                 FEQ 490
   20 CONTINUE                                                          FEQ 500
      IF (J.EQ.NY) GO TO 30                                             FEQ 510
      JJ=J+1                                                            FEQ 520
```

```
      NN=NP(II,JJ)                                                      FEG 530
      IF (NN.EQ.0) GO TO 30                                             FEG 540
      NC=MP+(NN-K)*2                                                    FEG 550
      A(2*K-1,NC)=EY(I,J+1)                                             FEG 560
      A(2*K,NC-1)=AY(I,J+1)                                             FEG 570
      A(2*K,NC)=BY(I,J+1)                                               FEG 580
   30 JJ=J                                                              FEG 590
      IF (I.EQ.1) GO TO 40                                              FEG 600
      II=I-1                                                            FEG 610
      NN=NP(II,JJ)                                                      FEG 620
      IF (NN.EQ.0) GO TO 40                                             FEG 630
      NC=MP+(NN-K)*2                                                    FEG 640
      A(2*K-1,NC)=EX(I,J)                                               FEG 650
      A(2*K,NC-1)=AX(I,J)                                               FEG 660
      A(2*K,NC)=BX(I,J)                                                 FEG 670
   40 IF (I.EQ.NX) GO TO 50                                             FEG 680
      II=I+1                                                            FEG 690
      NN=NP(II,JJ)                                                      FEG 700
      IF (NN.EQ.0) GO TO 50                                             FEG 710
      NC=MP+(NN-K)*2                                                    FEG 720
      A(2*K-1,NC)=EX(I+1,J)                                             FEG 730
      A(2*K,NC-1)=AX(I+1,J)                                             FEG 740
      A(2*K,NC)=BX(I+1,J)                                               FEG 750
   50 CONTINUE                                                          FEG 760
C                                                                       FEG 770
C     COMPUTE KNOWN VECTOR -                                            FEG 780
      DO 70 K=1,NBB                                                     FEG 790
      II=K-MP                                                           FEG 800
      R(K)=0.D0                                                         FEG 810
      DO 60 JJ=1,MBW                                                    FEG 820
      II=II+1                                                           FEG 830
      IF (II.LT.1) GO TO 60                                             FEG 840
      IF (II.GT.NBB) GO TO 70                                           FEG 850
      R(K)=R(K)+A(K,JJ)*X1(II)                                          FEG 860
   60 CONTINUE                                                          FEG 870
   70 CONTINUE                                                          FEG 880
C                                                                       FEG 890
      DO 80 K=1,NB                                                      FEG 900
      I=NPP(K,1)                                                        FEG 910
      J=NPP(K,2)                                                        FEG 920
C                                                                       FEG 930
C     NEWTON-RAPHSON RESIDUAL VECTOR -                                  FEG 940
      R(2*K-1)=-R(2*K-1)+XM(I,J)/DELT-XMASS(I,J)/DELT-G(I,J)            FEG 950
      R(2*K)=-R(2*K)+EN(I,J)/DELT-ENERGY(I,J)/DELT-QH(I,J)              FEG 960
C                                                                       FEG 970
C     LINEARIZED NEWTON-RAPHSON MATRIX -                                FEG 980
      A(2*K-1,MP)=A(2*K-1,MP)-F(I,J)*DX(I)*DZ(I,J)*DY(J)/DELT           FEG 990
      A(2*K-1,MP+1)=A(2*K-1,MP+1)-G(I,J)*DX(I)*DY(J)*DZ(I,J)/DELT       FEG1000
      A(2*K,MP-1)=A(2*K,MP-1)-C(I,J)*DX(I)*DY(J)*DZ(I,J)/DELT           FEG1010
      A(2*K,MP)=A(2*K,MP)-D(I,J)*DX(I)*DY(J)*DZ(I,J)/DELT               FEG1020
   80 CONTINUE                                                          FEG1030
      RETURN                                                            FEG1040

      END                                                               FEG1050-
```

```
      SUBROUTINE SOLVE(KKK,B,R,NEQ,IHALFB,NDIM,MDIM)                    SOL  10
C     ************************************************                  SOL  20
      IMPLICIT REAL*8(A-H,O-Z)                                          SOL  30
C                                                                       SOL  40
C   ASYMMETRIC BAND MATRIX EQUATION SOLVER                              SOL  50
C     ORIGINALLY PROGRAMED BY JAMES C. DUGUID                           SOL  60
C                                                                       SOL  70
C   KKK=1 TRIANGULARIZES THE BAND MATRIX B                              SOL  80
C   KKK=2 SOLVES FOR RIGHT SIDE R, SOLUTION RETURNS IN R                SOL  90
C                                                                       SOL 100
      DIMENSION B(NDIM,MDIM), R(NDIM)                                   SOL 110
      NRS=NEQ-1                                                         SOL 120
      IHBP=IHALFB+1                                                     SOL 130
      IF (KKK.EQ.2) GO TO 30                                            SOL 140
C                                                                       SOL 150
C  TRIANGULARIZE MATRIX A USING DOOLITTLE METHOD                        SOL 160
C                                                                       SOL 170
      DO 20 K=1,NRS                                                     SOL 180
      PIVOT=B(K,IHBP)                                                   SOL 190
      KK=K+1                                                            SOL 200
      KC=IHBP                                                           SOL 210
      DO 10 I=KK,NEQ                                                    SOL 220
      KC=KC-1                                                           SOL 230
      IF (KC.LE.0) GO TO 20                                             SOL 240
      C=-B(1,KC)/PIVOT                                                  SOL 250
      B(1,KC)=C                                                         SOL 260
      KI=KC+1                                                           SOL 270
      LIM=KC+IHALFB                                                     SOL 280
      DO 10 J=KI,LIM                                                    SOL 290
      JC=IHBP+J-KC                                                      SOL 300
   10 B(I,J)=B(1,J)+C*B(K,JC)                                           SOL 310
   20 CONTINUE                                                          SOL 320
      GO TO 100                                                         SOL 330
C                                                                       SOL 340
C  MODIFY LOAD VECTOR R                                                 SOL 350
C                                                                       SOL 360
   30 NN=NEQ+1                                                          SOL 370
      IBAND=2*IHALFB+1                                                  SOL 380
      DO 70 I=2,NEQ                                                     SOL 390
      JC=IHBP-I+1                                                       SOL 400
      JI=1                                                              SOL 410
      IF (JC.LE.0) GO TO 40                                             SOL 420
      GO TO 50                                                          SOL 430
   40 JC=1                                                              SOL 440
      JI=I-IHBP+1                                                       SOL 450
   50 SUM=0.0                                                           SOL 460
      DO 60 J=JC,IHALFB                                                 SOL 470
      SUM=SUM+B(I,J)*R(JI)                                              SOL 480
   60 JI=JI+1                                                           SOL 490
   70 R(I)=R(I)+SUM                                                     SOL 500
C                                                                       SOL 510
C   BACK SOLUTION                                                       SOL 520
```

```
C
      R(NEQ)=R(NEQ)/B(NEQ,IHBP)                                         SOL  530
      DO 90 IBACK=2,NEQ                                                 SOL  540
      I=NN-IBACK                                                        SOL  550
      JP=I                                                              SOL  560
      KR=IHBP+1                                                         SOL  570
      MR=MIN0(IBAND,IHALFB+IBACK)                                       SOL  580
      SUM=0.0                                                           SOL  590
      DO 80 J=KR,MR                                                     SOL  600
      JP=JP+1                                                           SOL  610
   80 SUM=SUM+B(I,J)*R(JP)                                              SOL  620
   90 R(I)=(R(I)-SUM)/B(I,IHBP)                                         SOL  630
  100 RETURN                                                            SOL  640
      END                                                               SOL  650
                                                                        SOL  660-
```

```
      SUBROUTINE PDATA(P,H,TIME,L,NY,NX)                                PDT  10
C     **********************************                                PDT  20
      IMPLICIT REAL*8(A-H,O-Z)                                          PDT  30
C                                                                       PDT  40
C     CALLED FROM MAIN                                                  PDT  50
C     PURPOSE:  TO PRINT COMPUTED PRESSURE AND ENTHALPY                 PDT  60
C     ------------------------------------------------------------------PDT  70
      DIMENSION P(20,10), H(20,10)                                      PDT  80
C     ------------------------------------------------------------------PDT  90
C                                                                       PDT 100
      PRINT 30, L,TIME                                                  PDT 110
      DO 10 J=1,NY                                                      PDT 120
   10 PRINT 40, (P(I,J),I=1,NX)                                         PDT 130
      PRINT 50                                                          PDT 140
      DO 20 J=1,NY                                                      PDT 150
   20 PRINT 40, (H(I,J),I=1,NX)                                         PDT 160
      RETURN                                                            PDT 170
C                                                                       PDT 180
   30 FORMAT (///11X,'STEP NUMBER',I4,10X,'TIME',E10.3/11X,15(1H*)///11XPDT 190
     1,'PRESSURE VALUES'/11X,15(1H-)//)                                 PDT 200
   40 FORMAT (/(11X,8(G12.5,2X)))                                       PDT 210
   50 FORMAT (//11X,'ENTHALPY VALUES'/11X,15(1H-)//)                    PDT 220
      END                                                               PDT 230-
```

```
      SUBROUTINE PHREG(INDEX)                                                PHR  05
C     ***********************                                                PHR  10
      IMPLICIT REAL*8(A-H,O-Z)                                               PHR  20
C                                                                            PHR  25
C     CALLED FORM MAIN                                                       PHR  30
C     PURPOSE:  TO DETERMINE WHAT THERMODYNAMIC REGION A                     PHR  40
C               FINITE-DIFFERENCE BLOCK IS IN                                PHR  50
C     ---------------------------------------------------------------------PHR  60
      DIMENSION INDEX(200)                                                   PHR  70
C                                                                            PHR  80
      COMMON /INPUT/ PHI(20,10),XK(20,10),YK(20,10),P(20,10),H(20,10),X(PHR  90
     1400),X1(400),NP(20,10),NPP(200,2),DX(20),DY(10),DZ(20,10),Q(20,10)PHR 100
      COMMON /CONTRO/ NK,NX,NXX,NY,NYY,NB,NBB,NT,DELT,TIME,PHFWT,DF,XKC,PHR 110
     1COND,COEF,BETA,IPRT                                                   PHR 120
C     ---------------------------------------------------------------------PHR 130
C     INDEX=-1 FOR COMPRESSED WATER                                          PHR 140
C     INDEX=0  FOR TWO-PHASE                                                 PHR 150
C     INDEX=1  FOR SUPERHEATED STEAM                                         PHR 160
      DO 10 K=1,NB                                                           PHR 170
      PP=X1(2*K-1)*0.1**7                                                    PHR 180
      HH=X1(2*K)*0.1**7                                                      PHR 190
C     COMPUTE SATURATED STEAM (HS) AND SATURATED WATER (HW) ENTHALPY -       PHR 200
      HS=2822.82-39.952/PP+2.54342/PP/PP-0.938879*PP*PP                      PHR 210
      HW=730.984+129.239*PP-10.0333*PP*PP+0.39881*PP*PP*PP
     1   -99.0697/PP+12.9267/PP/PP-0.628359/PP/PP/PP
      INDEX(K)=0                                                             PHR 240
      IF (HH.LT.HW) INDEX(K)=-1                                              PHR 250
      IF (HH.GT.HS) INDEX(K)=1                                               PHR 260
C     IF TEMPERATURE IS BELOW 50 DEG C (H=209.33) ASSUME IN COMP WATER       PHR 270
      IF (HH.LT.209.33D0) INDEX(K)=-1                                        PHR 280
   10 CONTINUE                                                               PHR 290
      RETURN                                                                 PHR 300
      END                                                                    PHR 310-
```

```
      SUBROUTINE BALNCE(ICX,KKK,PCEE)                                   BAL  05
C     ******************************                                    BAL  10
      IMPLICIT REAL*8(A-H,O-Z)                                          BAL  20
C                                                                       BAL  30
C     CALLED FROM MAIN                                                  BAL  40
C     PURPOSE:  TO COMPUTE MASS AND ENERGY BALANCE                      BAL  50
C     ------------------------------------------------------------------BAL  60
      COMMON /INPUT/ PHI(20,10),XK(20,10),YK(20,10),P(20,10),H(20,10),X(BAL  70
     1400),X1(400),NP(20,10),NPP(200,2),DX(20),DY(10),DZ(20,10),Q(20,10)BAL  80
      COMMON /CONTRO/ NK,NX,NXX,NY,NYY,NB,NBB,NT,DELT,TIME,PHFWT,DF,XKC,BAL  90
     1COND,COEF,BETA,IPRT                                               BAL 100
      COMMON /WORK/ AX(21,10),BX(21,10),EX(21,10),AY(20,11),BY(20,11),EYBAL 110
     1(20,11),TX(21,10),TXK(21,10),TY(20,11),TYK(20,11),XM(20,10),XMASS(BAL 120
     220,10),EN(20,10),ENERGY(20,10),C(20,10),D(20,10),F(20,10),G(20,10)BAL 130
     3,QH(20,10),XN(20,10),TM(20,10),DTP(20,10),DTH(20,10),PHIO(20,10),PBAL 140
     4OLD(20,10)                                                        BAL 150
      COMMON /CHECK/ IND(200),INDOLD(200)                               BAL 160
C     ------------------------------------------------------------------BAL 170
C                                                                       BAL 180
C     STATEMENT FUNCTIONS FOR THERMODYNAMIC PROPERTIES -                BAL 190
C                                                                       BAL 200
      F1(PX,HX)=1.00207+4.42607D-4*PX-5.47456D-5*HX+5.02875D-7*HX*PX-1.2BAL 210
     14791D-7*HX*HX                                                     BAL 220
      F2(PX,HX)=-2.41231+2.56222D-1*HX-9.31415D-3*PX*PX-2.2568D-5*HX*HX BAL 230
      F3(PX,HX)=-2.26162D-5+0.043844ID0*PX-1.79088D-5*PX*HX+3.69276D-8*PBAL 240
     1X*PX*PX*PX+5.17644D-13*HX*HX*HX*PX                                BAL 250
      F4(PX,HX)=-374.669D0+47.9921D0*PX-0.633606D0*PX*PX+7.39386D-5*HX*HBAL 260
     1X-3.3372D6/HX/HX/PX/PX+0.0357154D0/PX/PX/PX-1.1725D-9*HX*HX*HX*PX-BAL 270
     22.26861D15/HX/HX/HX/HX                                            BAL 280
C                                                                       BAL 290
      DELM=0.D0                                                         BAL 300
      DELE=0.D0                                                         BAL 310
      DO 70 I=1,NX                                                      BAL 320
      DO 70 J=1,NY                                                      BAL 330
      IF (DZ(I,J).GT.0.D0) GO TO 10                                     BAL 340
      XM(I,J)=0.D0                                                      BAL 350
      EN(I,J)=0.D0                                                      BAL 360
      GO TO 70                                                          BAL 370
   10 CONTINUE                                                          BAL 380
      PP=P(I,J)                                                         BAL 390
      HH=H(I,J)                                                         BAL 400
      PP=PP*0.1**7                                                      BAL 410
      HH=HH*0.1**7                                                      BAL 420
      K=NP(I,J)                                                         BAL 430
      PHI(I,J)=PHIO(I,J)*(1.D0+(P(I,J)-POLD(I,J))*BETA)                 BAL 440
      HW=730.984+129.239*PP-10.0333*PP*PP+0.39881*PP*PP*PP
     1   -99.0697/PP+12.9267/PP/PP-0.628359/PP/PP/PP
      IF (IND(K)) 20,30,40                                              BAL 470
   20 CONTINUE                                                          BAL 480
      DEN=F1(PP,HH)                                                     BAL 490
      TEMP=F2(PP,HH)                                                    BAL 500
      GO TO 50                                                          BAL 510
```

```
   30 CONTINUE                                                          BAL 520
      HS=2822.82-39.952/PP+2.54342/PP/PP-0.938879*PP*PP                 BAL 530
      TEMP=F2(PP,HW)                                                    BAL 540
      DW=F1(PP,HW)                                                      BAL 550
      DS=F3(PP,HS)                                                      BAL 560
      SW=DS*(HS-HH)/(HH*(DW-DS)-(HW*DW-HS*DS))                          BAL 570
      SST=1.0-SW                                                        BAL 580
      DEN=DW*SW+SST*DS                                                  BAL 590
      GO TO 50                                                          BAL 600
   40 CONTINUE                                                          BAL 610
      DEN=F3(PP,HH)                                                     BAL 620
      HW=730.984+129.239*PP-10.0333*PP*PP+0.39881*PP*PP*PP
     1   -99.0697/PP+12.9267/PP/PP-0.628359/PP/PP/PP
      HS=2822.82-39.952/PP+2.54342/PP/PP-0.938879*PP*PP
      TT1=F4(PP,HH)
      TT2=F4(PP,HS)
      TT3=F2(PP,HW)
      TEMP=TT1-TT2+TT3
   50 HROCK=TEMP*PHFWT                                                  BAL 640
      VOL=DY(J)*DX(I)*DZ(I,J)                                           BAL 650
      XMT=VOL*DEN*PHI(I,J)                                              BAL 660
      ENT=VOL*(PHI(I,J)*HH*DEN*10.**7+(1.0-PHI(I,J))*HROCK*DF)          BAL 670
      IF (ICX.EQ.0) GO TO 60                                            BAL 680
      DELM=DELM-XMASS(I,J)+XMT                                          BAL 690
      DELE=DELE-ENERGY(I,J)+ENT                                         BAL 700
   60 XM(I,J)=XMT                                                       BAL 710
      EN(I,J)=ENT                                                       BAL 720
   70 CONTINUE                                                          BAL 730
      IF (KKK.NE.NK) GO TO 90                                           BAL 740
      DO 80 I=1,NX                                                      BAL 750
      DO 80 J=1,NY                                                      BAL 760
      XMASS(I,J)=XM(I,J)                                                BAL 770
   80 ENERGY(I,J)=EN(I,J)                                               BAL 780
   90 CONTINUE                                                          BAL 790
      IF (ICX.EQ.0) GO TO 110                                           BAL 800
      QQQ=0.D0                                                          BAL 810
      EEE=0.D0                                                          BAL 820
      DO 100 I=1,NX                                                     BAL 830
      DO 100 J=1,NY                                                     BAL 840
      QQQ=QQQ+Q(I,J)*DELT                                               BAL 850
  100 EEE=EEE+QH(I,J)*DELT                                              BAL 860
      PCEM=(DELM-QQQ)*100.D0                                            BAL 870
      IF (QQQ.NE.0.D0) PCEM=PCEM/QQQ                                    BAL 880
      PCEE=(DELE-EEE)*100.D0                                            BAL 890
      IF (EEE.NE.0.D0) PCEE=PCEE/EEE                                    BAL 900
      WRITE (6,120) QQQ,EEE,DELM,DELE,PCEM,PCEE                         BAL 910
  110 CONTINUE                                                          BAL 920
      RETURN                                                            BAL 930
C                                                                       BAL 940
  120 FORMAT (//11X,23HMASS AND ENERGY BALANCE/11X,23(1H-)/32X,4HMASS,31BAL 950
     1X,4HHEAT/15X,12HDISCHARGE = ,G15.8,20X,G15.8/15X,12HSTORAGE =    ,GBAL 960
     215.8,20X,G15.8/15X,12H% ERROR =    ,G15.8,20X,G15.8)             BAL 970
      END                                                               BAL 980
```