SIMULATION OF SOLUTE TRANSPORT IN VARIABLY SATURATED POROUS MEDIA WITH SUPPLEMENTAL INFORMATION ON MODIFICATIONS TO THE U.S. GEOLOGICAL SURVEY'S COMPUTER PROGRAM VS2D

By R.W. Healy

U.S. Geological Survey

Water-Resources Investigations Report 90-4025

Denver, Colorado
1990

DEPARTMENT OF THE INTERIOR

MANUEL LUJAN, JR., Secretary

U.S. GEOLOGICAL SURVEY

Dallas L. Peck, Director

For additional information
write to:

Chief, Branch of Regional Research
U.S. Geological Survey
Box 25046, Mail Stop 418
Federal Center
Denver, CO 80225-0046

Copies of this report can
be purchased from:

U.S. Geological Survey
Books and Open-File Reports Section
Box 25425
Federal Center
Denver, CO 80225-0425

CONTENTS

FIGURES

TABLES

CONVERSION FACTORS

Metric (International System) units in this report may be converted to inch-pound units by the following conversion factors:

Multiply SI units	*By*	*To obtain inch-pound units*
centimeter (cm)	0.3937	inch
centimeter per cubic centimeter (cm/cm^3)	6.542	inch per cubic inch
centimeter per hour (cm/h)	0.3937	inch per hour
centimeter per second (cm/s)	0.03281	foot per second
cubic meter per hour (m^3/h)	35.32	cubic foot per hour
gram (gm)	0.002205	pound
kilopascal (kPa)	0.01450	pound per square inch
liter per hour (L/h)	0.2642	gallon per hour
meter (m)	3.281	foot
meter per hour (m/h)	3.281	foot per hour
meter per second (m/s)	3.201	foot per second
millimeter (mm)	0.03937	inch

SIMULATION OF SOLUTE TRANSPORT IN VARIABLY SATURATED POROUS MEDIA WITH SUPPLEMENTAL INFORMATION ON MODIFICATION TO THE U.S. GEOLOGICAL SURVEY'S COMPUTER PROGRAM VS2D

By R.W. Healy

ABSTRACT

This report documents computer program VS2DT for solving problems of solute transport in variably saturated porous media. The program uses a finite-difference approximation to the advection-dispersion equation. The program is an extension to the computer program VS2D developed by the U.S. Geological Survey, which simulates water movement through variably saturated porous media. Simulated regions can be one-dimensional columns, two-dimensional vertical cross sections, or axially symmetric, three-dimensional cylinders. Program options include: backward or centered approximations for both space and time derivatives, first-order decay, equilibrium adsorption as described by Freundlich or Langmuir isotherms, and ion exchange. Five test problems are used to demonstrate the ability of the computer program to accurately match analytical and previously published simulation results. Additional modifications to computer program VS2D are included as supplemental information.

The computer program is written in standard FORTRAN77. Extensive use of subroutines and function subprograms provides a modular code that can be easily modified for particular applications. A complete listing of data-input requirements and input and output for an example problem are included.

INTRODUCTION

Operations conducted at land surface or within the unsaturated zone may have considerable impact on the quality and quantity of water reaching local ground water reservoirs. Some of the more important of these operations include application of agricultural chemicals, solid-waste disposal, hazardous and radioactive-waste disposal, use of septic tanks, and accidental chemical spills. Understanding the fate of dissolved chemicals within the unsaturated zone can greatly aid in the prediction of the chemistry of the water that reaches aquifers. Such an understanding would also allow for evaluation of different preventative or remedial actions designed to protect our valuable ground-water resources. Computer models of water and solute movement within variably saturated porous media can be useful tools for gaining insight to processes that occur within the unsaturated zone. Computer models are a cost-effective means for predicting the effects of modifications to, or perturbations of, the unsaturated-zone system on the water contained in that system. Through a simple sensitivity analysis, the relative importance of different parameters that affect flow and transport can be investigated.

This report describes computer program VS2DT that simulates solute transport in porous media under variably saturated conditions. The program is an extension to the U.S. Geological Survey's computer program VS2D (Lappala and others, 1987), which simulates water movement through variably saturated porous media. The extension consists of four new subroutines and slight modifications to existing routines. VS2DT may be a useful tool in studies of water quality, ground-water contamination, waste disposal, or ground-water recharge. The program is user oriented and easy to use. However, its use must be accompanied by an awareness of the assumptions and limitations inherent in its development. This report describes theory and numerical implementation of the solute transport model. Details on simulation of water flow are contained in Lappala and others (1987), therefore little additional information on this topic is included in this report. Potential users of VS2DT should obtain a copy of Lappala and others (1987). The program is verified by comparing results to analytical solutions and previously published simulation results. Detailed description of data-input requirements and program structure are also included. Some additional modifications to computer program VS2D are presented as supplemental information.

Computer program VS2DT uses a finite-difference approximation to the advection-dispersion equation as well as the nonlinear water-flow equation (based on total hydraulic head). It can simulate problems in one, two (vertical cross section), or three dimensions (axially symmetric). The porous media may be heterogeneous and anisotropic, but principal directions must coincide with the coordinate axes. Boundary conditions for flow can take the form of fixed pressure heads, infiltration with ponding, evaporation from the soil surface, plant transpiration, or seepage faces. An extension to the program (Healy, 1987) also allows simulation of infiltration from trickle irrigation. Boundary conditions for solute transport include fixed solute concentration and fixed mass flux. Solute source/sink terms include first-order decay, equilibrium partitioning to the solid phase (as described by Langmuir or Freundlich isotherms), and ion exchange. The design of the program is modular, so that programmers can easily modify subroutines and functions in order to apply the model to particular field, laboratory, or hypothetical problems.

THEORY OF SOLUTE TRANSPORT IN VARIABLY SATURATED POROUS MEDIA

For purposes of this report solute transport is assumed to be described by the advection-dispersion equation. Derivation of that equation is based on mass conservation and Fick's law. Details of the derivation are beyond the scope of this report, but are contained in texts such as Bear (1979) or Hillel (1980).

Three mechanisms affect the movement of solutes under variably saturated conditions: (1) advective transport, in which solutes are moving with the flowing water; (2) hydrodynamic dispersion, in which molecular diffusion and variability of fluid velocity cause a spreading of solutes about the average direction of water flow; and (3) sources and sinks--including fluid sources, where a water of a specified chemical concentration is introduced to water of a different concentration, and chemical reactions such as radioactive decay or

adsorption to the solid phase. The advection-dispersion equation that describes solute transport under variably saturated conditions can be written as (Bear, 1979, p. 251):

$$\frac{\partial(\theta c)}{\partial t} = \nabla\cdot\theta\ \bar{\bar{D}}_h\cdot\nabla c - \nabla\cdot\theta\bar{v}c + SS \qquad (1)$$

where θ = volumetric moisture content, dimensionless;
c = concentration of chemical constituent, ML^{-3} (mass per unit volume of water);
t = time, T;
∇ = del operator = $\frac{\partial}{\partial x} + \frac{\partial}{\partial y} + \frac{\partial}{\partial z}$, L^{-1};
$\bar{\bar{D}}_h$ = hydrodynamic dispersion tensor, L^2T^{-1};
$\bar{v}$ = fluid velocity vector, LT^{-1}; and
SS = source/sink terms, $ML^{-3}T^{-1}$.

Advection

The second term in the right hand side of equation 1 represents the divergence of the advective flux. This term accounts for changes in solute concentrations due to water moving and carrying solute with it. A simple one-dimensional experiment is shown in figure 1a to illustrate the advective and dispersive components of solute transport. In the experiment, a steady downward flow of solute-free water is obtained through a vertical column. At time t_0 the solute concentration is instantaneously increased to C_0 and maintained at that concentration throughout the remainder of the experiment. Relative concentration of the column outflow over time (commonly called a breakthrough curve) is shown in figure 1c. If advection is the only driving force for transport, then the tracer will move through the column as a plug and the breakthrough curve will simply be a step function, as shown by the dashed line in figure 1c.

Hydrodynamic Dispersion

The first term on the right-hand side of equation 1 represents the divergence of the flux of chemicals due to hydrodynamic dispersion. Hydrodynamic dispersion refers to a spreading process whereby molecules of a solute gradually move in directions different from that of the average ground-water flow. This spreading process is illustrated in the previously described experiment by the solid line in figure 1c. The theory behind dispersion has been reviewed extensively in the literature (see, for example, Bear, 1972, 1979; Scheidegger, 1961; Konikow and Grove, 1977). Two mechanisms comprise this phenomenon. The first is called mechanical dispersion and is caused by variations in the velocity field at the microscopic level. These variations are related to the tortuous nature of flow paths through porous media and the differences in velocity that occur across a single pore. Flow paths are not straight, but must follow the pores (fig. 2). Therefore molecules of solute will also be carried through these paths.

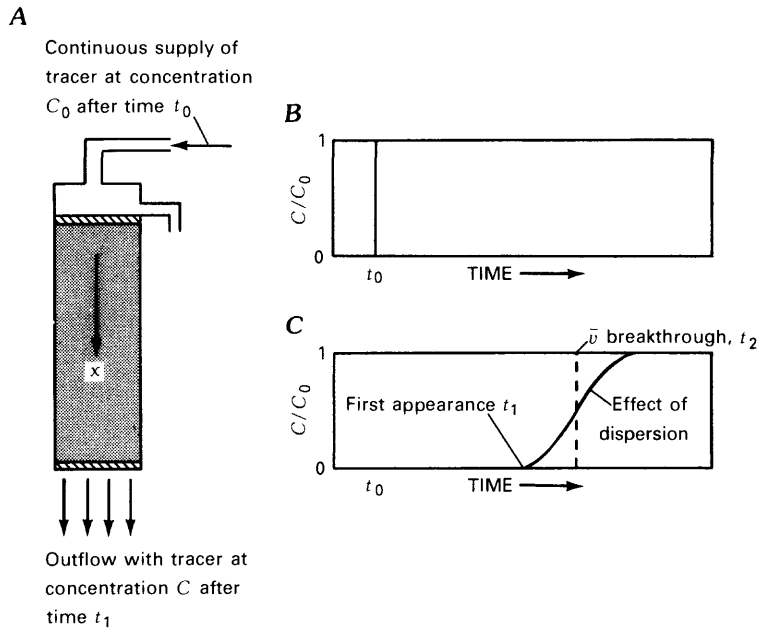


Figure 1.--Diagram showing effects of advection and dispersion of a tracer through a column of porous media: *A*) Column with steady flow and continuous supply of tracer after time t_0; *B*) step-function-type tracer input relation; *C*) relative tracer concentration in outflow from column (dashed line indicates plug flow condition and solid line illustrates effect of mechanical dispersion and molecular diffusion). Reproduced from Freeze and Cherry (1979, p. 390) and published with permission.

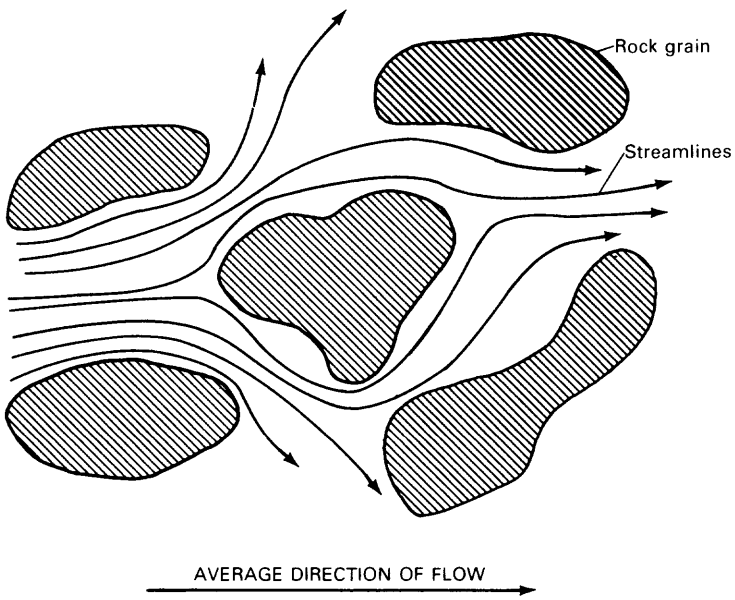


Figure 2.--Diagram showing spreading of flow paths.

The second mechanism contributing to hydrodynamic dispersion is molecular diffusion, which results from variations in solute concentrations. In the absence of water flow, molecules of solute will move from areas of high concentration to areas of low concentrations, in an effort to equalize concentrations everywhere. This mechanism also works when velocities are nonzero, causing lateral solute movement across streamtubes.

Following Bear (1979, p. 238) we can write the hydrodynamic dispersion tensor as the sum of tensors of mechanical dispersion ($\bar{\bar{D}}$) and molecular diffusion ($\bar{\bar{D}}_m$):

$$\bar{\bar{D}}_h = \bar{\bar{D}} + \bar{\bar{D}}_m \tag{2}$$

$$D_{ij} = \alpha_T|v|\delta_{ij} + (\alpha_L-\alpha_T)v_i v_j/|v| \tag{3}$$

$$D_{m_{ij}} = D_d\tau_{ij} \tag{4}$$

where α_T = transverse dispersivity of the porous medium, L;
$|v|$ = magnitude of the velocity vector, LT^{-1};
δ_{ij} = Kronecker delta, dimensionless
= 1 if $i = j$
= 0 if $i \neq j$;
α_L = longitudinal dispersivity of the porous media, L;
v_i = $i^{\underline{th}}$ component of the velocity vector, LT^{-1};
D_d = coefficient of molecular diffusion of solute in water, L^2T^{-1}; and
τ_{ij} = tortuosity, dimensionless.

In saturated porous media, dispersivity is theoretically a property of the geometry of the solid matrix. However, experimental data show a large scale effect, with dispersivities at the lab scale typically on the order of centimeters but at the field scale being on the order of several meters. There also is some question as to whether dispersivity varies as a function of moisture content in unsaturated porous media. In VS2DT, α_L and α_T are treated as constants. For this report, it is assumed that tortuosity is constant and uniformly aligned with the x and z axes so that $\tau_{xx} = \tau_{zz} = \tau$ and $\tau_{xz} = \tau_{zx} = 0$. Then, setting $D_m = D_d\tau$, we have $D_{m_{xx}} = D_{m_{zz}} = D_m$; $D_{m_{xz}} = D_{m_{zx}} = 0$. Therefore, the components of the two-dimensional hydrodynamic dispersion tensor can be written as:

$$D_{h_{xx}} = \alpha_L\frac{v_x^2}{|v|} + \alpha_T\frac{v_z^2}{|v|} + D_m \tag{5}$$

$$D_{h_{zz}} = \alpha_L\frac{v_z^2}{|v|} + \alpha_T\frac{v_x^2}{|v|} + D_m \tag{6}$$

$$D_{h_{zx}} = D_{h_{xz}} = (\alpha_L - \alpha_T)\, v_x v_z / |v| . \qquad (7)$$

Source/Sink Terms

Source/sink terms can be divided into 2 general categories: solute mass introduced to or removed from the domain by fluid sources and sinks; and mass introduced or removed by chemical reactions occurring within the water or between the water and the solid phase.

Fluid Sources and Sinks

Mathematically, the first category of source/sink terms can be represented by:

$$SS = c^* q \qquad (8)$$

where c^* = mass concentration in fluid source/sink, ML^{-3};
q = strength of fluid source/sink, T^{-1}.
When $q > 0$ (flow is into the system), c^* must be specified by the user.
When $q < 0$ (flow is out of the system), c^* is set equal to the ambient solute concentration at the location where flow is leaving the system, that is:

$$c^* = c.$$

Decay, Adsorption, and Ion Exchange

For the second category of Source/Sink Terms three types of reactions may be simulated by the program. The first is a linear decay of the solute (such as radioactive decay). This is described by:

$$SS = \lambda \theta c \qquad (9)$$

where λ = the decay constant, T^{-1}.

The second type of reaction that may be simulated with VS2DT is sorption of solute from the water phase to the solid phase through physical or chemical attraction. Sorption may actually be a very complex process, but it is treated simplistically in VS2DT. Since the movement of water in soils is often slow relative to the rate of adsorption, it is assumed, for purposes of this computer program, that adsorption is equilibrium controlled. Therefore, the rate of change of solute mass in the sorped state is given by:

$$SS = \frac{\partial \rho_b \tilde{c}}{\partial t} = \rho_b \frac{\partial \tilde{c}}{\partial c} \frac{\partial c}{\partial t} \qquad (10)$$

where $\tilde{c}$ = concentration of solute mass in solid phase, MM^{-1};
ρ_b = bulk density of solid phase, ML^{-3}.

Experimental data are usually used to describe the relation between c and $\tilde{c}$. Plots of $\tilde{c}$ as a function of c at constant temperature are called isotherms. Often, empirically derived formulae are fit to these isotherms. Two such formulae may be used in VS2DT--the Freundlich or the Langmuir isotherm.

The Freundlich isotherm is given by:

$$\tilde{c} = K_f c^n \tag{11}$$

$$\frac{\partial \tilde{c}}{\partial c} = n K_f c^{n-1} \tag{12}$$

where K_f = Freundlich adsorption constant, and

n = Freundlich exponent.

Typical Freundlich isotherms are shown in figure 3. These isotherms are characterized by an unlimited capacity of the solid to adsorb the solute. A special case of the Freundlich isotherm occurs when n = 1. This produces a linear isotherm:

$$\tilde{c} = K_d c \tag{13}$$

$$\frac{\partial \tilde{c}}{\partial c} = K_d \tag{13a}$$

where K_d = equilibrium distribution coefficient, L^3M^{-1}.

Linear isotherms are shown in figure 3. Because of its simplicity, the linear isotherm is probably the most widely used isotherm in solute-transport simulations. For nonionic organic compounds K_d primarily represents adsorption to organic matter in soils. Since organic content of soils can vary greatly among and within individual soil types, the following equation is commonly used to approximate K_d (Jury and others, 1983):

$$K_d = f_{oc} K_{oc} \tag{14}$$

where f_{oc} = fraction of organic carbon in soil, MM^{-1}; and

K_{oc} = organic carbon distribution coefficient, L^3M^{-1}.

This approximation requires knowledge of f_{oc} instead of K_d; f_{oc} is much easier to measure than K_d. Several authors have reported correlations between K_{oc} and K_{ow}, the octanol-water partition coefficient (Karickhoff, 1981; Chiou and others, 1983). Rao and Davidson (1980) developed the following equation:

$$\log(K_{oc}/1000) = 1.029 \log(K_{ow}/1000) - 0.18 \tag{15}$$

where K_{oc} and K_{ow} are in m^3Kg^{-1}.

Values of K_{ow} may be obtained in standard indices such as Corwin and Hansch (1979).

The Langmuir isotherm is given by:

$$\tilde{c} = \frac{K_1 Q c}{1 + K_1 c} , \tag{16}$$

$$\frac{\partial \tilde{c}}{\partial c} = \frac{K_1 Q}{(1 + K_1 c)^2} \tag{16a}$$

where K_1 = Langmuir adsorption constant, L^3M^{-1}; and
Q = maximum number of adsorption sites.

Langmuir isotherms are characterized by a fixed number of adsorption sites. Figure 3 shows example Langmuir isotherms.

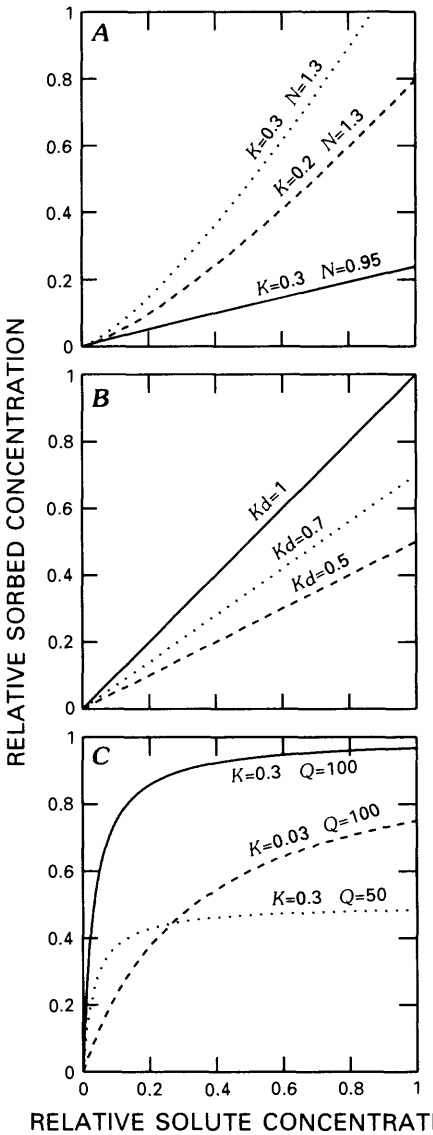


Figure 3.--Graph showing examples of isotherms: A) Freundlich; B) Linear; and C) Langmuir.

The third type of reaction is ion exchange, which is described by:

$$mc_1^n + n\tilde{c}_2 \rightleftarrows m\tilde{c}_1 + nc_2^m \tag{17}$$

where n is the valence for ion 1, and
m is the valence for ion 2.

The rate of change of ion concentration of solute mass in the solid phase can again be represented by equation 10. Four types of exchange are permitted in VS2DT, monovalent-monovalent exchange (m=n=1), divalent-divalent exchange (m=n=2), monovalent-divalent exchange (m=2, n=1), and divalent-monovalent exchange (m=1, n=2). The ion-exchange selectivity coefficient (K_m) is defined as:

$$K_m = \begin{cases} \dfrac{\tilde{c}_1 c_2}{\tilde{c}_2 c_1} & , \text{ if } m = n, \\ \dfrac{\tilde{c}_1^m c_2^n}{\tilde{c}_2^n c_1^m} & , \text{ if } m \neq n. \end{cases} \tag{18}$$

If only two ions are involved and C_o and $\hat{Q}$ are constant, where C_o is the total-solution concentration for ions 1 and 2, in terms of equivalents per volume; and $\hat{Q}$ is the ion-exchange capacity, in terms of equivalents per mass; then:

$$nc_1 + mc_2 = C_o \tag{19}$$

$$n\tilde{c}_1 + m\tilde{c}_2 = \hat{Q}. \tag{20}$$

By combining equations 18, 19, and 20, the second component in the exchange process can be eliminated. For monovalent-monovalent exchange (such as the exchange of sodium and potassium) the following equations are produced:

$$\tilde{c} = \frac{K_m \hat{Q} c}{c(K_m - 1) + C_o} \tag{21}$$

$$\frac{\partial \tilde{c}}{\partial c} = \frac{K_m \hat{Q} C_o}{[c(K_m - 1) + C_o]^2} . \tag{21a}$$

Divalent-divalent exchange (such as the exchange of calcium and strontium) is described by:

$$\tilde{c} = \frac{K_m \hat{Q} c}{2c(K_m - 1) + C_o} \tag{22}$$

$$\frac{\partial \tilde{c}}{\partial c} = \frac{K_m \hat{Q} C_o}{[2c(K_m-1) + C_o]^2} \tag{22a}$$

An example of monovalent-divalent exchange is the exchange of sodium with calcium. The following equations are produced for this exchange:

$$\tilde{c}^2(Co-c) + \tilde{c}K_m c^2 - c^2\hat{Q}K_m = 0 \tag{23}$$

$$\frac{\partial \tilde{c}}{\partial c} = \frac{\tilde{c}^2 - \tilde{c}2K_m c + 2c\hat{Q}K_m}{(Co-c)2\tilde{c} + K_m c^2} . \tag{23a}$$

In order to solve equation 23a, equation 23 must first be solved for $\tilde{c}$ by the quadratic formula.

Divalent-monovalent exchange (such as calcium-sodium exchange) is described by

$$\tilde{c}^2 4cK_m + \tilde{c}(-4c\hat{Q}K_m-(Co-2c)^2) + K_m c\hat{Q}^2 = 0 \tag{24}$$

$$\frac{\partial \tilde{c}}{\partial c} = \frac{-\tilde{c}^2 4K_m + \tilde{c}4(\hat{Q}K_m-(Co-2c)) - K_m\hat{Q}^2}{4cK_m(2\tilde{c}-\hat{Q}) - (Co-2c)^2} . \tag{24a}$$

Again, equation 24, which is quadratic in $\tilde{c}$, must be solved prior to solving equation 24a.

Additional information concerning the chemistry of adsorption and ion exchange can be found in texts such as Freeze and Cherry (1979) and Stumm and Morgan (1981). Bear (1972) and Grove and Stollenwerk (1984) present additional details on incorporating adsorption and ion-exchange into ground-water solute transport models.

Selection of adsorption or ion exchange must be made by the user at the time the computer program is compiled by selecting the appropriate version of the subroutine function VTRET. All other versions of that routine must be removed from the program or commented out. If ion exchange is selected, the user must take care to use consistent units for all variables. Ion exchange and adsorption cannot be simulated at the same time.

Boundary Conditions

The distinction between boundary conditions and source/sink terms is somewhat artificial; therefore, this discussion overlaps that in the previous section. Two types of boundaries may be specified for solute transport simulations: fixed concentration and fixed mass flux of solute. In addition, when fluid boundary conditions are such that water flow is into the system then the concentration of the water entering the system also must be specified.

When fluid boundary conditions are such that water flow is out of the system then the program assumes that the concentration of that water is identical to that in the finite-difference cell where the water is departing. An exception to this rule is removal of water from the system by evaporation. That water is assumed to be solute free.

Equation 1 can now be rewritten, assuming linear adsorption and noting that decay of solute mass in the solid phase also must be accounted for, as:

$$\frac{\partial}{\partial t}(\theta+\rho_b K_d)c = \nabla\cdot\theta\bar{\bar{D}}_h\cdot\nabla c - \nabla\cdot\theta\bar{v}c - \lambda(\theta+\rho_b K_d)c + c^*q \ . \qquad (25)$$

NUMERICAL IMPLEMENTATION

Following the derivation of the finite difference approximation for the fluid flow equation (Lappala and others, 1987), let us look at the conservation of mass for a finite-difference cell of volume V and surface area $\hat{S}$ (fig. 4). We have

$$\int_V \frac{\partial(\theta+\rho_b K_d)c}{\partial t}\,dV = \int_V \nabla\cdot\theta\bar{\bar{D}}_h\cdot\nabla c\,dV - \int_V \nabla\cdot\bar{v}\theta c\,dV - \int_V \lambda(\theta+\rho_b K_d)c\,dV + \int_V c^*q\,dV \qquad (26)$$

We can use the Gauss divergence theorem to transform the first two volume integrals on the right-hand side to surface integrals

$$\int_V \nabla\cdot\theta\bar{\bar{D}}_h\cdot\nabla c\,dV = \int_{\hat{S}} \theta\bar{\bar{D}}_h\cdot\nabla c\cdot\bar{n}\,d\hat{S} \qquad (27)$$

$$\int_V \nabla\cdot\bar{v}\theta c\,dV = \int_{\hat{S}} \bar{v}\theta c\cdot\bar{n}\,d\hat{S} \qquad (28)$$

where $\bar{n}$ is the outward normal unit vector.

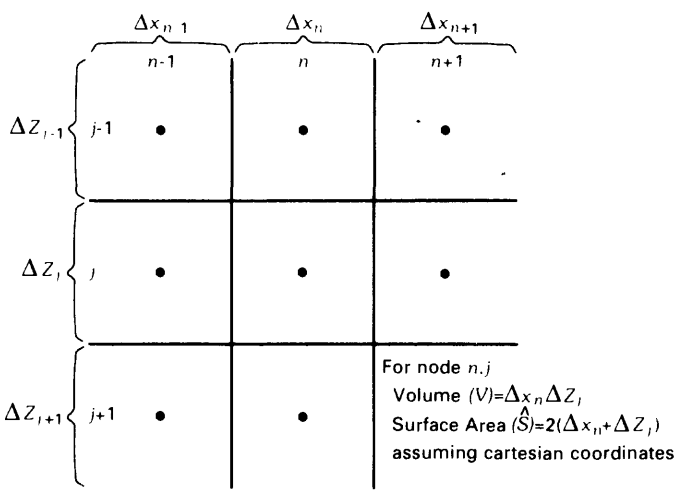


Figure 4.--Sketch showing finite-difference grid.

It is assumed that the volume V is small enough that within V the moisture content, bulk density, equilibrium distribution coefficient, and concentration can be considered constant, so that:

$$\int_V \frac{\partial(\theta+\rho_b K d)c}{\partial t} = V \frac{\partial(\theta+\rho_b K d)c}{\partial t} \tag{29}$$

$$\int_V \lambda(\theta+\rho_b K_d)c\,dV = V\lambda(\theta+\rho_b K_d)c; \tag{30}$$

$$\int_V c^* q\,dV = c^* qV = c^* q^* \tag{31}$$

where $q^* = qV$ = volumetric fluid flux, L^3T^{-1}.

We then have

$$V \frac{\partial(\theta+\rho_b K d)c}{\partial t} = \int_{\hat{S}} \theta\bar{\bar{D}}_h \cdot \nabla c \cdot \bar{n}\, d\hat{S} - \int_{\hat{S}} \bar{v}\theta c \cdot \bar{n} d\hat{S} - V\lambda(\theta+\rho_b K_d)c + c^* q^* \,. \tag{32}$$

Spatial Discretization

The integral describing dispersive flux in equation 32 can be approximated by realizing that the surface of the finite difference cell contains four active faces (this is because of the assumption of two-dimensional flow; if three-dimensional flow were to be considered, then the number of faces would be 6). Referring to figure 4, we can write:

$$\int_{\hat{S}} \theta\bar{\bar{D}}_h \cdot \nabla c \cdot \bar{n}\, d\hat{S} = \sum_{\ell=1}^{4} \int_{\hat{S}_\ell} \theta\bar{\bar{D}}_h \cdot \nabla c \cdot \bar{n}\, d\hat{S}_\ell \tag{33}$$

$$\begin{aligned} \approx &-\left[A\theta\left(D_{h_{xx}} \frac{\partial c}{\partial x} + D_{h_{xz}} \frac{\partial c}{\partial z}\right)\right]_{n-1/2,j} + \left[A\theta\left(D_{h_{xx}} \frac{\partial c}{\partial x} + D_{h_{xz}} \frac{\partial c}{\partial z}\right)\right]_{n+1/2,j} \\ &- \left[A\theta\left(D_{h_{zz}} \frac{\partial c}{\partial z} + D_{h_{zx}} \frac{\partial c}{\partial x}\right)\right]_{n,j-1/2} + \left[A\theta\left(D_{h_{zz}} \frac{\partial c}{\partial z} + D_{h_{zx}} \frac{\partial c}{\partial x}\right)\right]_{n,j+1/2} \end{aligned} \tag{34}$$

where ℓ = index to faces of cell n,j;
n = nodal index in x direction;
j = nodal index in z direction;
$n\pm1/2$, $j\pm1/2$ = indices to boundary faces of cell n,j;
A = surface area of cell face normal to flux direction, L^2; and directions are positive from left to right and top to bottom.

Terms along cell boundaries that appear in equation 33 are evaluated in the following manner:

$$\theta_{n-1/2,j} = \frac{1}{2}(\theta_{n-1,j} + \theta_{n,j}) \tag{35}$$

$$\left.\begin{aligned} A_{n-1/2,j} &= A_{n+1/2,j} = \Delta z_j \\ A_{n,j-1/2} &= A_{n,j+1/2} = \Delta x_n \end{aligned}\right\}$$ Note: These equations are for cartesian coordinates. For radial coordinates the areas are given in Lappala and others (1987).

$$\frac{\partial c}{\partial x}_{n-1/2,j} = \frac{c_{n,j} - c_{n-1,j}}{1/2(\Delta x_{n-1} + \Delta x_n)} \tag{36}$$

$$\frac{\partial c}{\partial z}_{n-1/2,j} = 1/2 \frac{c_{n,j+1} + c_{n-1,j+1} - c_{n,j-1} - c_{n-1,j-1}}{\Delta z_j + 1/2(\Delta z_{j-1} + \Delta z_{j-1})} \tag{37}$$

Δz_j = height of finite-difference cells in row j, L; and

Δx_n = width of finite-difference cells in column n, L.

Spatial discretization of the advective component in equation 32 can be accomplished with either central or backward differencing. The integral representing the advective flux can be approximated by:

$$\int_{\hat{S}} \bar{v}\theta c \cdot \bar{n} d\hat{S} = \sum_{\ell=1}^{4} \int_{\hat{S}_\ell} \bar{v}\theta c \cdot \bar{n} d\hat{S} \tag{38}$$

$$= -[A\theta v_x c]_{n-1/2,j} + [A\theta v_x c]_{n+1/2,j} - [A\theta v_z c]_{n,j-1/2} + [A\theta v_z c]_{n,j+1/2} \tag{39}$$

where

$v_{x_{n-1/2,j}}$ = velocity in x direction at n-1/2,j, positive from left to right;

$$= -\left[\frac{K_r(h)K}{\theta} \frac{\partial H}{\partial x}\right]_{n-1/2,j} \tag{40}$$

$$= \left[\frac{K_r(h)K}{\theta}\right]_{n-1/2,j} \frac{H_{n-1,j} - H_{n,j}}{1/2(\Delta x_n + \Delta x_{n-1})} \tag{41}$$

$v_{z_{n,j-1/2}}$ = velocity in z direction at n,j-1/2, positive from top to bottom;

H = total hydraulic head, L;
= h - z;
h = pressure head, L;
$K_r(h)$ = relative hydraulic conductivity, dimensionless; and
K = saturated hydraulic conductivity, LT^{-1}.

$$c_{n-1/2,j} = \begin{cases} 1/2(c_{n,j}+c_{n-1,j}), & \text{if central differencing in space is specified by the user;} \\ c_{n-1,j}, & \text{if backward differencing in space is specified and } v_{x_{n-1/2,j}} > 0; \\ c_{n,j}, & \text{if backward differencing in space is specified and } v_{x_{n-1/2,j}} < 0. \end{cases}$$

Temporal Discretization

The time derivative in equation 32 can be approximated by two different methods in the program. Either a fully backward-in-time (fully implicit) or a centered-in-time (Crank-Nicholson) approximation may be selected by the user. For either method we can write

$$\frac{\partial}{\partial t}(\theta+\rho_b K_d)c = c\frac{\partial \theta}{\partial t} + (\theta+\rho_b K_d)\frac{\partial c}{\partial t} \tag{42}$$

$$\approx c^{i+1/2}\frac{\theta^{i+1}-\theta^{i}}{\Delta t} + (\theta^{i+1/2}+\rho_b K_d)\frac{c^{i+1}-c^{i}}{\Delta t} \tag{43}$$

where i = index for previous time step;
i+1 = index for current time step;
Δt = length of the $i+1^{st}$ time step, T;
$c^{i+1/2}$ is assumed to be equal to c^{i+1}; and
$\theta^{i+1/2}$ is assumed to be equal to θ^{i+1}.

For the fully implicit formulation, concentrations on the right-hand side of equation 32 are all evaluated at the i+1 time level. For the time centered formulation, the terms on the right-hand side are evaluated as the average between the current time step and the previous time step. The time centered scheme is more accurate than the fully implicit scheme. It is second order correct in Δt while the fully implicit method is first order correct in Δt. However, as will be discussed later, for some problems the fully implicit methods may have some advantages. The final finite-difference form for equation 26 can now be written as:

$$\hat{A}^{i+1}c^{i+1}_{n-1,j} + \hat{B}^{i+1}c^{i+1}_{n,j-1} + \hat{C}^{i+1}c^{i+1}_{n+1,j} + \hat{D}^{i+1}c^{i+1}_{n,j+1} + \hat{E}^{i+1}c^{i+1}_{n,j} = RHS \quad (44)$$

$$\hat{A}^{i+1} = TC\left[(A\theta)_{n-1/2,j}\left[\frac{D_{h_{xx\ n-1/2,j}}}{1/2(\Delta x_n+\Delta x_{n-1})} + \frac{1}{2}v_{x_{n-1/2,j}}\right] + \hat{G} - \hat{H}\right]^{i+1} \quad (45a)$$

$$\hat{B}^{i+1} = TC\left[(A\theta)_{n,j-1/2}\left[\frac{D_{h_{zz\ n,j-1/2}}}{1/2(\Delta z_j+\Delta z_{j-1})} + \frac{1}{2}v_{z_{n,j-1/2}}\right] + \hat{F} - \hat{I}\right]^{i+1} \quad (45b)$$

$$\hat{C}^{i+1} = TC\left[(A\theta)_{n+1/2,j}\left[\frac{D_{h_{xx\ n+1/2,j}}}{1/2(\Delta x_n+\Delta x_{n+1})} - \frac{1}{2}v_{x_{n+1/2,j}}\right] - \hat{G} + \hat{H}\right]^{i+1} \quad (45c)$$

$$\hat{D}^{i+1} = TC\left[(A\theta)_{n,j+1/2}\left[\frac{D_{h_{zz\ n,j+1/2}}}{1/2(\Delta z_j+\Delta z_{j+1})} - \frac{1}{2}v_{z_{n,j+1/2}}\right] - \hat{F} + \hat{I}\right]^{i+1} \quad (45d)$$

$$\hat{E} = -\hat{A}-\hat{B}-\hat{C}-\hat{D} + TC\left[(A\theta v_x)_{n-1/2,j} + (A\theta v_z)_{n,j-1/2} - (A\theta v_x)_{n+1/2,j} - (A\theta v_z)_{n,j+1/2}\right]$$

$$- \frac{V}{\Delta t}(2\theta^{i+1}_{n,j}+\rho_b K_d-\theta^{i}_{n,j}) - V\lambda(\theta^{i+1}_{n,j}+\rho_b K_d) \quad (45e)$$

$$RHS = -\frac{V}{\Delta t}c^{i}_{n,j}(\theta^{i+1}_{n,j}+\rho_b K_d) - 2\ (1-TC)x[\hat{A}^{i}c^{i}_{n-1,j}+\hat{B}^{i}c^{i}_{n,j-1}+\hat{C}^{i}c^{i}_{n+1,j}+\hat{D}^{i}c^{i}_{n,j+1}+$$

$$\hat{E}^{i}c^{i}_{n,j}] - (\hat{F}^{i+1}+\hat{G}^{i+1})c^{i+1}_{n-1,j-1} + (\hat{H}^{i+1}+\hat{F}^{i+1})c^{i+1}_{n-1,j+1} +$$

$$(\hat{I}^{i+1}+\hat{G}^{i+1})c^{i+1}_{n+1,j-1} - (\hat{H}^{i+1}+\hat{I}^{i+1})c^{i+1}_{n+1,j+1} \quad (45f)$$

where

$$\hat{F} = \frac{1}{2} \frac{(A\theta D_{h_{xz}})_{n-1/2,j}}{\Delta z_j + 1/2(\Delta z_{j+1}+\Delta z_{j-1})}$$

$$\hat{G} = \frac{1}{2} \frac{(A\theta D_{h_{zx}})_{n,j-1/2}}{\Delta x_n + 1/2(\Delta x_{n-1}+\Delta x_{n+1})}$$

$$\hat{H} = \frac{1}{2} \frac{(A\theta D_{h_{zx}})_{n,j+1/2}}{\Delta x_n + 1/2(\Delta x_{n-1}+\Delta x_{n+1})}$$

$$\hat{I} = \frac{1}{2} \frac{(A\theta D_{h_{xz}})_{n+1/2,j}}{\Delta z_j + 1/2(\Delta z_{j-1}+\Delta z_{j+1})}$$

$$TC = \begin{cases} 1 & \text{, fully implicit; and} \\ 1/2 & \text{, time centered.} \end{cases}$$

The formulations given in equations 45 are based on central-difference approximations for the spatial derivatives in equation 26. If backward-in-space differences are used, equation 45 needs to be modified only slightly. For example, if $v_{x_{n-1/2,j}} > 0$ then equation 45a would become:

$$\hat{A} = TC\left[(A\theta)_{n-1/2,j}\left[\frac{D_{h_{xx\ n-1/2,j}}}{1/2(\Delta x_n+\Delta x_{n-1})} + v_{x_{n-1/2,j}}\right] + \hat{G} - \hat{H}\right]$$

and the term containing $v_{x_{n-1/2,j}}$ in equation 45e would be eliminated.

If fluid source/sink terms are present then equations 45e and 45f must be modified to account for them in the following manner:

$$\text{if } q^* > 0 \quad \text{then}$$

$$RHS = RHS + q^*c^* \tag{46}$$

$$\text{if } q^* < 0 \quad \text{then}$$

$$\hat{E} = \hat{E} - q^*. \tag{47}$$

Equation 44 must be solved for each node in the finite difference grid. Thus, we have reduced the problem to that of solving the matrix equation:

$$\bar{\bar{A}}\ \bar{c}^{i+1} = \overline{RHS} \tag{48}$$

where A = a pentadiagonal square coefficient matrix;
$\bar{c}^{i+1}$ = the vector of unknown concentrations at the i+1 time level; and
RHS = the vector defined by equation 45f.

As with the flow equation, VS2DT actually solves the residual form of equation 48 with an iterative matrix solver:

$$\bar{\bar{A}}\ \Delta\bar{c}^{i+1,k+1} = \overline{RHS}^{k} - \bar{\bar{A}}\bar{c}^{i+1,k}$$

where $\Delta\bar{c}^{i+1,k+1} = \bar{c}^{i+1,k+1} - \bar{c}^{i+1,k}$

k = iteration index; and
the terms at i+1 time level in equation 45f are assigned values from the $k^{\underline{th}}$ iteration.

Selection of fully implicit or time-centered differencing is a user option. The optimum method is problem dependent. Although the Crank-Nicholson method is more accurate, it can produce results which oscillate around the true solution. This oscillation is illustrated in the verification problems. Fully implicit time differencing eliminates the oscillations but can introduce numerical dispersion or the smearing of sharp fronts. Numerical dispersion can be controlled by limiting the size of each time step; however, small time steps can add great expense and computation time to each simulation.

Source/Sink Terms

Function subprograms (all named VTRET) have been written and tested for calculation of $\rho_b \frac{\partial \bar{c}}{\partial c}$ for adsorption and ion exchange. Six options are available to the user: Freundlich isotherm, Langmuir isotherm, monovalent-monovalent ion exchange, divalent-divalent ion exchange, monovalent-divalent ion exchange, and divalent-monovalent ion exchange.

As listed under Supplemental Information, the program is set up to use the Langmuir isotherm. The five other versions of VTRET are included as comment cards at the end of the program. To use any of the other options the required version of VTRET should be stripped of comment designation, compiled, and loaded with the compiled version of VS2DT that does not contain the Langmuir isotherm version of VTRET. Only one version of VTRET should be loaded with VS2DT at any one time. Variables required by the isotherm or ion-exchange option may vary with texture class (for example, if a simulation involves multiple soil types, then each soil type may have a different ion-exchange capacity).

Boundary and Initial Conditions

Specification of solute transport boundary conditions cannot be done independently of specification of flow boundaries. Two basic boundary conditions can be specified with regard to concentration: fixed-concentration node and a fixed-mass-flux node. In addition, for constant-head and constant-flux flow boundaries, the concentration of any flow entering the system must be specified. Table 1 lists the permissible combination of flow and transport boundary conditions. While some combinations that are not allowed may still be solved by the model, they are not permitted because no practical application for them exists.

Table 1.--*Summary of permissible combinations of boundary conditions*

[X, permitted; Y, mandatory; --, not allowed]

Flow boundary Conditions	Transport boundary conditions			
	Fixed Concentrations	Fixed mass flux	No specified boundary	Specified Concentration of inflow
Fixed head				
flow into domain	X	--	X	Y
flow out of domain	--	--	X	Y
Fixed flux				
into domain	X	--	X	Y
out of domain	--	--	X	Y
No specified boundary	X	X	X	--
Evaporation	--	--	X	--
Plant transpiration	--	--	X	--
Seepage face	--	--	X	--

For flow boundaries where flow is into the domain, there are two possible options for transport boundary conditions: 1) no specified boundary, for which the mass-flux rate into the domain is calculated as the influx rate times concentration of inflow (this is essentially treated as a fixed-mass-flux or Neumann boundary condition); and 2) fixed concentration or Dirichlet boundary condition, for which the mass flux rate into the domain is calculated as the sum of influx rate times concentrations of inflow plus the rate of dispersive flux from the boundary node. For flow boundaries where flow leaves the domain no transport boundary condition can be specified. Under this condition the rate of solute flux out of the domain is equal to the rate of water flux times the concentration at the exit node--diffusive flux out of the domain is not allowed. The evaporation boundary condition is treated differently from other boundaries where water leaves the domain; evaporating water is assumed to be solute free (no solute is allowed to leave the domain through evaporation). Therefore, solute may become concentrated in evaporation nodes as evaporation proceeds. The fixed-mass-flux boundary condition is used to represent a strictly diffusive flux and can be located only on nodes at which there is no inflow to or outflow from the domain.

Mass Balance

At the completion of every time step, the mass flux into and out of the system, as well as the change in mass stored in the system, is calculated. Printout of mass-balance results is an option in VS2DT. Fluxes into and out of the system are divided into dispersive/diffusive and advective fluxes. The former refers to fluxes dependent upon the concentration gradient between fixed concentration nodes and adjacent nodes. The latter represents changes in mass within the system due to mass entering or leaving the system with flowing water. When water flow is into the system, that water is assumed to have a concentration equal to that specified by the user. When water flows out of the system the concentration of that water is set equal to the concentration of the node from which the water is moving. The gain or loss of mass through source/sink terms also is determined.

The change in mass stored within the system over the last time step is calculated as:

$$\Delta SC^{i+1} = \sum_{n=1}^{NXR} \sum_{j=1}^{NLY} c_{n,j}^{i+1}\theta_{n,j}^{i+1}\left(1+\frac{Ss}{\phi}H_{n,j}^{i+1}\right) - c_{n,j}^{i}\theta_{n,j}^{i}\left(1+\frac{Ss}{\phi}H_{n,j}^{i}\right) V_{n,j} \quad (50)$$

where ΔSC^{i+1} = change in mass storage between time steps i and i+1, M;
NXR = number of columns in grid, dimensionless;
NLY = number of rows in grid, dimensionless;
Ss = specific storage, L^{-1};
ϕ = porosity, dimensionless.

The loss of mass due to decay and adsorption is calculated as:

$$\Delta SC_d^{i+1} = -\sum_{n=1}^{NXR} \sum_{j=1}^{NLY} V_{n,j} \lambda\Delta t\, c_{n,j}^{i+1}\left(\theta_{n,j}^{i+1}+\rho_b\frac{\partial\tilde{c}}{\partial c}\right) + \rho_b\frac{\partial\tilde{c}}{\partial c}\left(c_{n,j}^{i+1} - c_{n,j}^{i}\right) \quad (51)$$

where ΔSC_d^{i+1} = change in mass due to decay and adsorption between time steps i and i+1, M.

COMPUTER PROGRAM

This section contains information on program structure, data input, and considerations on spatial and temporal grid design for model application.

Program Structure

A listing of all new variables added to VS2DT for solute transport simulation is given in Table 2. An effort was made to keep the computer program in modular form so that it could be easily customized for particular applications. Three subroutines and one function routine were added to VS2D to allow simulation of solute transport, these are described below:

1)	VTVELO	Subroutine that calculates intercell velocities in the x and z directions.
2)	VTDCOEF	Subroutine that calculates the components of the dispersion coefficient tensor.
3)	VTSETUP	Subroutine that assembles the matrix equation and calls the matrix solving routine.
4)	VTRET	Function subroutine that calculates the adsorption term $\rho_b \frac{\partial \tilde{c}}{\partial c}$.

Six versions of routine VTRET are included in the program listing in Supplemental Information. These versions correspond to the Freundlich and Langmuir adsorption isotherms and monovalent-monovalent, divalent-divalent, monovalent-divalent, and divalent-monovalent ion exchange. When compiling the computer program, the user must select the appropriate version and be sure that the other versions are deleted or appear as comments. File definitions are similar to those described in Lappala and others (1987). However, when output is requested to Fortran file number 8, both pressure heads and concentrations are printed at the appropriate times. Similarly, concentrations are printed to Fortran file number 11 for selected observation points. The user also may now specify which mass-balance components are printed to file 9 (this option is described under Modifications to Computer Program VS2D in Supplemental Information).

Instructions for Data Input

Input-data formats are described in Table 3. The formats are very similar to the original VS2D input formats described by Lappala and others (1987). Several additional input variables are required for simulation of solute transport. If solute transport is not to be simulated then only two new variables need to be coded (ANG on line A-2, and TRANS on line A-6) in addition to those variables described in the original VS2D documentation. The variable RHOZ on line B-2 is no longer entered by the user. New users of VS2DT should obtain a copy of Lappala and others (1987) for additional information on input variables dealing with simulation of water flow.

Table 2.--*Definitions of new VS2DT program variables*

[NN, number of nodes]

Variable	Definition
DX1(NN)	XX Component of hydrodynamic dispersion tensor at left side of cell times $\Delta x/\Delta z$, L^2T^{-1}.
DX2(NN)	XZ Component of hydrodynamic dispersion tensor at left side of cell times $\Delta x/2\Delta z$, L^2T^{-1}.
DZ1(NN)	ZZ Component of hydrodynamic dispersion tensor at top of cell times $\Delta z/\Delta x$, L^2T^{-1}.
DZ2(NN)	ZX Component of hydrodynamic dispersion tensor at top of cell times $\Delta z/2\Delta x$, L^2T^{-1}.
VX(NN)	X Velocity at left side of cell, LT^{-1}.
VZ(NN)	Z Velocity at top of cell, LT^{-1}.
CC(NN)	Concentration, ML^{-3}.
COLD(NN)	Concentration at previous time step, ML^{-3}.
CS(NN)	Concentration of specified fluid sources, ML^{-3}.
QT(NN)	Fluid flux through constant head nodes, L^3T^{-1}.
NCTYP(NN)	Boundary condition or cell type indicator: 0 = internal node, 1 = specified concentration node, and 2 = specified solute flux node.
RET(NN)	Slope of adsorption isotherm times bulk density, dimensionless.
ANG	Angle at which grid is to be tilted, degrees.
TRANS	If = T, solute transport and flow are to be simulated; if = F, only flow is simulated.
TRANS1	If = T, matrix solver solves for head; if = F, matrix solver solves for concentration.
SSTATE	If = T, steady-state flow has been achieved.
CIS	If = T, centered-in-space differencing is used for transport equation; if = F, backward-in-space differencing is used.
CIT	If = T, centered-in-time differencing is used for transport equation; if = F, backward-in-time differencing is used.
EPS1	Convergence criteria for transport equation, ML^{-3}.
VPNT	If = T, velocities are written to file 6.
SORP	If = T, nonlinear sorption is to be simulated.

Table 3.--*Input data formats*

Card	Variable	Description
	[Line group A read by VSEXEC]	
A-1	TITL	80-character problem description (formatted read, 20A4).
A-2	TMAX	Maximum simulation time, T.
	STIM	Initial time (usually set to 0), T.
	ANG	Angle by which grid is to be tilted (Must be between -90 and +90 degrees, ANG = 0 for no tilting, see Supplemental Information for further discussion), degrees.
A-3	ZUNIT	Units used for length (A4).
	TUNIT	Units used for time (A4).
	CUNX	Units used for mass (A4).
Note: Line A-3 is read in 3A4 format, so the unit designations must occur in columns 1-4, 5-8, 9-12, respectively.		
A-4	NXR	Number of cells in horizontal or radial direction.
	NLY	Number of cells in vertical direction.
A-5	NRECH	Number of recharge periods.
	NUMT	Maximum number of time steps.
A-6	RAD	Logical variable = T if radial coordinates are used; otherwise = F.
	ITSTOP	Logical variable = T if simulation is to terminate after ITMAX iterations in one time step; otherwise = F.
	TRANS	Logical variable = T if solute transport is to be simulated.
Line A-6A is present only if TRANS = T.		
A-6A	CIS	Logical variable = T if centered-in-space differencing is to be used; = F if backward-in-space differencing is to be used for transport equation.
	CIT	Logical variable = T if centered-in-time differencing is to be used; = F if backward-in-time or fully implicit differencing is to be used.
	SORP	Logical variable = T if nonlinear sorption or ion exchange is to be simulated. Nonlinear sorption occurs when ion exchange, Langmuir isotherms, or Freundlich isotherms with n not equal to 1 are used.
A-7	F11P	Logical variable = T if head, moisture content, and saturation at selected observation points are to be written to file 11 at end of each time step; otherwise = F.

Table 3.--*Input data formats*--Continued

Card	Variable	Description
A-7--Continued	F7P	Logical variable = T if head changes for each iteration in every time step are to be written in file 7; otherwise = F.
	F8P	Logical variable = T if output of pressure heads (and concentrations if TRANS = T) to file 8 is desired at selected observation times; otherwise = F.
	F9P	Logical variable = T if one-line mass balance summary for each time step to be written to file 9; otherwise = F.
	F6P	Logical variable = T if mass balance is to be written to file 6 for each time step; = F if mass balance is to be written to file 6 only at observation times and ends of recharge periods.
A-8	THPT	Logical variable = T if volumetric moisture contents are to be written to file 6; otherwise = F.
	SPNT	Logical variable = T if saturations are to be written to file 6; otherwise = F.
	PPNT	Logical variable = T if pressure heads are to be written to file 6; otherwise = F.
	HPNT	Logical variable = T if total heads are to be written to file 6; otherwise = F.
	VPNT	Logical variable = T if velocities are to be written to file 6; requires TRANS = T.
A-9	IFAC	= 0 if grid spacing in horizontal (or radial) direction is to be read in for each column and multiplied by FACX. = 1 if all horizontal grid spacing is to be constant and equal to FACX. = 2 if horizontal grid spacing is variable, with spacing for the first two columns equal to FACX and the spacing for each subsequent column equal to XMULT times the spacing of the previous column, until the spacing equals XMAX, whereupon spacing becomes constant at XMAX.

Table 3.--*Input data formats*--Continued

Card	Variable	Description
A-9--Continued	FACX	Constant grid spacing in horizontal (or radial) direction (if IFAC=1); constant multiplier for all spacing (if IFAC=0); or initial spacing (if IFAC=2), L.
Line set A-10 is present if IFAC = 0 or 2.		
If IFAC = 0,		
A-10	DXR	Grid spacing in horizontal or radial direction. Number of entries must equal NXR, L.
If IFAC = 2,		
A-10	XMULT	Multiplier by which the width of each node is increased from that of the previous node.
	XMAX	Maximum allowed horizontal or radial spacing, L.
A-11	JFAC	= 0 if grid spacing in vertical direction is to be read in for each row and multiplied by FACZ. = 1 if all vertical grid spacing is to be constant and equal to FACZ. = 2 if vertical grid spacing is variable, with spacing for the first two rows equal to FACZ and the spacing for each subsequent row equal to ZMULT times the spacing at the previous row, until spacing equals ZMAX, whereupon spacing becomes constant at ZMAX.
	FACZ	Constant grid spacing in vertical direction (if JFAC=1); constant multiplier for all spacing (if JFAC =0); or initial vertical spacing (if JFAC=2), L.
Line set A-12 is present only if JFAC = 0 or 2.		
If JFAC = 0,		
A-12	DELZ	Grid spacing in vertical direction; number of entries must equal NLY, L.
If JFAC = 2,		
A-12	ZMULT	Multiplier by which each node is increased from that of previous node.
	ZMAX	Maximum allowed vertical spacing, L.
Line sets A-13 to A-14 are present only if F8P = T,		
A-13	NPLT	Number of time steps to write heads and concentrations to file 8 and heads, concentrations, saturations, and/or moisture contents to file 6.

Table 3.--*Input data formats*--Continued

Card	Variable	Description
A-14	PLTIM	Elapsed times at which pressure heads and concentrations are to be written to file 8, and heads, concentrations, saturations, and/or moisture contents to file 6, T.
Line sets A-15 to A-16 are present only if F11P = T,		
A-15	NOBS	Number of observation points for which heads, concentrations, moisture contents, and saturations are to be written to file 11.
A-16	J,N	Row and column of observation points. A double entry is required for each observation point, resulting in 2xNOBS values.
Lines A-17 and A-18 are present only if F9P = T.		
A-17	NMB9	Total number of mass balance components to be written to File 9.
A-18	MB9	The index number of each mass balance component to be written to file 9. (See table 7 in Supplemental Information for index key)
	[Line group B read by subroutine VSREAD]	
B-1	EPS	Closure criteria for iterative solution of flow equation, units used for head, L.
	HMAX	Relaxation parameter for iterative solution. See discussion in Lappala and others (1987) for more detail. Value is generally in the range of 0.4 to 1.2.
	WUS	Weighting option for intercell relative hydraulic conductivity: WUS = 1 for full upstream weighting. WUS = 0.5 for arithmetic mean. WUS = 0.0 for geometric mean.
	EPS1	Closure criteria for iterative solution of transport equation, units used for concentration, ML^{-3}. Present only if TRANS = T.
B-3	MINIT	Minimum number of iterations per time step.
	ITMAX	Maximum number of iterations per time step. Must be less than 200.
B-4	PHRD	Logical variable = T if initial conditions are read in as pressure heads; = F if initial conditions are read in as moisture contents.

Table 3.--*Input data formats*--Continued

Card	Variable	Description
B-5	NTEX	Number of textural classes or lithologies having different values of hydraulic conductivity, specific storage, and/or constants in the functional relations among pressure head, relative conductivity, and moisture content.
	NPROP	Number of flow properties to be read in for each textural class. When using Brooks and Corey or van Genuchten functions, set NPROP = 6, and when using Haverkamp functions, set NPROP = 8. When using tabulated data, set NPROP = 6 plus number of data points in table. [For example, if the number of pressure heads in the table is equal to N1, then set NPROP =3*(N1+1)+3]
	NPROP1	Number of transport properties to be read in for each textural class. For no adsorption set NPROP1 = 6. For a Langmuir or Freundlich isotherm set NPROP1 = 7. For ion exchange set NPROP1 = 8. Present only if TRANS = T.
Line sets B-6, B-7, and B-7A must be repeated NTEX times		
B-6	ITEX	Index to textural class.
B-7	ANIZ(ITEX)	Ratio of hydraulic conductivity in the z-coordinate direction to that in the x-coordinate direction for textural class ITEX.
	HK(ITEX,1)	Saturated hydraulic conductivity (K) in the x-coordinate direction for class ITEX, LT^{-1}.
	HK(ITEX,2)	Specific storage (S_s) for class ITEX, L^{-1}.
	HK(ITEX,3)	Porosity for class ITEX.

Definitions for the remaining sequential values on this line are dependent upon which functional relation is selected to represent the nonlinear coefficients. Four different functional relations are allowed: (1) Brooks and Corey, (2) van Genuchten, (3) Haverkamp, and (4) tabular data. The choice of which of these to use is made when the computer program is compiled, by including only the function subroutine which pertains to the desired relation (see discussion in Lappala and others (1987) for more detail).

Table 3.--*Input data formats*--Continued

Card	Variable	Description
B-7--Continued		
In the following descriptions, definitions for the different functional relations are indexed by the above numbers. For tabular data, all pressure heads are input first (in decreasing order from the largest to the smallest), all relative hydraulic conductivities are then input in the same order, followed by all moisture contents.		
	HK(ITEX,4)	(1) h_b, L. (must be less than 0.0).
		(2) α', L. (must be less than 0.0).
		(3) A', L. (must be less than 0.0).
		(4) Largest pressure head in table.
	HK(ITEX,5)	(1) Residual moisture content (θ_r).
		(2) Residual moisture content (θ_r).
		(3) Residual moisture content (θ_r).
		(4) Second largest pressure head in table.
	HK(ITEX,6)	(1) λ, pore-size distribution index.
		(2) β'.
		(3) B'.
		(4) Third largest pressure head in table.
	HK(ITEX,7)	(1) Not used.
		(2) Not used.
		(3) α, L. (must be less than 0.0).
		(4) Fourth largest pressure head in table.
	HK(ITEX,8)	(1) Not used.
		(2) Not used.
		(3) β.
		(4) Fifth largest pressure head in table.
For functional relations (1), (2), and (3) no further values are required on this line for this textural class. For tabular data (4), data input continues as follows:		
	HK(ITEX,9)	Next largest pressure head in table.
	K(ITEX,N1+3)	Minimum pressure head in table. (Here N1 = Number of pressure heads in table; NPROP = 3*(N1+1)+3).
	HK(ITEX,N1+4)	Always input a value of 99.
	HK(ITEX,N1+5)	Relative hydraulic conductivity corresponding to first pressure head.
	HK(ITEX,N1+6)	Relative hydraulic conductivity corresponding to second pressure head.
	.	
	.	
	.	
	HK(ITEX,2*N1+4)	Relative hydraulic conductivity corresponding to smallest pressure head.
	HK(ITEX,2*N1+5)	Always input a value of 99.
	HK(ITEX,2*N1+6)	Moisture content corresponding to first pressure head.

Table 3.--*Input data formats*--Continued

Card	Variable	Description
B-7--Continued		
HK(ITEX,2*N1+7)		Moisture content corresponding to second pressure head.
.		
.		
.		
HK(ITEX,3*N1+5)		Moisture content corresponding to smallest pressure head.
HK(ITEX,3*N1+6)		Always input a value of 99.
Regardless of which functional relation is selected there must be NPROP+1 values on line B-7.		
Line B-7A is present only if TRANS = T.		
B-7A	HT(ITEX,1)	α_L, L.
	HT(ITEX,2)	α_T, L.
	HT(ITEX,3)	D_m, L^2T^{-1}.
	HT(ITEX,4)	λ, decay constant, T^{-1}.
	HT(ITEX,5)	ρ_b (can be set to 0 for no adsorption or ion exchange), ML^{-3}.
	HT(ITEX,6)	= 0 for no adsorption or ion exchange, = K_d for linear adsorption isotherm, = K_1 for Langmuir isotherm, = K_f for Freundlich isotherm, = K_m for ion exchange.
	HT(ITEX,7)	= Q for Langmuir isotherm, = n for Freundlich isotherm (Note: n is a real, rather than an integer, variable), = $\hat{Q}$ for ion exchange, not used when adsorption is not simulated.
	HT(JTEX,8)	= C_0 for ion exchange, only used for ion exchanged.
B-8	IROW	If IROW = 0, textural classes are read for each row. This option is preferable if many rows differ from the others. IF IROW = 1, textural classes are read in by blocks of rows, each block consisting of all the rows in sequence consisting of uniform properties or uniform properties separated by a vertical interface.
Line set B-9 is present only if IROW = 0.		
B-9	JTEX	Indices (ITEX) for textural class for each node, read in row by row. There must be NLY*NXR entries.

Table 3.--*Input data formats*--Continued

Card	Variable	Description
Line set B-10 is present only if IROW = 1.		
As many groups of B-10 variables as are needed to completely cover the grid are required. The final group of variables for this set must have IR = NXR and JBT = NLY.		
B-10	IL	Left hand column for which texture class applies. Must equal 1 or [IR(from previous card)+1].
	IR	Right hand column for which texture class applies. Final IR for sequence of rows must equal NXR.
	JBT	Bottom row of all rows for which the column designations apply. JBT must not be increased from its initial or previous value until IR = NXR.
	JRD	Texture class within block.
Note: As an example, for a column of uniform material; IL = 1, IR = NXR, JBT = NLY, and JRD = texture class designation for the column material. One line will represent the set for this example.		
B-11	IREAD	If IREAD = 0, all initial conditions in terms of pressure head or moisture content as determined by the value of PHRD are set equal to FACTOR. If IREAD = 1, all initial conditions are read from file IU in user-designated format and multiplied by FACTOR. If IREAD = 2 initial conditions are defined in terms of pressure head, and an equilibrium profile is specified above a free-water surface at a depth of DWTX until a pressure head of HMIN is reached. All pressure heads above this are set to HMIN.
	FACTOR	Multiplier or constant value, depending on value of IREAD, for initial conditions, L.
Line B-12 is present only if IREAD = 2,		
B-12	DWTX	Depth to free-water surface above which an equilibrium profile is computed, L.
	HMIN	Minimum pressure head to limit height of equilibrium profile; must be less than zero, L.

Table 3.--*Input data formats*--Continued

Card	Variable	Description
Line B-13 is read only if IREAD = 1,		
B-13	IU	Unit number from which initial head values are to be read.
	IFMT	Format to be used in reading initial head values from unit IU. Must be enclosed in quotation marks, for example '(10X,E10.3)'.
B-14	BCIT	Logical variable = T if evaporation is to be simulated at any time during the simulation; otherwise = F.
	ETSIM	Logical variable = T if evapotranspiration (plant-root extraction) is to be simulated at any time during the simulation; otherwise = F.
Line B-15 is present only if BCIT = T or ETSIM = T.		
B-15	NPV	Number of ET periods to be simulated. NPV values for each variable required for the evaporation and/or evapotranspiration options must be entered on the following lines. If ET variables are to be held constant throughout the simulation code, NPV = 1.
	ETCYC	Length of each ET period, T.

Note: For example, if a yearly cycle of ET is desired and monthly values of PEV, PET, and the other required ET variables are available, then code NPV = 12 and ETCYC = 30 days. Then, 12 values must be entered for PEV, SRES, HA, PET, RTDPTH, RTBOT, RTTOP, and HROOT. Actual values, used in the program, for each variable are determined by linear interpolation based on time.

Line B-16 to B-18 are present only if BCIT = T.		
B-16	PEVAL	Potential evaporation rate (PEV) at beginning of each ET period. Number of entries must equal NPV, LT^{-1}.

To conform with the sign convention used in most existing equations for potential evaporation, all entries must be greater than or equal to 0. The program multiplies all nonzero entries by -1 so that the evaporative flux is treated as a sink rather than a source.

Table 3.--*Input data formats*--Continued

Card	Variable	Description
B-17	RDC(1,J)	Surface resistance to evaporation (SRES) at beginning of ET period, L^{-1}. For a uniform soil, SRES is equal to the reciprocal of the distance from the top active node to land surface, or 2./DELZ(2). If a surface crust is present, SRES may be decreased to account for the added resistance to water movement through the crust. Number of entries must equal NPV.
B-18	RDC(2,J)	Pressure potential of the atmosphere (HA) at beginning of ET period; may be estimated using equation 6 of Lappala and others (1987), L. Number of entries must equal NPV.
Lines B-19 to B-23 are present only if ETSIM = T.		
B-19	PTVAL	Potential evapotranspiration rate (PET) at beginning of each ET period, LT^{-1}. Number of entries must equal NPV. As with PEV, all values must be greater than or equal to 0.
B-20	RDC(3,J)	Rooting depth at beginning of each ET period, L. Number of entries must equal NPV.
B-21	RDC(4,J)	Root activity at base of root zone at beginning of each ET period, L^{-2}. Number of entries must equal NPV.
B-22	RDC(5,J)	Root activity at top of root zone at beginning of each ET period, L^{-2}. Number of entries must equal NPV.
Note: Values for root activity generally are determined empirically, but typically range from 0 to 3.0 cm/cm3. As programmed, root activity varies linearly from land surface to the base of the root zone, and its distribution with depth at any time is represented by a trapezoid. In general, root activities will be greater at land surface than at the base of the root zone.		
B-23	RDC(6,J)	Pressure head in roots (HROOT) at beginning of each ET period, L. Number of entries must equal NPV.
Lines B-24 and B-25 are present only if TRANS = T.		
B-24	IREAD	If IREAD = 0, all initial concentrations are set equal to FACTOR. If IREAD = 1, all initial concentrations are read from file IU in user designated format and multiplied by FACTOR.

Table 3.--*Input data formats*--Continued

Card	Variable	Description
B-24--Continued	FACTOR	Multiplier or constant value, depending on value of IREAD, for initial concentrations.
Line B-25 is present only if IREAD = 1.		
B-25	IU	Unit number from which initial concentrations are to be read.
	IFMT	Format to be used in reading initial head values from unit IU. Must be enclosed in quotation marks, for example '(10X, E10.3)'.
[Line group C read by subroutine VSTMER, NRECH sets of C lines are required]		
C-1	TPER	Length of this recharge period, T.
	DELT	Length of initial time step for this period, T.
C-2	TMLT	Multiplier for time step length.
	DLTMX	Maximum allowed length of time step, T.
	DLTMIN	Minimum allowed length of time step, T.
	TRED	Factor by which time-step length is reduced if convergence is not obtained in ITMAX iterations. Values usually should be in the range 0.1 to 0.5. If no reduction of time-step length is desired, input a value of 0.0.
C-3	DSMAX	Maximum allowed change in head per time step for this period, L.
	STERR	Steady-state head criterion; when the maximum change in head between successive time steps is less than STERR, the program assumes that steady state has been reached for this period and advances to next recharge period, L.
C-4	POND	Maximum allowed height of ponded water for constant flux nodes. See Lappala ans others (1987) for detailed discussion of POND, L.
C-5	PRNT	Logical variable = T if heads, concentration, moisture contents, and/or saturations are to be printed to file 6 after each time step; = F if they are to be written to file 6 only at observation times and ends of recharge periods.
C-6	BCIT	Logical variable = T if evaporation is to be simulated for this recharge period; otherwise = F.

Table 3.--*Input data formats*--Continued

Card	Variable	Description
C-6--Continued	ETSIM	Logical variable = T if evapotranspiration (plant-root extraction) is to be simulated for this recharge period; otherwise = F.
	SEEP	Logical variable = T if seepage faces are to be simulated for this recharge period; otherwise = F
C-7 to C-9 cards are present only if SEEP = T,		
C-7	NFCS	Number of possible seepage faces. Must be less than or equal to 4.
Line sets C-8 and C-9 must be reported NFCS times		
C-8	JJ	Number of nodes on the possible seepage face.
	JLAST	Number of the node which initially represents the highest node of the seep; value can range from 0 (bottom of the face) up to JJ (top of the face).
C-9	J,N	Row and column of each cell on possible seepage face, in order from the lowest to the highest elevation; JJ pairs of values are required.
C-10	IBC	Code for reading in boundary conditions by individual node (IBC=0) or by row or column (IBC=1). Only one code may be used for each recharge period, and all boundary conditions for period must be input in the sequence for that code.
Line set C-11 is read only if IBC = 0. One line should be present for each node for which new boundary conditions are specified.		
C-11	JJ	Row number of node.
	NN	Column number of node.
	NTX	Node type identifier for boundary conditions. = 0 for no specified boundary (needed for resetting some nodes after intial recharge period); = 1 for specified pressure head; = 2 for specified flux per unit horizontal surface area in units of LT-1; = 3 for possible seepage face; = 4 for specified total head; = 5 for evaporation; = 6 for specified volumetric flow in units of L^3T^{-1}.

Table 3.--*Input data formats*--Continued

Card	Variable	Description
C-11--Continued	PFDUM	Specified head for NTX = 1 or 4 or specified flux for NTX = 2 or 6. If codes 0, 3, or 5 are specified, the line should contain a dummy value for PFDUM or should be terminated after NTX by a blank and a slash.
	NTC	Node type identifier for transport boundary conditions = 0 for no specified boundary; = 1 for specified concentration, ML^{-3}; = 2 for specified mass flux, MT^{-1}. Present only if TRANS = T.
	CF	Specified concentration for NTC = 1 or NTX = 1,2,4, or 6; or specified flux for NTC = 2. Present only if TRANS = T.
C-12 is present only if IBC = 1. One card should be present for each row or column for which new boundary conditions are specified,		
C-12	JJT	Top node of row or column of nodes sharing same boundary condition.
	JJB	Bottom node of row or column of nodes having same boundary condition. Will equal JJT if a boundary row is being read.
	NNL	Left column in row or column of nodes having same boundary condition.
	NNR	Right column of row or column of nodes having same boundary condition. Will equal NNL if a boundary column is being read in.
	NTX	Same as line C-11.
	PFDUM	Same as line C-11.
	NTC	Same as line C-11.
	CF	Same as line C-11.
C-13	Designated end of recharge period. Must be included after line C-12 data for each recharge period. Two C-13 lines must be included after final recharge period. Line must always be entered as 999999 /.	

Considerations in Discretization

Users need to be aware that selection of spatial grid increments and time step sizes can have a large effect upon calculated results for the advection-dispersion equation. Those readers familiar with the flow portion of VS2D are well aware that fine spatial and temporal discretizations are required to accurately solve variably saturated flow problems involving

sharp wetting fronts (such as infiltration to dry soil). For such problems the discretizations are probably adequate for solute-transport simulation. However for other problems, solute transport simulations may require finer discretizations than that required for flow simulations in order to obtain accurate results.

Two common problems are encountered in approximating the advection-dispersion equation by the finite-difference method: numerical dispersion and numerical oscillation. Numerical dispersion arises from the use of backward differencing and is illustrated by the smearing of sharp concentration fronts. Backward-in-space differencing is first-order accurate in terms of Δx, while backward-in-time differencing is first-order accurate in terms of Δt. Kipp (1987) makes the following recommendations to insure that numerical dispersion remains small relative to actual physical dispersion:

$$\frac{\Delta x}{2} << \alpha_L \tag{52}$$

and

$$\frac{|v|\Delta t}{2} << \alpha_L \ . \tag{53}$$

Numerical oscillations arise from the use of central differences. It is illustrated by overshoot and undershoot in the vicinity of sharp concentration fronts. Centered-in-space differencing is second order accurate in Δx and hence introduces no numerical dispersion. Numerical oscillations may occur unless:

$$\frac{|v|\Delta y}{|D_{h_{zz}}|} + \frac{|v|\Delta x}{|D_{h_{xx}}|} \leqq 2 \tag{54}$$

This can be a very restrictive requirement. In practice a little more leeway is allowed especially for problems that do not involve sharp concentration fronts. Centered-in-time differencing is second order accurate in Δt. It can also cause oscillations, but criteria for determining a maximum Δt to ensure no oscillations are not as developed as for spatial discretization. In general, the differences between centered and backward time differencing are not as great as the differences encountered in spatial differencing.

Regardless of the discretization methods or refinements that are used, it is strongly recommended that the effects of grid size and time-step size be evaluated for any application of this computer program. This can be done with a simple sensitivity test by refining both the space and time grid. The results obtained with the original and refined grids should be compared and a decision made as to the significance of the differences.

MODEL VERIFICATION AND EXAMPLE PROBLEMS

The transport option of VS2DT was verified on five test problems. Three of the problems have analytical solutions. The other two problems are compared with results of other numerical models. No verification problems

involve ion exchange. However the ion-exchange options were all tested with the example problems presented by Grove and Stollenwerk (1984). Results obtained with VS2DT were virtually identical to those of Grove and Stollenwerk (1984).

Verification Problem 1

The first test problem involves fluid injection from a well in a fully saturated confined aquifer. Axial symmetry is assumed and radial coordinates are used in the simulation. The solute concentration within the aquifer is initially 0, while the concentration of the injected water is 1.0. This problem has been simulated previously with the finite-element program SUTRA by Voss (1984). Analytical solutions have been developed by Tang and Babu (1979) and Hsieh (1986). Hoopes and Harleman (1967) and Gelhar and Collins (1971) developed approximate analytical solutions. The analytical solution of Hsieh (1986) has the following form:

$$c(r^*,t^*) = C_0 \left(1+\int_0^\infty F(\nu)d\nu\right) \tag{55}$$

$$F(\nu) = \frac{2\exp[-\nu^2 t^* + (r^* - r_w^*)/2][Ai(y)Bi(y_w) - Ai(y_w)Bi(y)]}{\pi\nu([Ai(y_w)]^2 + [Bi(y_w)]^2)} \tag{56}$$

where $r^* = r/\alpha_L$;

r = radial distance from injection well ;

t^* = dimensionless time ;
$= Qt/(2\pi\theta_s b\alpha_L^2)$;

Q = injection rate ;
$= 225\ m^3/h$;

θ_s = moisture content at saturation ;
$= 0.20$;

b = thickness of aquifer ;
$= 10$ m ;

C_0 = concentration of injected water ;

$r_w^* = r_w/\alpha_L$;

r_w = radius of injection well ;
$= 0.05$ m ;

Ai = Airy function;

Bi = Airy function ;

$y = \frac{1-4r^*\nu}{4\nu^{4/3}}$; and

$y_w = \frac{1-4r_w^*\nu}{4\nu^{4/3}}$.

The spatial grid consisted of 3 rows and 188 columns. Spacing in the vertical direction was 10 m. Spacing in the radial direction increased from 0.05 m at the injection well by a factor of 1.2 until a maximum size of 5 m was reached. The total length of the grid in the radial direction was 847 m. Initial total head was 10.0 m everywhere in the aquifer. The following constants were used:

$K = .36$ m/h;
$\alpha_L = 10.$ m;
$\alpha_T = 0.$ m.

A pumping period of 2,000 h was simulated. The length of the initial time step was 1×10^{-7} h. The time-step size was increased for each subsequent time step by a factor of 1.5 until the maximum allowed time-step size of 2.0 h was reached. A total of 1,043 time steps were used. Flow boundaries consisted of a constant flux of +225 m^3/h at the injection well and a fixed head of 10.0 m at the radial boundary. Centered-in-time and centered-in-space differencing were selected.

Results of VS2DT and the analytical solution are shown in figure 5 for four times. The match between results is very good at all times.

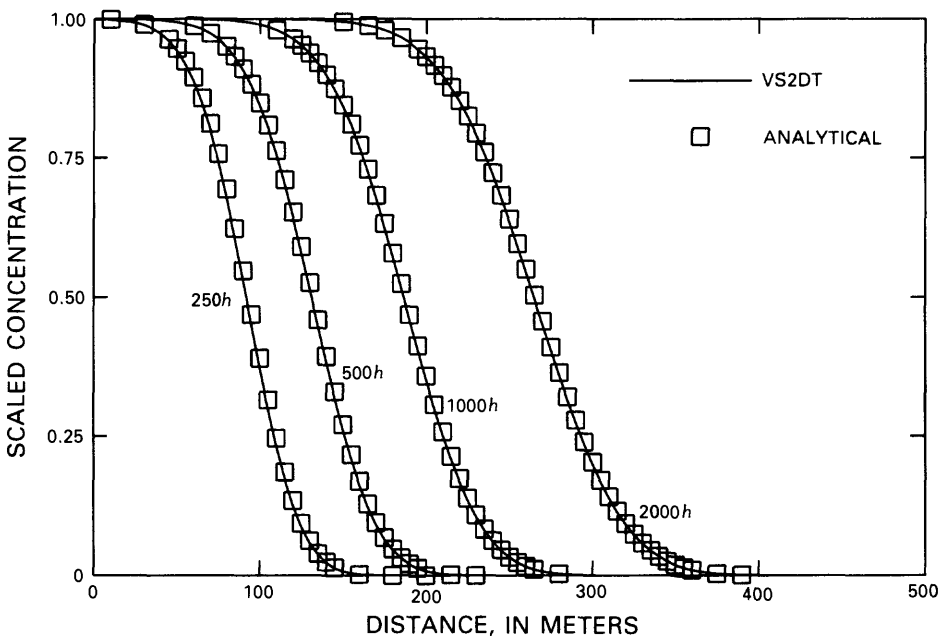


Figure 5.--Graph showing results of first verification problem: Analytical solution of Hsieh (1986) and numerical solution of VS2DT.

Verification Problem 2

In the second test problem, solute transport through a saturated one-dimensional column was simulated for a period of 7,200 s. Initial solute concentration was 0 at all points in the column. A steady-flow field was obtained in the column so that the interstitial velocity was 2.7778×10 m/s.

At time equal 0 the boundary at the top of the vertical column was set to a fixed concentration of 1.0. Ogata and Banks (1961) present an analytical solution to this problem. Kipp (1987) used the program HST to simulate the same problem.

The column was 160 m in length and was represented by 43 nodes. Spacing was set at 0.1 m at the top of the column and allowed to increase by a factor of 1.2 for each subsequent node. The maximum allowed node spacing was 8.0 m. The initial time step length was 1×10^{-7} s. This was increased by a factor of 1.5 for each subsequent step. The maximum allowed time step size was 200 s. A total of 86 time steps was used in the simulation. The following constants were used:

$K = 9.8\times10^{-4}$ m/s;
$\theta_s = 0.50$;
$\alpha_L = 10$ m; and
$D_m = 1\times10^{-10}$ m^2/s.

Results are shown in figure 6 at 7,200 s. A good match was obtained between the VS2DT results and the analytical solution.

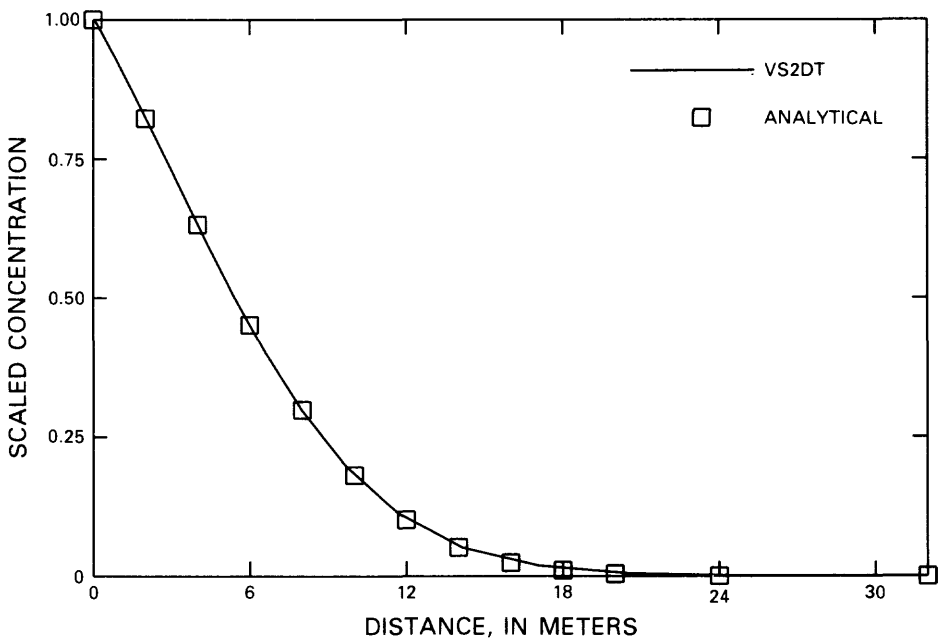


Figure 6.--Graph showing analytical and numerical results of second verification problem at 7,200 seconds.

Verification Problem 3

The third test involves infiltration of water containing a solute into a one-dimensional unsaturated solute-free column of soil. After 2.8 h the infiltrating water is solute free. The problem is based on a field experiment described by Warrick, Biggar, and Nielsen (1971). The problem has been

simulated numerically by van Genuchten (1982), Voss (1984), and Kwicklis (1987). The spatial grid consisted of a column containing 50 rows, each 2.5 cm in height. A period of 9 h was simulated. A constant time step size of .048 h was used. The hydraulic properties of the soil were represented by the following equations:

$$\theta(h) = \begin{cases} 0.6829 - 0.09524 \ln(|h|), & h \leq -29.484 \\ 0.4531 - 0.02732 \ln(|h|), & -29.484 < h \leq -14.495 \end{cases} \quad (57)$$

$$K_r(h) = \begin{cases} 5.1164 \times 10^4 \, |h|^{-3.4095}, & h \leq -29.484 \\ 13.672 \, |h|^{-.97814}, & -29.484 < h \leq -14.495 \end{cases} \quad (58)$$

where h = pressure head, in centimeters;
K = 37.8 cm/day;
α_L = 1.026 cm; and
D_m = 0.6 cm^2/day.

Initial concentrations were 0 everywhere in the column. Initial moisture contents and boundary conditions are:

$$\theta = \begin{cases} .15 + z/1{,}200, & 0 < z \leq 60 \text{ cm} \\ .20, & 60 \text{ cm} < z \end{cases};$$

$$h = -14.495 \qquad z = 0, \qquad t > 0; \text{ and}$$

$$c = \begin{cases} 209 & z = 0, \quad t < .11667 \text{ days} \\ 0 & z = 0, \quad t \geq .11667 \text{ days}. \end{cases}$$

Figures 7 and 8 show VS2DT results using centered-in-time and centered-in-space differencing (CTCS) for 2 and 9 h along with the results of van Genuchten (1982), who assumed that the correct solution was obtained by using a very fine grid. The results are in good agreement at all times, but the simulated concentration peak at 9.0 h lags the true solution slightly. Small oscillations in concentrations at the tail of the plume can be seen at 9 h because of the use of centered-in-space differencing. It is interesting to note that the wetting front is propagated more quickly through the column than is the solute front.

To illustrate the effects of using the various differencing options, the problem was rerun using the following differencing schemes:

1) backward-in-time, centered-in-space (BTCS)
2) backward-in-time, backward-in-space (BTBS)

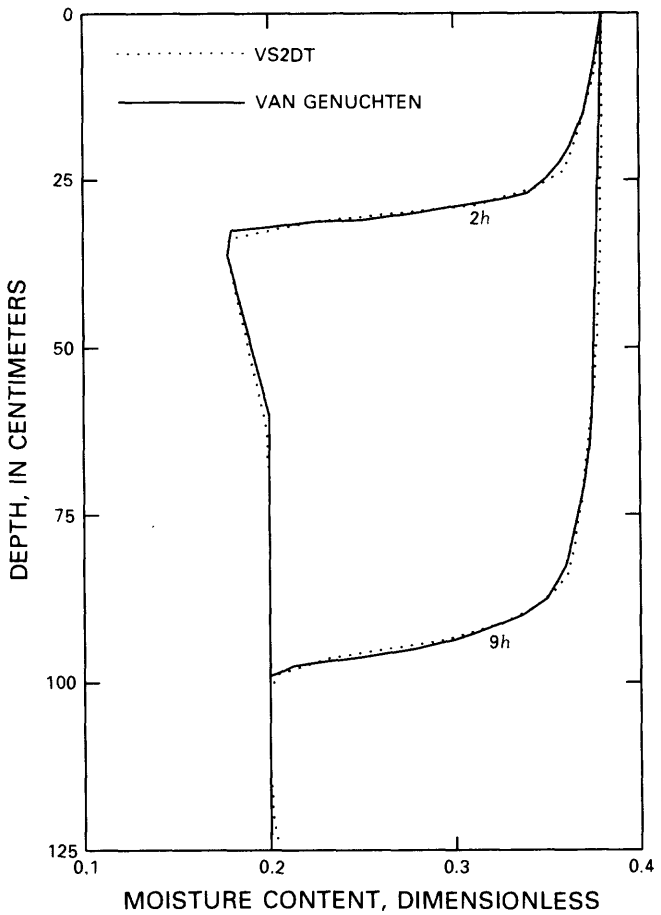


Figure 7.--Graph showing results of third verification problem, moisture content versus depth for VS2DT and van Genuchten (1982).

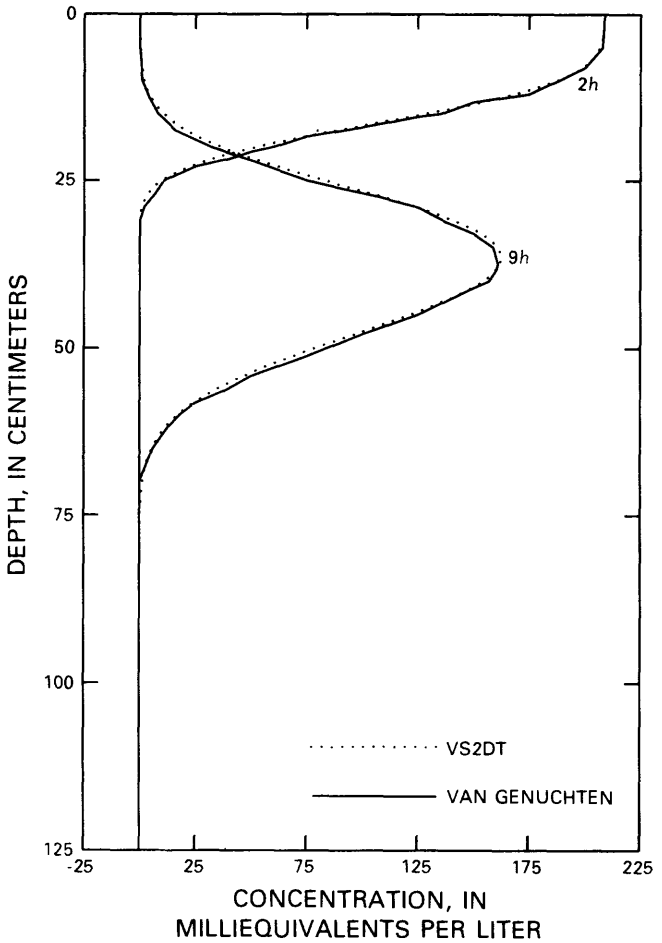


Figure 8.--Graph showing results of third verification problem, concentration versus depth for VS2DT (centered-in-time and centered-in-space differencing) and van Genuchten (1982).

Results for the concentration field are shown in figures 9 and 10. The BTCS simulation (fig. 9) produced concentration profiles similar to those in figure 8, but with slightly more smearing at the leading and trailing edges of the profile. The BTBS simulation (fig. 10) produced quite a different profile, with decreased peak concentrations and drastic smearing on both sides of the peak.

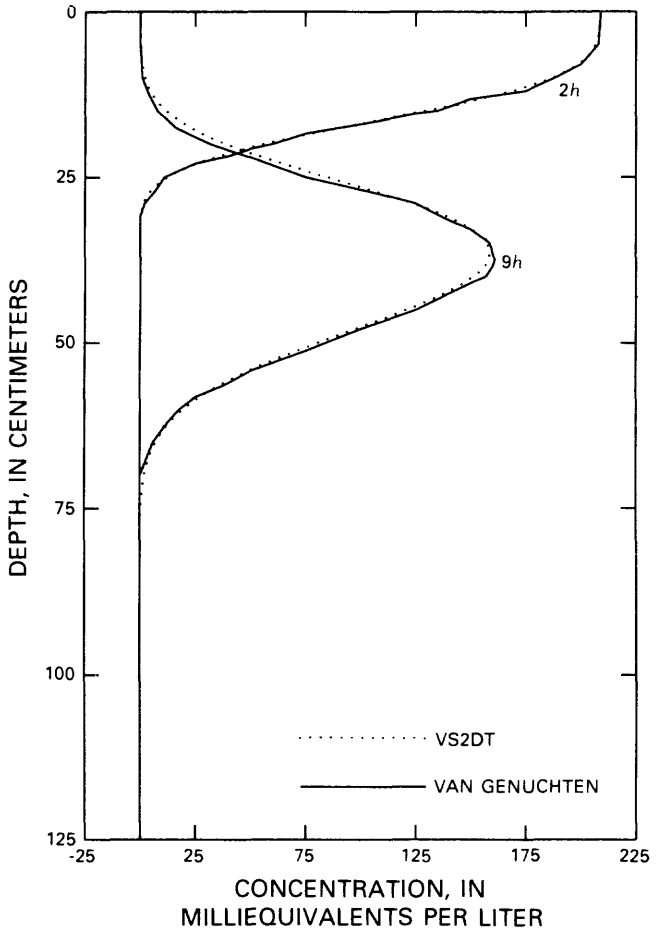


Figure 9.--Graph showing results of third verification problem, concentration versus depth for VS2DT (backward-in-time and centered-in-space differencing) and van Genuchten (1982).

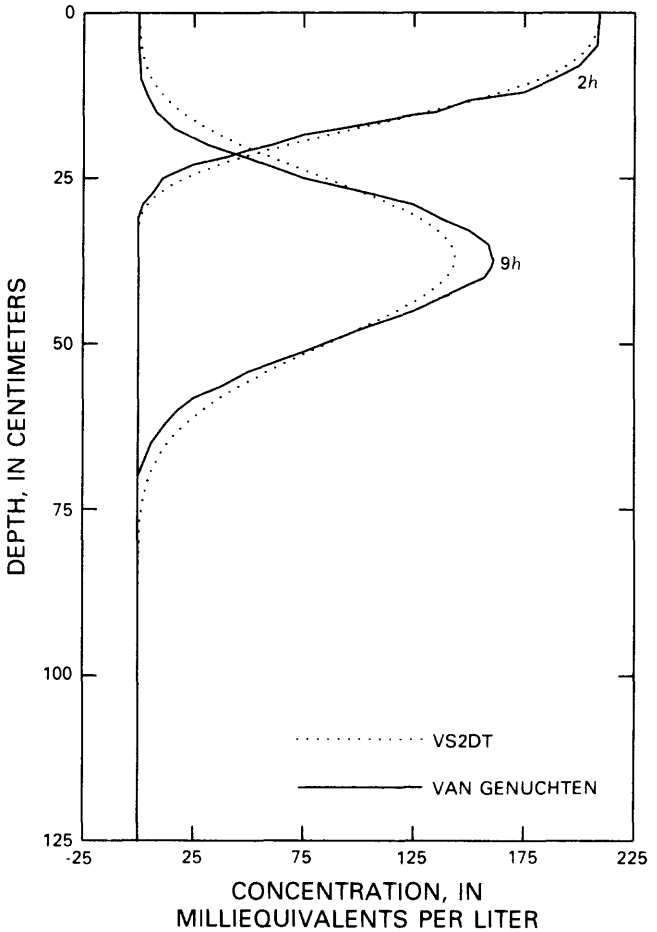


Figure 10.--Graph showing results of third verification problem, concentration versus depth for VS2DT (backward-in-time and backward-in-space differencing) and van Genuchten (1982).

Verification Problem 4

The fourth verification problem involves two-dimensional transport of a nonconservative tracer in a vertical plane. The problem is taken from Huyakorn and others (1985). The vertical section is 10 cm in height and 15 cm in width (fig. 11). Initial pressure head is everywhere -90 cm. The right-hand boundary is maintained at that pressure head. The top and bottom of the section are no-flow boundaries, as is the bottom 6 cm of the left-hand boundary. Water flows into the domain along the top 4 cm of the left-hand side. The pressure heads there are fixed at h = z-4, where z is measured

positive downward and both h and z are in cm. Initial solute concentration is 0 in the plane. The inflowing water is given a concentration of 1 mg/L. Grid spacing was uniform with $\Delta x = \Delta z = 1$ cm. The time-step size was initially 0.01 days and was increased by a factor of 1.2 for each subsequent time step, with the maximum allowed time-step size of 0.05 d. The following hydraulic functions and physical properties were used:

$$\theta(h) = .45 + .003h \tag{59}$$

$$K_r(\theta) = 3.33\ \theta - .5 \tag{60}$$

K = 1 cm/d;
α_L = 1 cm;
α_T = 0 cm;
D_m = 0.01 cm^2/d;
λ = 0.001 d^{-1}; and
$\rho_b K_d$ = θ.

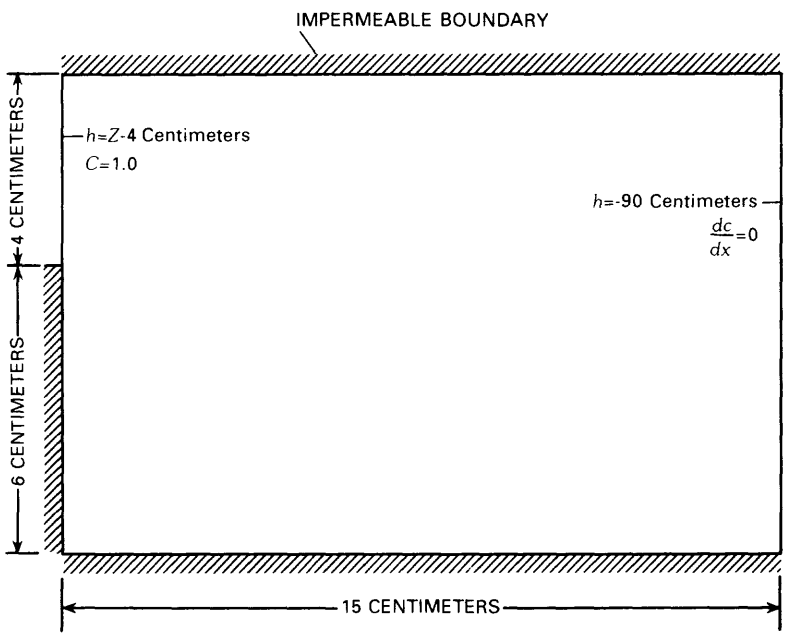


Figure 11.--Sketch showing boundary and initial conditions for verification problem 4.

Results for VS2DT, using backward-in-time and centered-in-space differencing, and the finite-element model of Huyakorn and others (1985) are shown in terms of a horizontal profile (fig. 12) and a vertical profile (fig. 13). In general, the results of the two models are very similar. Because the manner in which nodes are treated is different for finite-element

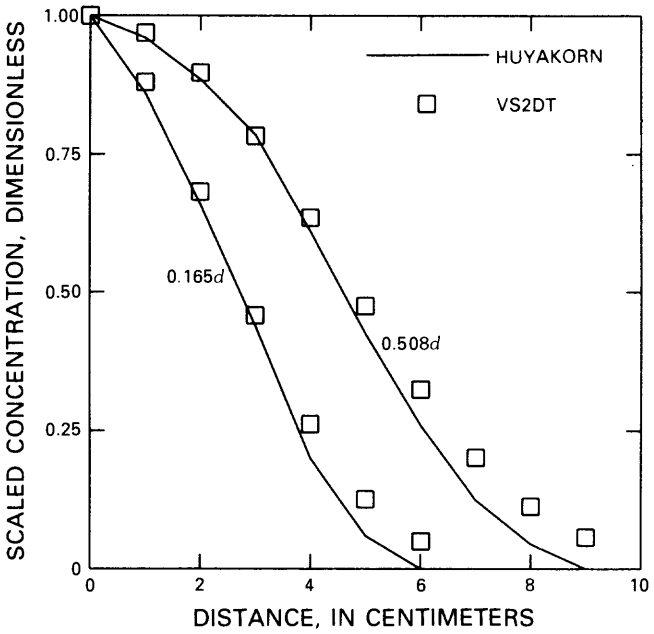


Figure 12.--Graph showing horizontal distribution of solute concentration for verification problem 4 for VS2DT, at a depth of 0.5 centimeter, and Huyakorn and others (1985) at a depth of 0 centimeter.

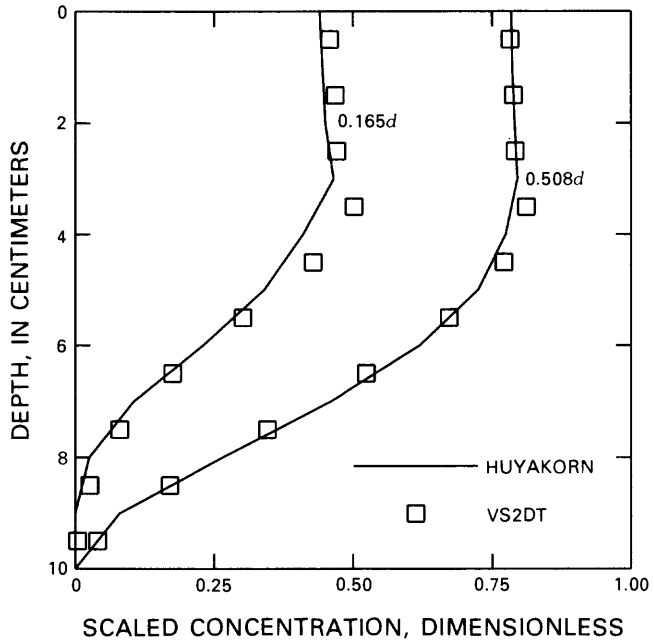


Figure 13.--Graph showing vertical distribution of solute concentration for verification problem 4 at a distance of 3 centimeters from left-hand boundary for VS2DT and Huyakorn and others (1985).

techniques than for block-centered finite-difference techniques, the horizontal profile in figure 12 is for a depth of 0. cm for the Huyakorn and others (1985) results and a depth of 0.5 cm for the VS2DT results. This explains the slightly greater concentrations predicted by VS2DT at larger distances from the left-hand boundary.

Verification Problem 5

The fifth verification problem involves one-dimensional transport through a saturated soil column with steady water flow and both first-order decay and linear sorption. The initial solute concentration in the column is zero. At times greater than zero, the inflowing water has a concentration of c_0. The analytical solution to this problem (assuming a semi-infinite column) is given by Bear (1972, p. 630) as:

$$c(z,t) = \frac{1}{2} c_0 e^{(vz/2D)}\{e^{-z\beta} \operatorname{erfc}[\frac{z - (v'^2+4\lambda D')^{1/2}t}{2(D't)^{1/2}}] +$$

$$e^{z\beta} \operatorname{erfc}[\frac{z + (v'^2+4\lambda D')^{1/2}t}{2(D't)^{1/2}}]\} \qquad (61)$$

where $\beta^2 = \left(\frac{v}{2D}\right)^2 + \frac{\lambda}{D'}$;

$v' = v/(1+\rho_b K_d/\theta_s)$; and

$D' = \alpha|v|/(1+\rho_b K_d/\theta_s)$.

The following constants were used:

length of column = 35 cm;
v = 0.1 cm/s;
α = 1.0 cm;
D_m = 0;
Δt = 1 s;
ρ_b = 1.587 gm/cm^3;
θ_s = 0.37;
Δz = 0.2 cm;
λ = 0.0, 0.01; and
K_d = 0.0, 0.3.

The water flowing into the column was maintained at a concentration of c_0 for 160 s, after which the concentration was set to zero for an additional 320 s. As shown in figure 14, the numerical results of VS2DT produce a good match with analytical results at a distance of 8 cm for the column inlet at all times for the three different cases.

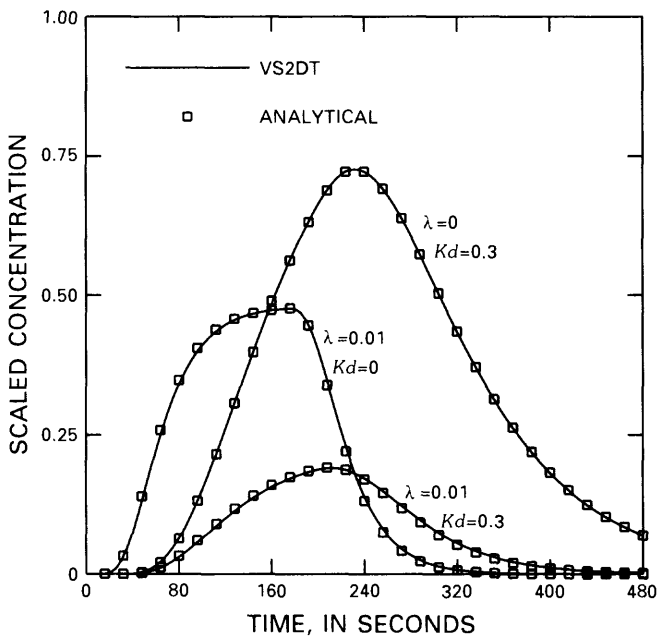


Figure 14.--Graph showing analytical and numerical results at distance of 8 centimeters from column inlet for fifth verification problem.

Example Problem

The purpose of this example is to demonstrate the data-input requirements and output listing for the simulator. The example can also be used as a test of the code after installation on any computer. The problem involves infiltration of water containing a solute concentration of 1.0 into a partially saturated one-dimensional soil column containing solute free water. The soil is a sandy loam with moisture and hydraulic-conductivity curves described by van Genuchten (1980):

$$\theta(h) = \theta_r + (\theta_s - \theta_r)\left[\frac{1}{1 + (\frac{h}{\alpha'})^{\beta'}}\right]^{\gamma}$$

$$K_r(h) = \frac{\{1-(\frac{h}{\alpha'})^{\beta'-1}[1+(\frac{h}{\alpha'})^{\beta'}]^{-\gamma}\}^2}{[1+(\frac{h}{\alpha'})^{\beta'}]^{\gamma/2}}$$

The following constants were used:

$\theta_s = 0.45$;
$\theta_r = 0.10$;
$\beta' = 2.75$;
$\alpha' = -40$ cm;
$\gamma = 0.64$;
$K = 6.25$ cm/h; and
$\alpha_L = 10$ cm.

The column is 40. cm in height. Uniform spacing (Δz = 1 cm) and time-step sizes (Δt = .005 hr) are used. Initial conditions were h = -120. cm and c = 0. everywhere. A constant flux of 5.5 cm/h was applied to the top of the column for a period of 0.50 h. The infiltrating water had a solute concentration of 1.0. A full listing of input data for the problem is shown in Table 4. Tables 5 and 6 show results printed to Fortran files 6 and 9, respectively.

Table 4.--*Input data for example problem*

EXAMPLE PROBLEM 1-D INFILTRATION	
0.50 0. 0.	A2--IMAX,STIM,ANG
CMHOURGRAM	A3--ZUNIT,TUNIT,CUNX
3 42	A4--NXR,NLY
2 600	A5--NRECH,NUMT
F T T	A6--RAD,ITSTOP,TRANS
T T F	A6A--CIS,CIT,SORP
F T T T F	A7--F11P,F7P,F8P,F9P,F6P
F F T F T	A8--THPT,SPNT,PPNT,HPNT,VPNT
1 1.	A9--IFAC,FACX
1 1.	A11-IFAC,FACZ
1	A13--NPLT
0.50	A14--PLTIM
6	A17--NMB9
31 33 40 42 70 72	A18--MB9
.0005 .90 0.00 0.00001	B1--EPS,HMAX,WUS,EPS1
2 100	B3--MINIT,ITMAX
T	B4--PHRD
1 6 7	B5--NTEX,NPROP,NPROP1
1	B6--ITEX
1. 10.0 0. .45 -40. .10 2.75	B7--ANIZ,HK
10. 0. 0.000001 0. 0. 0. 1.	B7A--HT
1	B8--IROW
1 3 42 1	B10--IL,IR,JBT,JRD
0 -120.	B11--IREAD,FACTOR
F F	B15--NPV,ETCYC
0 0.	B24--IREAD,FACTOR
0.50 .005	C1--TPER,DELT
1.0 0.005 0.005 0.0	C2--TMLT,DLTMX,DLTMIN,TRED
100. 0.	C3--DSMAX,STERR
0.	C4--POND
F	C5--PRNT
F F F	C6--BCIT,ETSIM,SEEP
0	C10--IBC
2 2 2 5.5 0 1.0	C11--JJ,NN,NTX,PFDUM,NTC,CF
999999 /	C13
999999 /	C13

Table 5.--*Output to file 6 for example problem*

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+                                                                      +
+                              VS2DT                                   +
+    SIMULATION OF 2-DIMENSIONAL VARIABLY                              +
+    SATURATED FLOW AND SOLUTE TRANSPORT                               +
+    THROUGH POROUS MEDIA.  VERSION DATED                              +
+                              4-1-90                                  +
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++


**********************************************************************************************************
    EXAMPLE PROBLEM 1-D INFILTRATION
**********************************************************************************************************

        SPACE AND TIME CONSTANTS
        ------------------------
    MAXIMUM SIMULATION TIME =     0.5000 HOUR
    STARTING TIME =     0.0000
    NUMBER OF RECHARGE PERIODS =         2
    MAXIMUM NUMBER OF TIME STEPS =        600
    NUMBER OF ROWS =    42
    NUMBER OF COLUMNS =     3
    AXES TILTED BY ANGLE =     0.00
        SOLUTION OPTIONS
        ----------------
    WRITE ALL PRESSURE HEADS TO FILE 8 AT OBSERVATION TIMES? T
    STOP SOLUTION IF MAXIMUM NO. OF ITERATIONS EXCEEDED IN ANY TIME STEP?,T
    WRITE MAXIMUM CHANGE IN HEAD FOR EACH ITERATION TO FILE 7? T
    WRITE RESULTS AT SELECTED OBSERVATION POINTS TO FILE 11? F
    WRITE MASS BALANCE RATES TO FILE 9? T
    WRITE MASS BALANCE RATES TO FILE 6? F
    WRITE MOISTURE CONTENTS TO FILE 6? F
    WRITE SATURATIONS TO FILE 6? F
    WRITE PRESSURE HEADS TO FILE 6? T
    WRITE TOTAL HEADS TO FILE 6? F
    WRITE VELOCITIES TO FILE 6? T
                                          GRID SPACING IN VERTICAL DIRECTION, IN   CM
    1.000     1.000     1.000     1.000     1.000     1.000     1.000     1.000     1.000     1.000
    1.000     1.000     1.000     1.000     1.000     1.000     1.000     1.000     1.000     1.000
    1.000     1.000     1.000     1.000     1.000     1.000     1.000     1.000     1.000     1.000
    1.000     1.000     1.000     1.000     1.000     1.000     1.000     1.000     1.000     1.000
    1.000     1.000
```

Table 5.--*Output to file 6 for example problem*--Continued

```
                                                GRID SPACING IN HORIZONTAL OR RADIAL DIRECTION, IN   CM
    1.000     1.000     1.000
    TIMES AT WHICH H WILL BE WRITTEN TO FILE 08
        0.5000
    MASS BALANCE COMPONENTS WRITTEN TO FILE 9
      31  33  40  42  70  72
    COORDINATE SYSTEM IS RECTANGULAR
    TRANSPORT TO BE SIMULATED
    CENTRAL DIFFERENCING IN SPACE USED FOR TRANSPORT EQUATION
    CENTRAL DIFFERENCING IN TIME USED FOR TRANSPORT EQUATION
    MATRIX EQUATIONS TO BE SOLVED BY SIP
         INITIAL MOISTURE PARAMETERS
         ---------------------------

    CONVERGENCE CRITERIA FOR SIP FOR FLOW =   5.000E-04   CM
    CONVERGENCE CRITERIA FOR SIP FOR TRANSPORT =   1.000E-05
    DAMPING FACTOR, HMAX =    9.000E-01
    GEOMETRIC MEAN USED FOR INTERCELL CONDUCTIVITY
    NUMBER OF SOIL TEXTURAL CLASSES =          1
    NUMBER OF SOIL PARAMETERS FOR EACH CLASS =          6
    NUMBER OF TRANSPORT PARAMETERS FOR EACH CLASS =          7
    MINIMUM PERMITTED NO. OF ITERATIONS/TIME STEP =          2
    MAXIMUM PERMITTED NO. OF ITERATIONS/TIME STEP =        100
                                    CONSTANTS FOR SOIL TEXTURAL CLASSES

         ANISOTROPY        KSAT      SPECIFIC     POROSITY
                                     STORAGE
           ALPHAL          ALPHAT       DM          LAMBDA     B DENSITY
CLASS # 1
           1.000D+00    1.000D+01    0.000D-01    4.500D-01   -4.000D+01    1.000D-01    2.750D+00
           1.000D+01    0.000D-01    1.000D-06    0.000D-01    0.000D-01    0.000D-01    1.000D+00
     TEXTURAL CLASS INDEX MAP


    TEXTURAL CLASSES READ IN BY BLOCK
        1  111
        2  111
        3  111
        4  111
        5  111
        6  111
        7  111
        8  111
```

Table 5.--*Output to file 6 for example problem*--Continued

```
  9  111
 10  111
 11  111
 12  111
 13  111
 14  111
 15  111
 16  111
 17  111
 18  111
 19  111
 20  111
 21  111
 22  111
 23  111
 24  111
 25  111
 26  111
 27  111
 28  111
 29  111
 30  111
 31  111
 32  111
 33  111
 34  111
 35  111
 36  111
 37  111
 38  111
 39  111
 40  111
 41  111
 42  111
INITIAL PRESSURE HEAD OR MOISTURE CONTENT WAS SET TO A CONSTANT VALUE OF  -1.200E+02
INITIAL CONCENTRATION SET TO A CONSTANT VALUE OF    0.000E-01
5SIP ITERATION PARAMETERS:  0.1421085D-13  0.8131982D+00  0.9651051D+00  0.9934816D+00  0.9987823D+00
 EXAMPLE PROBLEM 1-D INFILTRATION
TOTAL ELAPSED TIME =   0.000E-01 HOUR
TIME STEP     0

                                PRESSURE HEAD
```

Table 5.--*Output to file 6 for example problem*--Continued

```
Z, IN
  CM                    X OR R DISTANCE, IN   CM
                0.50
   0.50-1.20E+02
   1.50-1.20E+02
   2.50-1.20E+02
   3.50-1.20E+02
   4.50-1.20E+02
   5.50-1.20E+02
   6.50-1.20E+02
   7.50-1.20E+02
   8.50-1.20E+02
   9.50-1.20E+02
  10.50-1.20E+02
  11.50-1.20E+02
  12.50-1.20E+02
  13.50-1.20E+02
  14.50-1.20E+02
  15.50-1.20E+02
  16.50-1.20E+02
  17.50-1.20E+02
  18.50-1.20E+02
  19.50-1.20E+02
  20.50-1.20E+02
  21.50-1.20E+02
  22.50-1.20E+02
  23.50-1.20E+02
  24.50-1.20E+02
  25.50-1.20E+02
  26.50-1.20E+02
  27.50-1.20E+02
  28.50-1.20E+02
  29.50-1.20E+02
  30.50-1.20E+02
  31.50-1.20E+02
  32.50-1.20E+02
  33.50-1.20E+02
  34.50-1.20E+02
  35.50-1.20E+02
  36.50-1.20E+02
  37.50-1.20E+02
  38.50-1.20E+02
  39.50-1.20E+02
```

Table 5.--*Output to file 6 for example problem*--Continued

CONCENTRATION

Z, IN CM	X OR R DISTANCE, IN CM
	0.50
0.50	0.00E-01
1.50	0.00E-01
2.50	0.00E-01
3.50	0.00E-01
4.50	0.00E-01
5.50	0.00E-01
6.50	0.00E-01
7.50	0.00E-01
8.50	0.00E-01
9.50	0.00E-01
10.50	0.00E-01
11.50	0.00E-01
12.50	0.00E-01
13.50	0.00E-01
14.50	0.00E-01
15.50	0.00E-01
16.50	0.00E-01
17.50	0.00E-01
18.50	0.00E-01
19.50	0.00E-01
20.50	0.00E-01
21.50	0.00E-01
22.50	0.00E-01
23.50	0.00E-01
24.50	0.00E-01
25.50	0.00E-01
26.50	0.00E-01
27.50	0.00E-01
28.50	0.00E-01
29.50	0.00E-01
30.50	0.00E-01
31.50	0.00E-01
32.50	0.00E-01
33.50	0.00E-01
34.50	0.00E-01
35.50	0.00E-01
36.50	0.00E-01
37.50	0.00E-01
38.50	0.00E-01

Table 5.--*Output to file 6 for example problem*--Continued

```
39.50 0.00E-01
  DATA FOR RECHARGE PERIOD     1

      LENGTH OF THIS PERIOD =   5.000E-01 HOUR
      LENGTH OF INITIAL TIME STEP FOR THIS PERIOD = 5.000E-03 HOUR
      MULTIPLIER FOR TIME STEP =  1.000E+00
      MAXIMUM TIME STEP SIZE =  5.000E-03 HOUR
      MINIMUM TIME STEP SIZE =  5.000E-03 HOUR
      TIME STEP REDUCTION FACTOR =  0.000E-01
           MAXIMUM PRESSURE HEAD CHANGE ALLOWED IN ONE TIME STEP = 100.000
           STEADY-STATE CLOSURE CRITERION =  0.000E-01
           MAXIMUM DEPTH OF PONDING =  0.000E-01
           PRINT SOLUTION AFTER EVERY TIME STEP? F
           SIMULATE EVAPORATION? F
           SIMULATE EVAPOTRANSPIRATION? F
           SIMULATE SEEPAGE FACES? F

  NODE TYPE AND INITIAL BOUNDARY CONDITIONS FOR PERIOD    1
  LEGEND:
           0 = INTERIOR CELL
           1 = SPECIFIED PRESSURE HEAD CELL
           2 = SPECIFIED FLUX CELL
           3 = POTENTIAL SEEPAGE FACE NODE
           5 = NODE FOR WHICH EVAPORATION IS PERMITTED


  1     000
  2     020
  3     000
  4     000
  5     000
  6     000
  7     000
  8     000
  9     000
 10     000
 11     000
 12     000
 13     000
 14     000
 15     000
 16     000
 17     000
```

Table 5.--*Output to file 6 for example problem*--Continued

```
   18     000
   19     000
   20     000
   21     000
   22     000
   23     000
   24     000
   25     000
   26     000
   27     000
   28     000
   29     000
   30     000
   31     000
   32     000
   33     000
   34     000
   35     000
   36     000
   37     000
   38     000
   39     000
   40     000
   41     000
   42     000
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.10000E+00     ROW   41  COLUMN    2
TIME STEP     1  TIME =   0.5000E-02  NIT =  15  NIT1 =   6
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.10000E+00     ROW   41  COLUMN    2
TIME STEP     2  TIME =   0.1000E-01  NIT =   9  NIT1 =   6
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.10000E+00     ROW   41  COLUMN    2
TIME STEP     3  TIME =   0.1500E-01  NIT =   9  NIT1 =   6
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.10000E+00     ROW   41  COLUMN    2
TIME STEP     4  TIME =   0.2000E-01  NIT =   9  NIT1 =   6
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.10000E+00     ROW   41  COLUMN    2
TIME STEP     5  TIME =   0.2500E-01  NIT =  10  NIT1 =   6
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.99999E-01     ROW   41  COLUMN    2
TIME STEP     6  TIME =   0.3000E-01  NIT =  10  NIT1 =   6
```

Table 5.--*Output to file 6 for example problem*--Continued

```
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.99999E-01     ROW   41  COLUMN    2
TIME STEP     7  TIME =   0.3500E-01  NIT =  10  NIT1 =   6
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.99999E-01     ROW   41  COLUMN    2
TIME STEP     8  TIME =   0.4000E-01  NIT =   9  NIT1 =   6
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.99999E-01     ROW   41  COLUMN    2
TIME STEP     9  TIME =   0.4500E-01  NIT =   9  NIT1 =   6
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.99999E-01     ROW   41  COLUMN    2
TIME STEP    10  TIME =   0.5000E-01  NIT =  10  NIT1 =   6
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.99999E-01     ROW   41  COLUMN    2
               .
               .
               .
TIME STEP    91  TIME =   0.4550E+00  NIT =   8  NIT1 =   5
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.99999E-01     ROW   41  COLUMN    2
TIME STEP    92  TIME =   0.4600E+00  NIT =   8  NIT1 =   5
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.99999E-01     ROW   41  COLUMN    2
TIME STEP    93  TIME =   0.4650E+00  NIT =   8  NIT1 =   5
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.99999E-01     ROW   41  COLUMN    2
TIME STEP    94  TIME =   0.4700E+00  NIT =   8  NIT1 =   5
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.99999E-01     ROW   41  COLUMN    2
TIME STEP    95  TIME =   0.4750E+00  NIT =   8  NIT1 =   5
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.99999E-01     ROW   41  COLUMN    2
TIME STEP    96  TIME =   0.4800E+00  NIT =   8  NIT1 =   5
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.99999E-01     ROW   41  COLUMN    2
TIME STEP    97  TIME =   0.4850E+00  NIT =   8  NIT1 =   5
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.99999E-01     ROW   41  COLUMN    2
TIME STEP    98  TIME =   0.4900E+00  NIT =   8  NIT1 =   5
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
                                     VERTICAL       0.99999E-01     ROW   41  COLUMN    2
TIME STEP    99  TIME =   0.4950E+00  NIT =   8  NIT1 =   5
     MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL     0.00000E+00     ROW    0  COLUMN    0
```

Table 5.--*Output to file 6 for example problem*--Continued

```
                                    VERTICAL       0.99999E-01     ROW   41  COLUMN    2
TIME STEP   100  TIME =   0.5000E+00  NIT =   8  NIT1 =   5
     EXAMPLE PROBLEM 1-D INFILTRATION
    TOTAL ELAPSED TIME =   5.000E-01 HOUR
    TIME STEP   100


                                                PRESSURE HEAD

 Z, IN
   CM                  X OR R DISTANCE, IN   CM
             0.50
    0.50-2.66E+01
    1.50-2.73E+01
    2.50-2.82E+01
    3.50-2.91E+01
    4.50-3.02E+01
    5.50-3.14E+01
    6.50-3.28E+01
    7.50-3.45E+01
    8.50-3.65E+01
    9.50-3.89E+01
   10.50-4.20E+01
   11.50-4.62E+01
   12.50-5.22E+01
   13.50-6.22E+01
   14.50-8.44E+01
   15.50-1.11E+02
   16.50-1.19E+02
   17.50-1.20E+02
   18.50-1.20E+02
   19.50-1.20E+02
   20.50-1.20E+02
   21.50-1.20E+02
   22.50-1.20E+02
   23.50-1.20E+02
   24.50-1.20E+02
   25.50-1.20E+02
   26.50-1.20E+02
   27.50-1.20E+02
   28.50-1.20E+02
   29.50-1.20E+02
   30.50-1.20E+02
   31.50-1.20E+02
```

Table 5.--*Output to file 6 for example problem*--Continued

```
  32.50-1.20E+02
  33.50-1.20E+02
  34.50-1.20E+02
  35.50-1.20E+02
  36.50-1.20E+02
  37.50-1.20E+02
  38.50-1.19E+02
  39.50-1.19E+02
                                                X-VELOCITY
Z, IN
  CM                    X OR R DISTANCE, IN   CM
                0.50
   0.50 0.00E-01
   1.50 0.00E-01
   2.50 0.00E-01
   3.50 0.00E-01
   4.50 0.00E-01
   5.50 0.00E-01
   6.50 0.00E-01
   7.50 0.00E-01
   8.50 0.00E-01
   9.50 0.00E-01
  10.50 0.00E-01
  11.50 0.00E-01
  12.50 0.00E-01
  13.50 0.00E-01
  14.50 0.00E-01
  15.50 0.00E-01
  16.50 0.00E-01
  17.50 0.00E-01
  18.50 0.00E-01
  19.50 0.00E-01
  20.50 0.00E-01
  21.50 0.00E-01
  22.50 0.00E-01
  23.50 0.00E-01
  24.50 0.00E-01
  25.50 0.00E-01
  26.50 0.00E-01
  27.50 0.00E-01
  28.50 0.00E-01
  29.50 0.00E-01
  30.50 0.00E-01
```

Table 5.--*Output to file 6 for example problem*--Continued

31.50	0.00E-01
32.50	0.00E-01
33.50	0.00E-01
34.50	0.00E-01
35.50	0.00E-01
36.50	0.00E-01
37.50	0.00E-01
38.50	0.00E-01
39.50	0.00E-01

Z-VELOCITY

Z, IN CM	X OR R DISTANCE, IN CM
	0.50
0.50	0.00E-01
1.50	1.39E+01
2.50	1.39E+01
3.50	1.38E+01
4.50	1.37E+01
5.50	1.36E+01
6.50	1.35E+01
7.50	1.33E+01
8.50	1.31E+01
9.50	1.28E+01
10.50	1.24E+01
11.50	1.19E+01
12.50	1.10E+01
13.50	9.66E+00
14.50	6.89E+00
15.50	1.96E+00
16.50	2.43E-01
17.50	4.11E-02
18.50	2.39E-02
19.50	2.25E-02
20.50	2.24E-02
21.50	2.24E-02
22.50	2.24E-02
23.50	2.24E-02
24.50	2.24E-02
25.50	2.24E-02
26.50	2.24E-02
27.50	2.24E-02
28.50	2.24E-02
29.50	2.24E-02

Table 5.--*Output to file 6 for example problem*--Continued

```
 30.50 2.24E-02
 31.50 2.24E-02
 32.50 2.24E-02
 33.50 2.23E-02
 34.50 2.22E-02
 35.50 2.18E-02
 36.50 2.09E-02
 37.50 1.87E-02
 38.50 1.45E-02
 39.50 8.11E-03
                                          CONCENTRATION
Z, IN
  CM                  X OR R DISTANCE, IN   CM
              0.50
  0.50 7.31E-01
  1.50 7.05E-01
  2.50 6.77E-01
  3.50 6.49E-01
  4.50 6.20E-01
  5.50 5.91E-01
  6.50 5.61E-01
  7.50 5.30E-01
  8.50 4.98E-01
  9.50 4.65E-01
 10.50 4.31E-01
 11.50 3.94E-01
 12.50 3.54E-01
 13.50 3.08E-01
 14.50 2.41E-01
 15.50 6.43E-02
 16.50 1.39E-03
 17.50 4.18E-06
 18.50 7.15E-09
 19.50 1.15E-11
 20.50 1.83E-14
 21.50 2.90E-17
 22.50 4.59E-20
 23.50 7.24E-23
 24.50 1.14E-25
 25.50 1.79E-28
 26.50 2.81E-31
 27.50 4.39E-34
 28.50 6.84E-37
```

Table 5.--*Output to file 6 for example problem*--Continued

```
29.50 1.06E-39
30.50 1.65E-42
31.50 2.55E-45
32.50 3.92E-48
33.50 6.01E-51
34.50 9.14E-54
35.50 1.36E-56
36.50 1.94E-59
37.50 2.47E-62
38.50 2.45E-65
39.50 1.36E-68
                  ---------- MASS BALANCE SUMMARY FOR TIME STEP 100 ----------
                      PUMPING PERIOD NUMBER    1
                      TOTAL ELAPSED SIMULATION TIME =  5.000E-01 HOUR

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+                                                                                                                              +
+                                                                                        TOTAL THIS          RATE THIS         +
+                          VOLUMETRIC FLOW BALANCE                        TOTAL          TIME STEP           TIME STEP         +
+                                                                         CM**3            CM**3             CM**3/HOUR        +
+  FLUX INTO DOMAIN ACROSS SPECIFIED PRESSURE HEAD BOUNDARIES --       0.00000E-01      0.00000E-01         0.00000E-01       +
+ FLUX OUT OF DOMAIN ACROSS SPECIFIED PRESSURE HEAD BOUNDARIES --      0.00000E-01      0.00000E-01         0.00000E-01       +
+           FLUX INTO DOMAIN ACROSS SPECIFIED FLUX BOUNDARIES --       2.75000E+00      2.75000E-02         5.50000E+00       +
+         FLUX OUT OF DOMAIN ACROSS SPECIFIED FLUX BOUNDARIES --       0.00000E-01      0.00000E-01         0.00000E-01       +
+                                     TOTAL FLUX INTO DOMAIN --       2.75000E+00      2.75000E-02         5.50000E+00       +
+                                   TOTAL FLUX OUT OF DOMAIN --       0.00000E-01      0.00000E-01         0.00000E-01       +
+                                                EVAPORATION --       0.00000E-01      0.00000E-01         0.00000E-01       +
+                                              TRANSPIRATION --       0.00000E-01      0.00000E-01         0.00000E-01       +
+                                   TOTAL EVAPOTRANSPIRATION --       0.00000E-01      0.00000E-01         0.00000E-01       +
+                           CHANGE IN FLUID STORED IN DOMAIN --       2.75000E+00      2.75000E-02         5.49999E+00       +
+                                       FLUID VOLUME BALANCE --       3.10821E-06      4.34478E-08         8.68957E-06       +
+                                                                                                                              +
+                                                                                                                              +
+                            SOLUTE MASS BALANCE                                                                               +
+                                                                         GRAM             GRAM              GRAM/HOUR         +
+  FLUX INTO DOMAIN ACROSS SPECIFIED PRESSURE HEAD BOUNDARIES --       0.00000E-01      0.00000E-01         0.00000E-01       +
+ FLUX OUT OF DOMAIN ACROSS SPECIFIED PRESSURE HEAD BOUNDARIES --      0.00000E-01      0.00000E-01         0.00000E-01       +
+           FLUX INTO DOMAIN ACROSS SPECIFIED FLUX BOUNDARIES --       2.75000E+00      2.75000E-02         5.50000E+00       +
+         FLUX OUT OF DOMAIN ACROSS SPECIFIED FLUX BOUNDARIES --       0.00000E-01      0.00000E-01         0.00000E-01       +
+                      DIFFUSIVE/DISPERSIVE FLUX INTO DOMAIN --       0.00000E-01      0.00000E-01         0.00000E-01       +
+                    DIFFUSIVE/DISPERSIVE FLUX OUT OF DOMAIN --       0.00000E-01      0.00000E-01         0.00000E-01       +
+                                     TOTAL FLUX INTO DOMAIN --       2.75000E+00      2.75000E-02         5.50000E+00       +
+                                   TOTAL FLUX OUT OF DOMAIN --       0.00000E-01      0.00000E-01         0.00000E-01       +
```

Table 5.--*Output to file 6 for example problem*--Continued

```
+                                  TOTAL EVAPOTRANSPIRATION --      0.00000E-01          0.00000E-01          0.00000E-01    +
+                                        FIRST ORDER DECAY --      0.00000E-01          0.00000E-01          0.00000E-01    +
+                                   ADSORPTION/ION EXCHANGE --      0.00000E-01          0.00000E-01          0.00000E-01    +
+                               CHANGE IN SOLUTE STORED IN DOMAIN --      2.71061E+00          2.73234E-02          5.46469E+00    +
+                                      SOLUTE MASS BALANCE --      3.93940E-02          1.76551E-04          3.53103E-02    +
+                                                                                                                              +
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
   ********************************************************************************************************
   END OF SIMULATION
   ********************************************************************************************************
   TOTAL NUMBER OF ITERATIONS FOR FLOW EQUATION =    842
   TOTAL NUMBER OF ITERATIONS FOR TRANSPORT EQUATION =    543
```

Table 6.--*Output to file 9 for example problem*

5.000E-03	-3.103E-07	-6.205E-05	2.750E-02	5.500E+00	3.428E-03	6.857E-01
1.000E-02	-7.551E-07	-8.897E-05	5.500E-02	5.500E+00	5.719E-03	4.581E-01
1.500E-02	-9.124E-07	-3.145E-05	8.250E-02	5.500E+00	7.266E-03	3.094E-01
2.000E-02	-9.552E-07	-8.564E-06	1.100E-01	5.500E+00	8.465E-03	2.399E-01
2.500E-02	-8.418E-07	2.267E-05	1.375E-01	5.500E+00	9.596E-03	2.262E-01
3.000E-02	-6.997E-07	2.843E-05	1.650E-01	5.500E+00	1.062E-02	2.055E-01
3.500E-02	-6.579E-07	8.364E-06	1.925E-01	5.500E+00	1.147E-02	1.686E-01
4.000E-02	-6.893E-07	-6.282E-06	2.200E-01	5.500E+00	1.219E-02	1.451E-01
4.500E-02	-7.285E-07	-7.832E-06	2.475E-01	5.500E+00	1.291E-02	1.443E-01
5.000E-02	-7.079E-07	4.119E-06	2.750E-01	5.500E+00	1.365E-02	1.477E-01
5.500E-02	-5.442E-07	3.272E-05	3.025E-01	5.500E+00	1.434E-02	1.381E-01
6.000E-02	-5.051E-07	7.821E-06	3.300E-01	5.500E+00	1.494E-02	1.192E-01
6.500E-02	-5.111E-07	-1.191E-06	3.575E-01	5.500E+00	1.547E-02	1.059E-01
7.000E-02	-5.488E-07	-7.545E-06	3.850E-01	5.500E+00	1.599E-02	1.053E-01
7.500E-02	-5.332E-07	3.116E-06	4.125E-01	5.500E+00	1.655E-02	1.107E-01
8.000E-02	-4.952E-07	7.600E-06	4.400E-01	5.500E+00	1.710E-02	1.108E-01
8.500E-02	-4.168E-07	1.569E-05	4.675E-01	5.500E+00	1.761E-02	1.021E-01
9.000E-02	-4.026E-07	2.832E-06	4.950E-01	5.500E+00	1.807E-02	9.058E-02
9.500E-02	-4.041E-07	-3.022E-07	5.225E-01	5.500E+00	1.849E-02	8.437E-02
1.000E-01	-4.217E-07	-3.517E-06	5.500E-01	5.500E+00	1.892E-02	8.574E-02
	.					
	.					
	.					
4.050E-01	2.384E-06	8.108E-06	2.227E+00	5.500E+00	3.581E-02	4.190E-02
4.100E-01	2.436E-06	1.029E-05	2.255E+00	5.500E+00	3.601E-02	4.112E-02
4.150E-01	2.481E-06	9.026E-06	2.282E+00	5.500E+00	3.621E-02	3.947E-02
4.200E-01	2.527E-06	9.248E-06	2.310E+00	5.500E+00	3.640E-02	3.812E-02
4.250E-01	2.560E-06	6.593E-06	2.337E+00	5.500E+00	3.659E-02	3.729E-02
4.300E-01	2.593E-06	6.615E-06	2.365E+00	5.500E+00	3.678E-02	3.765E-02
4.350E-01	2.620E-06	5.368E-06	2.392E+00	5.500E+00	3.697E-02	3.846E-02
4.400E-01	2.656E-06	7.109E-06	2.420E+00	5.500E+00	3.717E-02	3.951E-02
4.450E-01	2.692E-06	7.311E-06	2.447E+00	5.500E+00	3.737E-02	3.971E-02
4.500E-01	2.739E-06	9.387E-06	2.475E+00	5.500E+00	3.756E-02	3.922E-02
4.550E-01	2.781E-06	8.478E-06	2.502E+00	5.500E+00	3.775E-02	3.782E-02
4.600E-01	2.826E-06	8.928E-06	2.530E+00	5.500E+00	3.793E-02	3.655E-02
4.650E-01	2.859E-06	6.594E-06	2.557E+00	5.500E+00	3.811E-02	3.559E-02
4.700E-01	2.892E-06	6.543E-06	2.585E+00	5.500E+00	3.829E-02	3.571E-02
4.750E-01	2.917E-06	5.130E-06	2.612E+00	5.500E+00	3.847E-02	3.631E-02
4.800E-01	2.950E-06	6.471E-06	2.640E+00	5.500E+00	3.866E-02	3.734E-02
4.850E-01	2.982E-06	6.529E-06	2.667E+00	5.500E+00	3.885E-02	3.774E-02
4.900E-01	3.025E-06	8.512E-06	2.695E+00	5.500E+00	3.904E-02	3.758E-02
4.950E-01	3.065E-06	7.964E-06	2.722E+00	5.500E+00	3.922E-02	3.648E-02
5.000E-01	3.108E-06	8.690E-06	2.750E+00	5.500E+00	3.939E-02	3.531E-02

SUMMARY

A computer program, VS2DT, has been developed and tested for simulating solute transport in variably saturated porous media. The program is an extension to the U.S. Geological Survey's computer program VS2D for simulating water movement through variably saturated porous media. The finite-difference method is used to solve the advection-dispersion equation. The user may select either backward or centered approximations for time and space derivatives. The program also allows the following processes to be simulated: first-order decay of the solute, equilibrium adsorption of solute to the solid phase (as described by Freundlich or Langmuir isotherms), and ion exchange. The ability of the program to accurately match analytical results and results of other simulations is demonstrated with five verification problems.

The computer program is written in standard FORTRAN77 and is modular in structure. It can easily be modified or customized for particular applications. Modifications to the original version of VS2D are described as Supplemental Information. A complete listing of VS2DT is given, as well as data input requirements and listings of input and output for an example problem.

REFERENCES

Bear, Jacob, 1972, Dynamics of fluids in porous media: New York, American Elsevier Publishing Co., 764 p.

Bear, Jacob, 1979, Hydraulics of groundwater: New York, McGraw-Hill, 569 p.

Chiou, C.T., Porter, P.E., and Schmedding, D.W., 1983, Partition equilibria of nonionic organic compounds between soil organic matter and water: Environmental Science and Technology, v. 16, p. 4-10.

Corwin, T., and Hansch, C., eds., 1979, Substituents contents for correlation and level analysis in chemistry and biology: New York, John Wiley and Sons.

Gelhar, L. W., and Collins, M.A., 1971, General analysis of longitudinal dispersion in nonuniform flow: Water Resources Research, v. 7, no. 6, p. 1511-1521.

Freeze, R.A., and Cherry, J.A., 1979, Groundwater: Englewood Cliffs, N.J., Prentice-Hall, Inc., 604 p.

Grove, D.B., and Stollenwerk, K.G., 1984, Computer model of one-dimensional equilibrium controlled sorption processes: U.S. Geological Survey Water-Resources Investigations Report 84-4059, 58 p.

Healy, R.W., 1987, Simulation of trickle irrigation, an extension to the U.S. Geological Survey's computer program VS2D: U.S. Geological Survey Water Resources Investigations Report 87-4086, 61 p.

Hillel, Daniel, 1980, Fundamentals of soil physics: New York, Academic Press, 413 p.

Hoopes, J.A., and Harleman, D.R., 1967, Dispersion in radial flow from a recharge well: Journal of Geophysical Research, v. 72, no. 14, p. 3595-3607.

Hsieh, P.A., 1986, A new formula for the analytical solution of the radial dispersion problem: Water Resources Research, v. 22, no. 11, p. 1,597-1,605.

Huyakorn, P.S., Mercer, J.W., and Ward, D.S., 1985, Finite element matrix and mass balance computational schemes for transport in variably saturated porous media: Water Resources Research, v. 21, no. 3, p. 346-358.

Jury, W.A., Spencer, W.F., and Farmer, W.J., 1983, Behavior assessment model for trace organics in soils: I. Model description: Soil Science Society of America Journal, v. 12, p. 558-564.

Karickhoff, S.W., 1981, Semi-empirical estimation of sorption of hydrophobic pollutants on natural sediments: Chemisphere, v. 10, p. 833-846.

Kipp, K.L., 1987, HST3D: A computer code for simulation of heat and solute transport in three-dimensional ground-water flow systems: U.S. Geological Survey Water-Resources Investigations Report 86-4095, 517 p.

Konikow, L.F., and Grove, D.B., 1977, Derivation of equations describing solute transport in ground water: U.S. Geological Survey Water-Resources Investigations Report 77-19, 30 p.

Kwicklis, E.M., 1987, Development of the numerical simulator VSFAST (variably saturated flow and solute transport): A computer code to solve the governing equations in partially or wholly saturated media: Boulder, University of Colorado Master's Thesis, 310 p.

Lappala, E.G., Healy, R.W., and Weeks, E.P., 1987, Documentation of computer program VS2D to solve the equations of fluid flow in variably saturated porous media: U.S. Geological Survey Water-Resources Investigations Report 83-4099, 184 p.

Ogata, Akio, and Banks, R.B., 1961, A solution of the differential equation of longitudinal dispersion in porous media: U.S. Geological Survey Professional Paper 411-A, 7 p.

Rao, P.S.C., and Davidson, J.M., 1980, Estimation of pesticide retention and transformation parameters required in nonpoint source pollution models, in, Overcash, M.R., ed., Environmental impact of nonpoint source pollution: Ann Arbor, Michigan, Ann Arbor Science Publishing, Inc., p. 23-67.

Scheidegger, A.E., 1961, General theory of dispersion in porous media: Journal of Geophysical Research, v. 66, no. 10, p. 3273-3278.

Stumm, Werner, and Morgan, J.J., 1981, Aquatic chemistry: New York, John Wiley and Sons, 780 p.

Tang, D.H., and Babu, D.K., 1979, Analytical solution of a velocity dependent dispersion problem: Water Resources Research, v. 15, no. 6, p. 1,471-1,478.

van Genuchten, M.Th., 1980, A closed-form equation for predicting the hydraulic conductivity of unsaturated soils: Soil Science Society of America Proceedings, v. 44, no. 5, p. 892-898.

van Genuchten, M.Th., 1982, A comparison of numerical solutions of the one-dimensional unsaturated-saturated flow and mass transport equations: Advances in Water Resources, v. 5, no. 1, p. 47-55.

Voss, C.I., 1984, A finite-element simulation model for saturated-unsaturated, fluid-density-dependent ground-water flow with energy transport or chemically reactive single-species solute transport: U.S. Geological Survey Water-Resources Investigations Report 84-4369, 409 p.

Warrick, A.W., Biggar, J.W., and Nielsen, D.R., 1971, Simultaneous solute and water transfer for an unsaturated soil: Water Resources Research, v. 7, no. 5, p. 1216-1225.

SUPPLEMENTAL INFORMATION

Three items are presented in this section. The first is a description of recent modifications to VS2D other than those related to the solute transport option. The second item is a complete listing of the revised version of VS2DT. The final item is a flow chart for VS2DT.

Modifications to Computer Program VS2D

In an effort to improve the efficiency and usefulness of computer program VS2DT, several minor modifications have been incorporated into the original version of the code as listed in Lappala and others (1987). These are detailed below.

(1) The x and z axes may now be tilted for a simulation. This option requires input of the angle of rotation (ANG on card A-2), which is referenced from horizontal. ANG = 0 corresponds to no tilting. Figure 15 illustrates how the finite-difference grid is treated in the program for different rotation angles. ANG must be between -90 and +90 degrees. Because elevation is an important factor in the infiltration/ponding and seepage face boundaries, incorporation of the tilted-axes option required that the subroutines VSPOND and VSFAC be rewritten. The new versions are contained in the following program listing. The algorithms used in these subroutines are still identical to those described in Lappala and others (1987). Because cross-derivative terms are not included in the finite-difference approximation to the flow equation, it is necessary that the principal directions of the hydraulic-conductivity tensor be aligned with the coordinate axes. Therefore, the value for HK(ITEX,1), on input line B-7, must correspond to the saturated hydraulic conductivity in the direction of the tilted x axis. Similarly, the value for ANIZ(ITEX), on the same input line, must represent the ratio of hydraulic conductivity in the direction of the tilted z axis to that in the direction of the tilted x axis.

(2) Selection of mass-balance components for output for file 9 is now a user option. There are 72 components that can be selected. These are listed in table 7, along with the index number that must be included on input card A-18. A maximum of 24 components may be selected for any simulation. The output format for each component is E11.4. The first item in each output line is simulated time. Mass balance information is written to file 9 at the end of every time step. It is anticipated that file 9 results will be used primarily for generating computer plots, therefore no column headings are included in the file.

(3) Fluid mass balance is now given in terms of volume rather than mass. Therefore the variable RHOZ is no longer used in the program and input card B-2 must not appear in the input data stream.

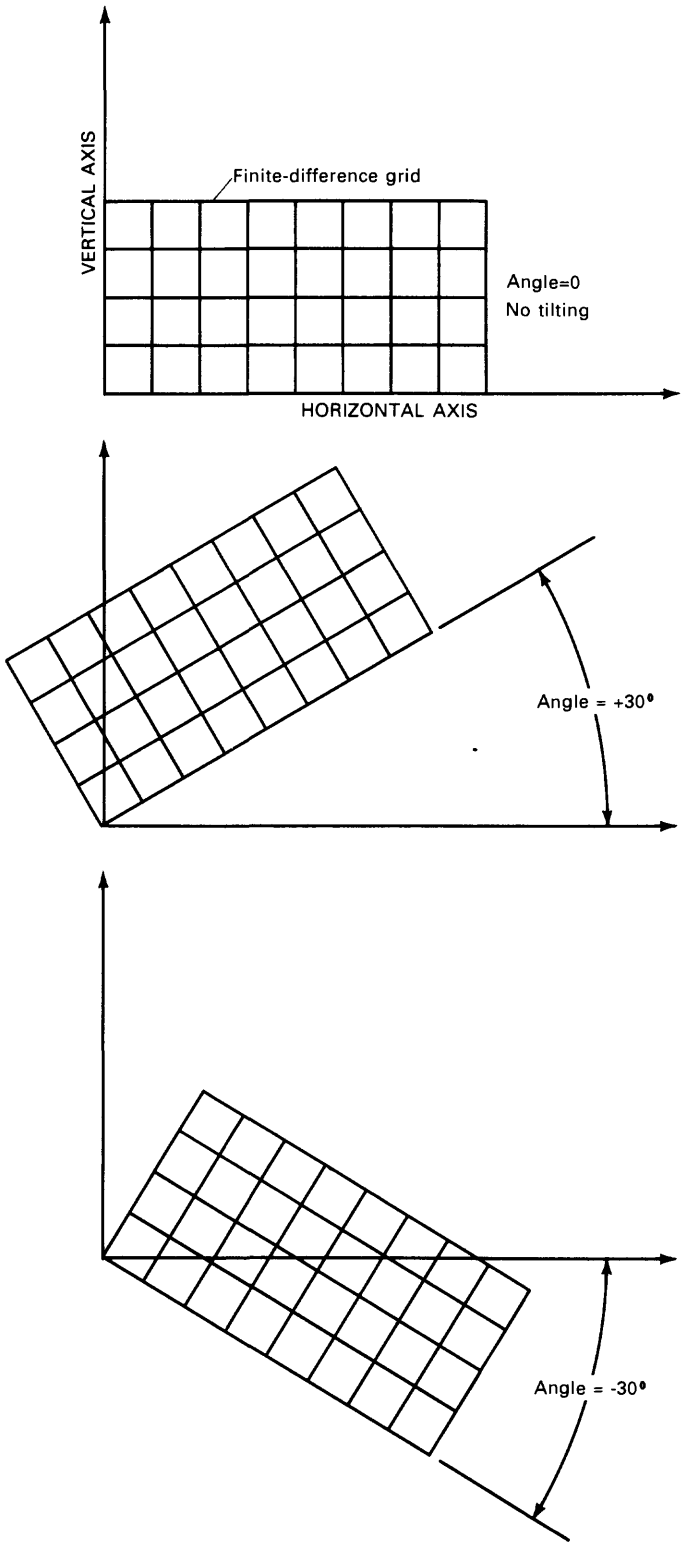


Figure 15.--Sketch showing tilting of finite-difference grid for different angles.

Table 7.--*Index of Mass-Balance Components for Output to File 9*

Index Number	Component	
1	Flow in across specified head boundaries	- total for simulation
2	Flow in across specified head boundaries	- total for time step
3	Flow in across specified head boundaries	- rate for time step
4	Flow out across specified head boundaries	- total for simulation
5	Flow out across specified head boundaries	- total for time step
6	Flow out across specified head boundaries	- rate for time step
7	Flow in across specified flux boundaries	- total for simulation
8	Flow in across specified flux boundaries	- total for time step
9	Flow in across specified flux boundaries	- rate for time step
10	Flow out across specified flux boundaries	- total for simulation
11	Flow out across specified flux boundaries	- total for time step
12	Flow out across specified flux boundaries	- rate for time step
13	Total flow in	- total for simulation
14	Total flow in	- total for time step
15	Total flow in	- rate for time step
16	Total flow out	- total for simulation
17	Total flow out	- total for time step
18	Total flow out	- rate for time step
19	Evaporation	- total for simulation
20	Evaporation	- total for time step
21	Evaporation	- rate for time step
22	Transpiration	- total for simulation
23	Transpiration	- total for time step
24	Transpiration	- rate for time step
25	Evaporation + Transpiration	- total for simulation
26	Evaporation + Transpiration	- total for time step
27	Evaporation + Transpiration	- rate for time step
28	Change in fluid stored in domain	- total for simulation
29	Change in fluid stored in domain	- total for time step
30	Change in fluid stored in domain	- rate for time step
31	Fluid volumetric balance	- total for simulation
32	Fluid volumetric balance	- total for time step
33	Fluid volumetric balance	- rate for time step
34	Solute flux in across specified pressure head boundaries	- total for simulation
35	Solute flux in across specified pressure head boundaries	- total for time step
36	Solute flux in across specified pressure head boundaries	- rate for time step
37	Solute flux out across specified pressure head boundaries	- total for simulation
38	Solute flux out across specified pressure head boundaries	- total for time step
39	Solute flux out across specified pressure head boundaries	- rate for time step
40	Solute flux in across specified flux boundaries	- total for simulation
41	Solute flux in across specified flux boundaries	- total for time step
42	Solute flux in across specified flux boundaries	- rate for time step
43	Solute flux out across specified flux boundaries	- total for simulation
44	Solute flux out across specified flux boundaries	- total for time step
45	Solute flux out across specified flux boundaries	- rate for time step
46	Diffusive/Dispersive flux in across specified flux boundaries	- total for simulation
47	Diffusive/Dispersive flux in across specified flux boundaries	- total for time step
48	Diffusive/Dispersive flux in across specified flux boundaries	- rate for time step
49	Diffusive/Dispersive flux out across specified flux boundaries	- total for simulation
50	Diffusive/Dispersive flux out across specified flux boundaries	- total for time step
51	Diffusive/Dispersive flux out across specified flux boundaries	- rate for time step

Table 7.--*Index of Mass Balance Components for Output to File 9*--Continued

Index Number	Component	
52	Total solute flux in	- total for simulation
53	Total solute flux in	- total for time step
54	Total solute flux in	- rate for time step
55	Total solute flux out	- total for simulation
56	Total solute flux out	- total for time step
57	Total solute flux out	- rate for time step
58	Solute flux out through evapotranspiration	- total for simulation
59	Solute flux out through evapotranspiration	- total for time step
60	Solute flux out through evapotranspiration	- rate for time step
61	First order decay of solute	- total for time step
62	First order decay of solute	- total for time step
63	First order decay of solute	- rate for time step
64	Adsorption or ion exchange of solute	- total for simulation
65	Adsorption or ion exchange of solute	- total for time step
66	Adsorption or ion exchange of solute	- rate for time step
67	Change in solute stored in domain	- total for simulation
68	Change in solute stored in domain	- total for time step
69	Change in solute stored in domain	- rate for time step
70	Solute mass balance	- total for simulation
71	Solute mass balance	- total for time step
72	Solute mass balance	- rate for time step.

```
      SUBROUTINE VSEXEC                                                 100
C                                                                       200
C******                                                                 300
CVSEXEC                                                                 400
C******                                                                 500
C -------------------------------------------------------------         600
C         ************ PROGRAM VS2D ********************                700
C                                                                       800
C   PROGRAM TO SOLVE FOR:                                               900
C      TWO DIMENSIONAL VERTICAL SECTION OR CYLINDRICAL THREE           1000
C          DIMENSIONAL FLUID FLOW AND SOLUTE TRANSPORT UNDER           1100
C          VARIABLY SATURATED CONDITIONS                               1200
C                                                                      1300
C       FLUID FLOW IS SOLVED FOR BY AN IMPLICIT FINITE DIFFERENCE      1400
C          FORMULATION OF THE COMBINED RICHARDS AND COOPER-JACOB       1500
C          EQUATIONS FOR FLUID CONTINUITY.                             1600
C                                                                      1700
C   -------      VERSION AS OF APRIL 1, 1990        ----------         1800
C .............................................................        1900
C                                                                      2000
C          DEFINITION OF FUNCTIONAL RELATIONSHIPS REQUIRED             2100
C  VSHKU = RELATIVE HYDRAULIC CONDUCTIVITY AS A FUNCTION OF            2200
C             PRESSURE HEAD                                            2300
C  VSTHU = VOLUMETRIC MOISTURE CONTENT AS A FUNCTION OF PRESSURE HEAD  2400
C  VSDTHU = FIRST DERIVATIVE OF MOISTURE CONTENT WITH RESPECT          2500
C             TO PRESSURE HEAD                                         2600
C  VSTHNV = PRESSURE HEAD AS A FUNCTION OF VOLUMETRIC MOISTURE         2700
C             CONTENT                                                  2800
C  VSRDF = ROOT ACTIVITY AS A FUNCTION OF TIME AND DEPTH               2900
C  VTRET = BULK DENSITY TIMES SLOPE OF ADSORPTION ISOTHERM.            3000
C                                                                      3100
C -------------------------------------------------------------       3200
C                                                                      3300
C      SPECIFICATIONS FOR ARRAYS AND SCALARS                           3400
C                                                                      3500
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                              3600
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2             3700
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                            3800
      COMMON/KCON/HX(1600),NTYP(1600)                                  3900
      COMMON/RPROP/HK(10,100),HT(10,20),ANIZ(10)                       4000
      COMMON/MPROP/THETA(1600),THLST(1600)                             4100
      COMMON/PRESS/P(1600),PXXX(1600),CS1,CS2                          4200
      COMMON/DISCH/Q(1600),QQ(1600),ETOUT,ETOUT1                       4300
      COMMON/HCON/HCND(1600),HKLL(1600),HKTT(1600)                     4400
      COMMON/EQUAT/A(1600),B(1600),C(1600),D(1600),E(1600),RHS(1600),  4500
     &XI(1600)                                                         4600
      COMMON/JTXX/JTEX(1600)                                           4700
      COMMON/DUMM/DUM(1600)                                            4800
      COMMON/SPFC/JSPX(3,25,8),NFC(8),JLAST(8),NFCS                    4900
      COMMON/PTET/DPTH(1600),RT(1600),RDC(6,25),ETCYC,                 5000
     &PEVAL(25),PTVAL(25),PET,PEV,HROOT,HA,SRES,RTDPTH,                5100
     &RTBOT,RTTOP,NPV                                                  5200
      COMMON/PND/POND                                                  5300
      COMMON/PLOTT/PLTIM(50),IJOBS(50),JPLT,NPLT,NOBS                  5400
      COMMON/WGT/WUS,WDS                                               5500
      COMMON/SCON/DHMX(201),DELT,HMAX,TMAX,EPS,NUMT,ITMAX,MINIT,ITEST  5600
      COMMON/SCN1/TMPX,TMLT,DLTMX,DLTMIN,TRED                          5700
      COMMON/TCON/STIM,DSMAX,KTIM,NIT,NIT1,KP                          5800
      COMMON/JCON/JSTOP,JFLAG                                          5900
      COMMON/TRXX/DX1(1600),DX2(1600),DZ1(1600),DZ2(1600),VX(1600),    6000
     &VZ(1600),CC(1600),COLD(1600),CS(1600),QT(1600),NCTYP(1600),      6100
     &RET(1600)                                                        6200
      COMMON/TRXY1/AO(1600),BO(1600),CO(1600),DO(1600),EO(1600)        6300
      LOGICAL TRANS,TRANS1,SORP,SSTATE                                 6400
```

```
      COMMON/TRXY/MB9(72),NMB9,EPS1,TRANS,TRANS1,SORP,SSTATE                6500
      LOGICAL RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                            6600
      LOGICAL F7P,F11P,F8P,F9P,F6P,PRNT                                     6700
      LOGICAL THPT,SPNT,PPNT,HPNT,VPNT                                      6800
      COMMON/LOG1/RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                        6900
      COMMON/LOG2/F7P,F11P,F8P,F9P,F6P,PRNT                                 7000
      COMMON/LOG4/THPT,SPNT,PPNT,HPNT,VPNT                                  7100
      CHARACTER*80 TITL                                                     7200
      CHARACTER*4 ZUNIT,TUNIT,CUNX                                          7300
      COMMON/SCHAR/TITL,ZUNIT,TUNIT,CUNX                                    7400
      SAVE IFET,IFET1,NITT,NITT1                                            7500
      DIMENSION KDUM(50,2)                                                  7600
C                                                                           7700
C-----------------------------------------------------------------------    7800
C                                                                           7900
C  ---- READ AND WRITE PROBLEM TITLE AND SPACE AND TIME CONSTANTS           8000
C                                                                           8100
      READ (05,4000) TITL                                                   8200
      READ (5,*) TMAX,STIM,ANG                                              8300
      READ (05,4010) ZUNIT,TUNIT,CUNX                                       8400
      READ (05,*) NXR,NLY                                                   8500
      READ (05,*) NRECH,NUMT                                                8600
      WRITE (06,4060)                                                       8700
      WRITE (06,4070) TITL,TMAX,TUNIT,STIM,NRECH,NUMT,NLY,NXR               8800
      WRITE(06,4080) ANG                                                    8900
      IF(ANG.GT.90..OR.ANG.LT.-90.)THEN                                     9000
      WRITE(06,4090)                                                        9100
      STOP                                                                  9200
      END IF                                                                9300
      READ (05,*) RAD,ITSTOP,TRANS                                          9400
      IF(TRANS) READ(05,*)CIS,CIT,SORP                                      9500
      READ (05,*) F11P,F7P,F8P,F9P,F6P                                      9600
      READ (05,*) THPT,SPNT,PPNT,HPNT,VPNT                                 9700
      WRITE (06,4100) F8P,ITSTOP,F7P,F11P,F9P,F6P                           9800
      WRITE (06,4110) THPT,SPNT,PPNT,HPNT,VPNT                              9900
      NLYY=NLY-1                                                           10000
      NXRR=NXR-1                                                           10100
      NNODES=NLY*NXR                                                       10200
C                                                                          10300
C   IF NUMBER OF NODES IS GREATER THAN ARRAY DIMENSIONS THEN                10400
C   TERMINATE SIMULATION                                                   10500
C                                                                          10600
      IF(NNODES.GT.1600.OR.NXR.GT.600.OR.NLY.GT.600) THEN                  10700
      WRITE (06,4020) NLY,NXR                                              10800
      STOP                                                                 10900
      END IF                                                               11000
C                                                                          11100
C   ESTABLISH HORIZONTAL OR RADIAL SPACING                                 11200
C                                                                          11300
      READ (05,*) IFAC,FACX                                                11400
      IF(IFAC.GT.0) GO TO 20                                               11500
C                                                                          11600
C   READ IN SPACING FOR EACH COLUMN                                        11700
C                                                                          11800
      READ (05,*) (DXR(K),K=1,NXR)                                         11900
      DO 10 K=1,NXR                                                        12000
   10 DXR(K)=DXR(K)*FACX                                                   12100
      GO TO 60                                                             12200
   20 IF(IFAC.EQ.2) GO TO 40                                               12300
      DO 30 K=1,NXR                                                        12400
   30 DXR(K)=FACX                                                          12500
      GO TO 60                                                             12600
C                                                                          12700
C  IF IFAC=2, HORIZONTAL NODE SPACING IS INCREMENTED BY A CONSTANT         12800
C    MULTIPLIER UNTIL A USER-SPECIFIED MAXIMUM IS REACHED, WHERE-          12900
C    UPON THE SPACING BECOMES CONSTANT                                     13000
C                                                                          13100
```

```
   40 READ (05,*) XMULT,XMAX                                          13200
      DXR(1)=FACX                                                     13300
      DXR(2)=FACX                                                     13400
      DO 50 K=3,NXRR                                                  13500
      DXR(K)=DXR(K-1)*XMULT                                           13600
      IF(DXR(K) .GT. XMAX)DXR(K)=XMAX                                 13700
   50 CONTINUE                                                        13800
      DXR(NXR)=DXR(NXRR)                                              13900
C                                                                     14000
C   ESTABLISH VERTICAL SPACING                                        14100
C                                                                     14200
   60 READ (05,*) JFAC,FACZ                                           14300
      IF(JFAC.GT.0) GO TO 80                                          14400
C                                                                     14500
C   READ IN VERTICAL SPACINGS INDIVIDUALLY                            14600
C                                                                     14700
      READ (05,*) (DELZ(K),K=1,NLY)                                   14800
      DO 70 K=1,NLY                                                   14900
   70 DELZ(K)=DELZ(K)*FACZ                                            15000
      GO TO 120                                                       15100
   80 IF(JFAC.EQ.2) GO TO 100                                         15200
      DO 90 K=1,NLY                                                   15300
   90 DELZ(K)=FACZ                                                    15400
      GO TO 120                                                       15500
C                                                                     15600
C   ESTABLISH VERTICAL SPACING BY PROGRESSION, AS ABOVE FOR HORIZ.    15700
C                                                                     15800
  100 READ (05,*) ZMULT,ZMAX                                          15900
      DELZ(1)=FACZ                                                    16000
      DELZ(2)=FACZ                                                    16100
      DO 110 K=3,NLYY                                                 16200
      DELZ(K)=DELZ(K-1)*ZMULT                                         16300
      IF(DELZ(K) .GT. ZMAX)DELZ(K)=ZMAX                               16400
  110 CONTINUE                                                        16500
      DELZ(NLY)=DELZ(NLYY)                                            16600
  120 CONTINUE                                                        16700
C                                                                     16800
C     DETERMINE HORIZONTAL AND VERTICAL COORDINATES                   16900
C                                                                     17000
      RX(1)=-0.5 *DXR(1)                                              17100
      DO 130 N=2,NXR                                                  17200
      RX(N)=RX(N-1)+0.5 *(DXR(N-1)+DXR(N))                            17300
  130 CONTINUE                                                        17400
      DZZ(1)=-0.5 *DELZ(1)                                            17500
      DO 140 J=2,NLY                                                  17600
  140 DZZ(J)=DZZ(J-1)+0.5 *(DELZ(J-1)+DELZ(J))                        17700
      WRITE (06,4120) ZUNIT,(DELZ(K),K=1,NLY)                         17800
      WRITE (06,4130) ZUNIT,(DXR(K),K=1,NXR)                          17900
      PI=3.141592654                                                  18000
      PI2=PI+PI                                                       18100
      ANG=ANG/360.                                                    18200
      IF(ANG.EQ.0) THEN                                               18300
      CS1=1                                                           18400
      CS2=0.                                                          18500
      ELSE                                                            18600
      IF(ANG.EQ.0.25.OR.ANG.EQ.-0.25) THEN                            18700
      CS1=0.                                                          18800
      ELSE                                                            18900
      CS1=DCOS(ANG*PI2)                                               19000
      END IF                                                          19100
      CS2=-DSIN(ANG*PI2)                                              19200
      END IF                                                          19300
C                                                                     19400
C    READ DATA FOR MONITORING TIMES AND POINTS                        19500
C                                                                     19600
      NPLT=0                                                          19700
      IF(F8P) THEN                                                    19800
```

```
      READ (05,*) NPLT                                                  19900
      IF(NPLT.GT.50)NPLT=50                                             20000
      IF(NPLT.EQ.0)NPLT=1                                               20100
      READ (05,*) (PLTIM(K),K=1,NPLT)                                   20200
      WRITE (06,4140) (PLTIM(K),K=1,NPLT)                               20300
      END IF                                                            20400
      IF(F11P) THEN                                                     20500
      READ (05,*) NOBS                                                  20600
      READ (05,*) ((KDUM(K,J),J=1,2),K=1,NOBS)                          20700
      WRITE (06,4150) ((KDUM(K,J),J=1,2),K=1,NOBS)                      20800
      DO 150 K=1,NOBS                                                   20900
      N=NLY*(KDUM(K,2)-1)+KDUM(K,1)                                     21000
  150 IJOBS(K)=N                                                        21100
      END IF                                                            21200
      IF (F9P) THEN                                                     21300
      READ(05,*)NMB9                                                    21400
      READ(05,*) (MB9(K),K=1,NMB9)                                      21500
      WRITE(06,4160) (MB9(K),K=1,NMB9)                                  21600
      END IF                                                            21700
      PLTIM(NPLT+1)=TMAX+TMAX                                           21800
      IF(RAD) THEN                                                      21900
      WRITE(06,4050)                                                    22000
      ELSE                                                              22100
      WRITE (06,4040)                                                   22200
      END IF                                                            22300
      IF(TRANS) THEN                                                    22400
      WRITE(06,4240)                                                    22500
      IF(CIS) THEN                                                      22600
      WRITE(6,4200)                                                     22700
      ELSE                                                              22800
      WRITE(6,4210)                                                     22900
      END IF                                                            23000
      IF(CIT) THEN                                                      23100
      WRITE(6,4220)                                                     23200
      ELSE                                                              23300
      WRITE(6,4230)                                                     23400
      END IF                                                            23500
      IF(SORP) WRITE(06,4250)                                           23600
      END IF                                                            23700
      IF(F11P) WRITE (11,4030) TITL,TUNIT,ZUNIT,ZUNIT,ZUNIT,ZUNIT       23800
C                                                                       23900
C     INITIALIZE CONSTANTS                                              24000
C                                                                       24100
      ITEST=0                                                           24200
      KTIM=0                                                            24300
      NITT=0                                                            24400
      NITT1=0                                                           24500
      JFLAG=1                                                           24600
      KP=0                                                              24700
      WRITE (06,4170)                                                   24800
C                                                                       24900
C                                                                       25000
C   READ AND WRITE INITIAL VALUES OF PRESSURE HEAD, TOTAL HEAD,         25100
C    THETA, AND SATURATION                                              25200
C -------------------------------------------------------------        25300
C                                                                       25400
      CALL VSREAD                                                       25500
      CALL VSSIP                                                        25600
      IFET=0                                                            25700
      IFET2=0                                                           25800
      CALL VSOUTP                                                       25900
C                                                                       26000
C -------------------------------------------------------------        26100
C       START OF TIME LOOP                                              26200
C -------------------------------------------------------------        26300
C                                                                       26400
  160 IF(JFLAG.EQ.1)IFET1=1                                             26500
```

```
      CALL VSTMER                                                       26600
      TRANS1=.FALSE.                                                    26700
      IF(.NOT.SSTATE) THEN                                              26800
C                                                                       26900
C   SET UP AND SOLVE MATRIX EQUATIONS FOR FLOW                          27000
C                                                                       27100
  170 CALL VSMGEN                                                       27200
C                                                                       27300
C   CHECK FOR PONDING DURING THIS TIME STEP                             27400
C                                                                       27500
      CALL VSPOND(IFET,IFET1,IFET2)                                     27600
C                                                                       27700
C   IF PONDING HAS OCCURRED, EQUATIONS NEED TO BE SOLVED AGAIN          27800
C                                                                       27900
      IF(IFET.NE.0) THEN                                                28000
      IF(NIT.LT.ITMAX) THEN                                             28100
      GO TO 170                                                         28200
      ELSE                                                              28300
      WRITE(6,4260)                                                     28400
      END IF                                                            28500
      END IF                                                            28600
C                                                                       28700
C   REEVALUATE NONLINEAR COEFFICIENTS                                   28800
C                                                                       28900
      CALL VSCOEF                                                       29000
      NITT=NITT+NIT                                                     29100
      END IF                                                            29200
      IF((TRANS.OR.VPNT).AND..NOT.SSTATE) CALL VTVELO                   29300
      IF(TRANS) THEN                                                    29400
      IF(.NOT.SSTATE) THEN                                              29500
C                                                                       29600
C   DETERMINE VELOCITIES AND DISPERSION TENSOR                          29700
C                                                                       29800
      CALL VTDCOEF                                                      29900
      END IF                                                            30000
      TRANS1=.TRUE.                                                     30100
C                                                                       30200
C   SET UP AND SOLVE MATRIX EQUATION FOR TRANSPORT                      30300
C                                                                       30400
      CALL VTSETUP                                                      30500
      NITT1=NITT1+NIT1                                                  30600
      END IF                                                            30700
C                                                                       30800
C   PRINT RESULTS AND COMPUTE MASS BALANCE COMPONENTS                   30900
C                                                                       31000
      CALL VSOUTP                                                       31100
      CALL VSFLUX                                                       31200
      IF(JSTOP.NE.1) GO TO 160                                          31300
C                                                                       31400
C-----------------------------------------------------------------      31500
C     END OF TIME LOOP                                                  31600
C-----------------------------------------------------------------      31700
C                                                                       31800
      WRITE (06,4180)                                                   31900
      WRITE (6,4190) NITT,NITT1                                         32000
      RETURN                                                            32100
 4000 FORMAT(A80)                                                       32200
 4010 FORMAT(4A4)                                                       32300
 4020 FORMAT(5X,20(1H*),1X,31HDIMENSIONS TOO LARGE FOR ARRAYS,          32400
     &1X,20(1H*)/5X,6HNLY = ,I5,2X,6H,NXR = ,I5)                        32500
 4030 FORMAT(A80/21HMONITORING POINT FILE/2X,6HTIME,  ,A4,2X,           32600
     &  6H XR,  ,A4,2X,6H Z,    ,A4,2X,6H H,   ,A4,2X,6H  P,  ,A4,     32700
     & 2X,6H THETA,4X,8H     SAT)                                       32800
 4040 FORMAT(5X,32HCOORDINATE SYSTEM IS RECTANGULAR)                    32900
 4050 FORMAT(5X,27HCOORDINATE SYSTEM IS RADIAL)                         33000
 4060 FORMAT(35X,60(1H+)/35X,1H+,26X,6H VS2DT,26X,1H+/35X,              33100
     &1H+,4X,36HSIMULATION OF 2-DIMENSIONAL VARIABLY,18X,1H+/           33200
```

```
     &35X,1H+,4X,35HSATURATED FLOW AND SOLUTE TRANSPORT,19X,1H+          33300
     &/35X,1H+,4X,41HTHROUGH POROUS MEDIA.  VERSION DATED     ,13X,1H+  33400
     &/35X,1H+,26X,'4-1-90',26X,1H+                                     33500
     & /35X,60(1H+)//)                                                  33600
 4070 FORMAT(//,1X,100(1H*)/5X,A80/1X,100(1H*)//10X,                    33700
     &24HSPACE AND TIME CONSTANTS/10X,23(1H-)/                          33800
     & 5X,26HMAXIMUM SIMULATION TIME = ,F10.4,1X,A4/                    33900
     &5X,'STARTING TIME = ',F10.4,/                                     34000
     &5X,28HNUMBER OF RECHARGE PERIODS =,I10/                           34100
     &4X,32H MAXIMUM NUMBER OF TIME STEPS = ,I10/                       34200
     &5X,17HNUMBER OF ROWS = ,I5/5X,20HNUMBER OF COLUMNS = ,I5)         34300
 4080 FORMAT(5X,'AXES TILTED BY ANGLE = ',F8.2)                         34400
 4090 FORMAT(1X,'ANGLE OF AXES TILTING MUST BE BETWEEN -90 AND 90 ',    34500
     &'DEGREES',/,1X,'SIMULATION TERMINATED')                           34600
 4100 FORMAT(10X,16HSOLUTION OPTIONS/10X,16(1H-)/                       34700
     &5X,'WRITE ALL PRESSURE HEADS TO FILE 8',                          34800
     &23H AT OBSERVATION TIMES? ,L1,/                                   34900
     &5X,28HSTOP SOLUTION IF MAXIMUM NO.,                               35000
     &42H OF ITERATIONS EXCEEDED IN ANY TIME STEP?,L1/5X,               35100
     &'WRITE MAXIMUM CHANGE IN HEAD FOR EACH ITERATION TO FILE 7? ',    35200
     &L1/5X,'WRITE RESULTS AT SELECTED OBSERVATION POINTS TO ',         35300
     &9HFILE 11? , L1/,5X,36HWRITE MASS BALANCE RATES TO FILE 9? L1/    35400
     &5X,36HWRITE MASS BALANCE RATES TO FILE 6? ,L1)                    35500
 4110 FORMAT(1H ,4X,35HWRITE MOISTURE CONTENTS TO FILE 6? ,L1/          35600
     & 5X,29HWRITE SATURATIONS TO FILE 6? ,L1/                          35700
     & 5X,32HWRITE PRESSURE HEADS TO FILE 6? ,L1/                       35800
     & 5X,29HWRITE TOTAL HEADS TO FILE 6? ,L1/                          35900
     &5X,'WRITE VELOCITIES TO FILE 6? ',L1)                             36000
 4120 FORMAT(50X,39HGRID SPACING IN VERTICAL DIRECTION, IN ,A4/         36100
     & (10(F10.3)))                                                     36200
 4130 FORMAT(50X,47HGRID SPACING IN HORIZONTAL OR RADIAL DIRECTION,     36300
     &,3H IN,1X,A4/(10F10.3))                                           36400
 4140 FORMAT(5X,43HTIMES AT WHICH H WILL BE WRITTEN TO FILE 08          36500
     &/(5X,10F10.4))                                                    36600
 4150 FORMAT(5X,37HROW AND COLUMN OF OBSERVATION POINTS:/               36700
     & 3X,10(2X,2I4))                                                   36800
 4160 FORMAT(5X,'MASS BALANCE COMPONENTS WRITTEN TO FILE 9',            36900
     &/,5X,24I4)                                                        37000
 4170 FORMAT(5X,36HMATRIX EQUATIONS TO BE SOLVED BY SIP)                37100
 4180 FORMAT(5X,100(1H*)/5X,17HEND OF SIMULATION/                       37200
     &   5X,100(1H*))                                                   37300
 4190 FORMAT(5X,'TOTAL NUMBER OF ITERATIONS FOR FLOW EQUATION = ',I6    37400
     &/5X,'TOTAL NUMBER OF ITERATIONS FOR TRANSPORT EQUATION = ',I6)    37500
 4200 FORMAT(5X,'CENTRAL DIFFERENCING IN SPACE USED FOR TRANSPORT',     37600
     &' EQUATION')                                                      37700
 4210 FORMAT(4X,' BACKWARD DIFFERENCING IN SPACE USED FOR TRANSPORT',   37800
     &' EQUATION')                                                      37900
 4220 FORMAT(4X,' CENTRAL DIFFERENCING IN TIME USED FOR TRANSPORT',     38000
     &' EQUATION')                                                      38100
 4230 FORMAT(4X,' BACKWARD DIFFERENCING IN TIME USED FOR TRANSPORT',    38200
     &' EQUATION')                                                      38300
 4240 FORMAT(4X,' TRANSPORT TO BE SIMULATED')                           38400
 4250 FORMAT(4X,' NONLINEAR SORPTION TO BE SIMULATED')                  38500
 4260 FORMAT(5X,'-- WARNING --  INFILTRATION/PONDING BOUNDARY WAS NOT'  38600
     &' SOLVED ACCURATELY FOR THIS TIME STEP')                          38700
      END                                                               38800
      BLOCK DATA  DAT1                                                  38900
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               39000
      COMMON/PRESS/P(1600),PXXX(1600),CS1,CS2                           39100
      COMMON/KCON/HX(1600),NTYP(1600)                                   39200
      COMMON/MPROP/THETA(1600),THLST(1600)                              39300
      COMMON/HCON/HCND(1600),HKLL(1600),HKTT(1600)                      39400
      COMMON/PTET/DPTH(1600),RT(1600),RDC(6,25),ETCYC,                  39500
     &PEVAL(25),PTVAL(25),PET,PEV,HROOT,HA,SRES,RTDPTH,                 39600
     &RTBOT,RTTOP,NPV                                                   39700
      COMMON/DISCH/Q(1600),QQ(1600),ETOUT,ETOUT1                        39800
      COMMON/EQUAT/A(1600),B(1600),C(1600),D(1600),E(1600),RHS(1600),   39900
```

```
     &XI(1600)                                                          40000
      COMMON/TRXX/DX1(1600),DX2(1600),DZ1(1600),DZ2(1600),VX(1600),     40100
     &VZ(1600),CC(1600),COLD(1600),CS(1600),QT(1600),NCTYP(1600),       40200
     &RET(1600)                                                         40300
      DATA P/1600*0.0/,PXXX/1600*0.0/,HX/1600*0.0/,THETA/1600*0.0/,     40400
     &THLST/1600*0.0/                                                   40500
      DATA HCND/1600*0.0/,HKLL/1600*0.0/,HKTT/1600*0.0/,DPTH/1600*0.0/, 40600
     &RT/1600*0.0/,PTVAL/25*0.0/,PEVAL/25*0.0/                          40700
      DATA Q/1600*0.0/,QQ/1600*0.0/                                     40800
      DATA A/1600*0.0/,B/1600*0.0/,C/1600*0.0/,D/1600*0.0/,             40900
     &E/1600*0.0/,RHS/1600*0.0/,XI/1600*0.0/                            41000
      DATA DX1/1600*0.0/,DX2/1600*0.0/,DZ1/1600*0.0/,DZ2/1600*0.0/,     41100
     &VX/1600*0.0/,VZ/1600*0.0/,CC/1600*0.0/,COLD/1600*0.0/,            41200
     &CS/1600*0.0/,QT/1600*0.0/,RET/1600*0.0/                           41300
      END                                                               41400
      SUBROUTINE VSREAD                                                 41500
C******                                                                 41600
CVSREAD                                                                 41700
C******                                                                 41800
C                                                                       41900
C   PURPOSE: TO READ INITIAL HEAD AND SATURATION DATA                   42000
C                                                                       42100
C ---------------------------------------------------------------       42200
C                                                                       42300
C   SPECIFICATIONS FOR ARRAYS AND SCALARS                               42400
C                                                                       42500
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               42600
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2              42700
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                             42800
      COMMON/KCON/HX(1600),NTYP(1600)                                   42900
      COMMON/RPROP/HK(10,100),HT(10,20),ANIZ(10)                        43000
      COMMON/MPROP/THETA(1600),THLST(1600)                              43100
      COMMON/PRESS/P(1600),PXXX(1600),CS1,CS2                           43200
      COMMON/DISCH/Q(1600),QQ(1600),ETOUT,ETOUT1                        43300
      COMMON/HCON/HCND(1600),HKLL(1600),HKTT(1600)                      43400
      COMMON/JTXX/JTEX(1600)                                            43500
      COMMON/DUMM/DUM(1600)                                             43600
      COMMON/PTET/DPTH(1600),RT(1600),RDC(6,25),ETCYC,                  43700
     &PEVAL(25),PTVAL(25),PET,PEV,HROOT,HA,SRES,RTDPTH,                 43800
     &RTBOT,RTTOP,NPV                                                   43900
      COMMON/WGT/WUS,WDS                                                44000
      COMMON/SCON/DHMX(201),DELT,HMAX,TMAX,EPS,NUMT,ITMAX,MINIT,ITEST   44100
      COMMON/SCN1/TMPX,TMLT,DLTMX,DLTMIN,TRED                           44200
      COMMON/TCON/STIM,DSMAX,KTIM,NIT,NIT1,KP                           44300
      COMMON/TRXX/DX1(1600),DX2(1600),DZ1(1600),DZ2(1600),VX(1600),     44400
     &VZ(1600),CC(1600),COLD(1600),CS(1600),QT(1600),NCTYP(1600),       44500
     &RET(1600)                                                         44600
      LOGICAL TRANS,TRANS1,SORP,SSTATE                                  44700
      COMMON/TRXY/MB9(72),NMB9,EPS1,TRANS,TRANS1,SORP,SSTATE            44800
      LOGICAL PHRD                                                      44900
      LOGICAL RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                        45000
      COMMON/LOG1/RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                    45100
      CHARACTER*80 TITL                                                 45200
      CHARACTER*36 IFMT                                                 45300
      CHARACTER*4 ZUNIT,TUNIT,CUNX                                      45400
      COMMON/SCHAR/TITL,ZUNIT,TUNIT,CUNX                                45500
      DIMENSION IDUM(0600)                                              45600
C-----------------------------------------------------------------------45700
C                                                                       45800
C   READ AND WRITE INITIAL DATA FOR SIMULATION                          45900
C                                                                       46000
      IF (TRANS) THEN                                                   46100
      READ (5,*) EPS,HMAX,WUS,EPS1                                      46200
      ELSE                                                              46300
      READ(5,*) EPS,HMAX,WUS                                            46400
      EPS1=0.0                                                          46500
      END IF                                                            46600
```

```
      READ (5,*) MINIT,ITMAX                                            46700
      READ (05,*) PHRD                                                  46800
      IF(TRANS) THEN                                                    46900
      READ (05,*) NTEX,NPROP,NPROP1                                     47000
      ELSE                                                              47100
      READ (05,*) NTEX,NPROP                                            47200
      NPROP1=0                                                          47300
      END IF                                                            47400
C                                                                       47500
C   CHECK THAT SUM OF WEIGHTING FACTORS IS EQUAL TO ONE                 47600
C                                                                       47700
      WRITE (6,4000) EPS,ZUNIT,EPS1,HMAX                                47800
      IF(WUS.EQ.1) THEN                                                 47900
      WDS=0.                                                            48000
      WRITE(06,4020)                                                    48100
      ELSE                                                              48200
      IF(WUS.EQ.0.5) THEN                                               48300
      WDS=0.5                                                           48400
      WRITE(06,4070)                                                    48500
      ELSE                                                              48600
      WUS=0.0                                                           48700
      WRITE(06,4010)                                                    48800
      END IF                                                            48900
      END IF                                                            49000
      WRITE (6,4080) NTEX,NPROP,NPROP1,MINIT,ITMAX                      49100
      IF(ITMAX.GT.200) GO TO 210                                        49200
      WRITE (06,4100)                                                   49300
      IF (TRANS) WRITE(06,4110)                                         49400
C                                                                       49500
C   READ AND WRITE MATERIAL PROPERTIES FOR EACH TEXTURAL CLASS          49600
C                                                                       49700
      DO 20J22=1,10                                                     49800
      DO 10J23=1,100                                                    49900
   10 HK(J22,J23)=0.                                                    50000
      DO 20J23=1,20                                                     50100
   20 HT(J22,J23)=0.                                                    50200
      DO 30J22=1,NTEX                                                   50300
      READ (5,*) J                                                      50400
      READ (5,*) ANIZ(J),(HK(J,I),I=1,NPROP)                           50500
      WRITE (6,4120) J,ANIZ(J),(HK(J,I),I=1,NPROP)                      50600
      IF(TRANS) THEN                                                    50700
      READ(5,*) (HT(J,I),I=1,NPROP1)                                    50800
      WRITE(6,4130) (HT(J,I),I=1,NPROP1)                                50900
      END IF                                                            51000
   30 CONTINUE                                                          51100
      WRITE (06,4140)                                                   51200
C                                                                       51300
C    READ TEXTURAL CLASS INDEX MAP                                      51400
C                                                                       51500
      READ (05,*) IROW                                                  51600
      IF(IROW.EQ.0) THEN                                                51700
      WRITE(06,4090)                                                    51800
      DO 50 J=1,NLY                                                     51900
      READ (05,*) (IDUM(N),N=1,NXR)                                     52000
      WRITE (06,4150) J,(IDUM(N),N=1,NXR)                               52100
      DO 40 N=1,NXR                                                     52200
      IN=NLY*(N-1)+J                                                    52300
      J22=IDUM(N)                                                       52400
      HX(IN)=HK(J22,1)                                                  52500
   40 JTEX(IN)=J22                                                      52600
   50 CONTINUE                                                          52700
      ELSE                                                              52800
C                                                                       52900
C    READ TEXTURE CLASSES BY BLOCK--EITHER CONTINUOUS LAYERS OR         53000
C    LAYERS BOUNDED BY VERTICAL DISCONTINUITIES.                        53100
C                                                                       53200
      WRITE (06,4040)                                                   53300
```

```
      JTP=1                                                          53400
   60 READ (05,*) IL,IR,JBT,JRD                                      53500
      DO 70 N=IL,IR                                                  53600
      IDUM(N)=JRD                                                    53700
   70 CONTINUE                                                       53800
      IF(IR.LT.NXR) GO TO 60                                         53900
      DO 80 J=JTP,JBT                                                54000
   80 WRITE (06,4150) J,(IDUM(N),N=1,NXR)                            54100
      DO 90 J=JTP,JBT                                                54200
      DO 90 N=1,NXR                                                  54300
      IN=NLY*(N-1)+J                                                 54400
      J22=IDUM(N)                                                    54500
      HX(IN)=HK(J22,1)                                               54600
      JTEX(IN)=J22                                                   54700
   90 CONTINUE                                                       54800
      IF(JBT.EQ.NLY) GO TO 100                                       54900
      JTP=JBT+1                                                      55000
      GO TO 60                                                       55100
      END IF                                                         55200
  100 CONTINUE                                                       55300
C                                                                    55400
C   BORDERS OF DOMAIN ARE ALL SET TO NO FLOW BOUNDARIES              55500
C                                                                    55600
      DO 110 I=1,NLY                                                 55700
      I1=NNODES-I+1                                                  55800
      HX(I)=0                                                        55900
  110 HX(I1)=0                                                       56000
      DO 120 I=2,NXR                                                 56100
      I1=(I-1)*NLY                                                   56200
      HX(I1)=0                                                       56300
  120 HX(I1+1)=0                                                     56400
C                                                                    56500
C   READ INITIAL HEADS OR MOISTURE CONTENTS                          56600
C                                                                    56700
      READ (05,*) IREAD,FACTOR                                       56800
      IF(IREAD.EQ.2) THEN                                            56900
      READ (05,*) DWTX,HMIN                                          57000
      WRITE (06,4190) DWTX,ZUNIT,HMIN,ZUNIT,DWTX,ZUNIT               57100
C                                                                    57200
C   CALCULATE EQUILIBRIUM INITIAL HEAD PROFILE                       57300
C                                                                    57400
      DO 130 J=2,NLYY                                                57500
      DO 130 N=2,NXRR                                                57600
      IN=NLY*(N-1)+J                                                 57700
      IF(HX(IN).EQ.0.) GO TO 130                                     57800
      IF(CS1.EQ.1.) THEN                                             57900
      Z1=DZZ(J)                                                      58000
      ELSE                                                           58100
      Z1=DZZ(J)*CS1+(RX(N))*CS2                                      58200
      END IF                                                         58300
      P1=Z1-DWTX                                                     58400
      IF(P1.LT.HMIN)P1=HMIN                                          58500
      P(IN)=P1-Z1                                                    58600
      PXXX(IN)=P(IN)                                                 58700
  130 CONTINUE                                                       58800
      ELSE                                                           58900
      IF(IREAD.NE.1) THEN                                            59000
      WRITE (6,4170) FACTOR                                          59100
      ELSE                                                           59200
      READ (05,*) IU,IFMT                                            59300
      WRITE (06,4180) IU,FACTOR                                      59400
      END IF                                                         59500
      DO 160 J=1,NLY                                                 59600
      IF(IREAD.NE.0) THEN                                            59700
C                                                                    59800
C   READ INITIAL CONDITIONS FROM FILE IU                             59900
C                                                                    60000
```

```
      READ (IU,FMT=IFMT) (DUM(N),N=1,NXR)                                60100
      ELSE                                                               60200
      DO 140 N=1,NXR                                                     60300
  140 DUM(N)=FACTOR                                                      60400
      END IF                                                             60500
      DO 150 N=1,NXR                                                     60600
      IN=NLY*(N-1)+J                                                     60700
      IF(IREAD.EQ.1)DUM(N)=DUM(N)*FACTOR                                 60800
      IF(CS1.EQ.1.) THEN                                                 60900
      Z1=DZZ(J)                                                          61000
      ELSE                                                               61100
      Z1=DZZ(J)*CS1+(RX(N))*CS2                                          61200
      END IF                                                             61300
      IF(.NOT.PHRD) THEN                                                 61400
      IF(DUM(N).LE.0.) THEN                                              61500
      WRITE(6,4230) J,N                                                  61600
      STOP                                                               61700
      END IF                                                             61800
C                                                                        61900
C   CONVERT INITIAL MOISTURE CONTENTS TO HEADS                           62000
C                                                                        62100
      P(IN)=VSTHNV(DUM(N),JTEX(IN),HK)-Z1                                62200
      THETA(IN)=DUM(N)                                                   62300
      ELSE                                                               62400
      P(IN)=DUM(N)-Z1                                                    62500
      END IF                                                             62600
      PXXX(IN)=P(IN)                                                     62700
  150 CONTINUE                                                           62800
  160 CONTINUE                                                           62900
C                                                                        63000
C   COMPUTE INITIAL NONLINEAR COEFFICIENT VALUES                         63100
C                                                                        63200
      END IF                                                             63300
      CALL VSCOEF                                                        63400
C                                                                        63500
C   IF ET IS TO BE SIMULATED, ALL VARIABLES MUST BE ENTERED HERE.        63600
C                                                                        63700
      READ(05,*) BCIT,ETSIM                                              63800
      IF(BCIT .OR. ETSIM) THEN                                           63900
C                                                                        64000
C    COMPUTE DEPTHS FOR ET CALCULATIONS                                  64100
C                                                                        64200
      DPTH(1)=-.5 *DELZ(1)                                               64300
      DO 170 J=2,NLYY                                                    64400
      DO 170 N=2,NXRR                                                    64500
      IN=NLY*(N-1)+J                                                     64600
      JM1=IN-1                                                           64700
      IF(HX(IN).NE.0.) THEN                                              64800
      IF(HX(JM1).EQ.0.) THEN                                             64900
      DPTH(IN)=0.0                                                       65000
      ELSE                                                               65100
      DPTH(IN)=DPTH(JM1)+DELZ(J-1)                                       65200
      END IF                                                             65300
      END IF                                                             65400
  170 CONTINUE                                                           65500
      WRITE (6,4200)                                                     65600
      CALL VSOUT(2,DPTH)                                                 65700
C                                                                        65800
C   READ EVAPORATION VARIABLES                                           65900
C                                                                        66000
      READ(05,*)NPV,ETCYC                                                66100
      WRITE(6,4030) NPV,ETCYC,TUNIT                                      66200
      IF(BCIT) THEN                                                      66300
      READ (05,*)(PEVAL(I),I=1,NPV)                                      66400
      READ(05,*)  (RDC(1,I),I=1,NPV)                                     66500
      READ(05,*)  (RDC(2,I),I=1,NPV)                                     66600
      WRITE (06,4050)ZUNIT,TUNIT,ZUNIT,ZUNIT,(I,PEVAL(I),RDC(1,I),RDC(2,  66700
```

```
     *I),I=1,NPV)                                                        66800
      END IF                                                             66900
      IF (ETSIM )THEN                                                    67000
C                                                                        67100
C   READ TRANSPIRATION VARIABLES                                         67200
C                                                                        67300
      READ(05,*)(PTVAL(I),I=1,NPV)                                       67400
      READ(05,*) (RDC(3,I),I=1,NPV)                                      67500
      READ(05,*) (RDC(4,I),I=1,NPV)                                      67600
      READ(05,*) (RDC(5,I),I=1,NPV)                                      67700
      READ(05,*) (RDC(6,I),I=1,NPV)                                      67800
      WRITE(06,4060)ZUNIT,TUNIT,ZUNIT,ZUNIT,ZUNIT,ZUNIT,(I,PTVAL(I),    67900
     *(RDC(J,I),J=3,6),I=1,NPV)                                          68000
      END IF                                                             68100
      END IF                                                             68200
      DO 180 IN=1,NNODES                                                 68300
      NTYP(IN)=0                                                         68400
      NCTYP(IN)=0                                                        68500
      IF(HX(IN).GT.0) THLST(IN)=THETA(IN)                                68600
  180 CONTINUE                                                           68700
C                                                                        68800
C    READ INITIAL CONCENTRATIONS IF TRANSPORT EQUATION IS TO             68900
C    BE SOLVED                                                           69000
C                                                                        69100
      IF (TRANS) THEN                                                    69200
      READ(05,*) IREAD,FACTOR                                            69300
      IF(IREAD.EQ.0) THEN                                                69400
      WRITE(6,4210) FACTOR                                               69500
      DO 190 N=1,NNODES                                                  69600
      CC(N)=FACTOR                                                       69700
      COLD(N)=FACTOR                                                     69800
  190 CONTINUE                                                           69900
      ELSE                                                               70000
      READ(05,*)IU,IFMT                                                  70100
      WRITE(06,4220) IU,FACTOR                                           70200
      DO 200 J=1,NLY                                                     70300
      READ(IU,FMT=IFMT) (DUM(N),N=1,NXR)                                 70400
      DO 200 N=1,NXR                                                     70500
      IN=NLY*(N-1)+J                                                     70600
      CC(IN)=DUM(N)*FACTOR                                               70700
      COLD(IN)=CC(IN)                                                    70800
  200 CONTINUE                                                           70900
      END IF                                                             71000
C                                                                        71100
C   COMPUTE INTERCELL CONDUCTANCES                                       71200
C                                                                        71300
      END IF                                                             71400
      CALL VSHCMP                                                        71500
      RETURN                                                             71600
  210 WRITE (06,4160) ITMAX                                              71700
      STOP                                                               71800
 4000 FORMAT(10X,27HINITIAL MOISTURE PARAMETERS/10X,27(1H_)//            71900
     &5X,'CONVERGENCE CRITERIA FOR SIP FOR FLOW =',1PE12.3,1X,A4/        72000
     &5X,'CONVERGENCE CRITERIA FOR SIP FOR TRANSPORT =',1PE12.3,1X,/     72100
     &5X,23HDAMPING FACTOR, HMAX = ,1PE12.3)                             72200
 4010 FORMAT(5X,46HGEOMETRIC MEAN USED FOR INTERCELL CONDUCTIVITY)       72300
 4020 FORMAT(5X,45HUPSTREAM WEIGHTING USED FOR INTERCELL CONDUCT         72400
     &,5HIVITY)                                                          72500
 4030 FORMAT(//15X,'NUMBER OF EVAPORATION AND/OR EVAPOTRASPIRATION PER'  72600
     &,'IODS = ',I4,/,15X,'LENGTH OF EACH PERIOD = ',F10.4,2X,A4)        72700
 4040 FORMAT(5X,'TEXTURAL CLASSES READ IN BY BLOCK')                     72800
 4050 FORMAT(//5X,'EVAPORATION   POTENTIAL      SURFACE   ATMOSHERIC',   72900
     &/'       PERIOD        RATE       RESISTANCE    PRESSURE',        73000
     &/19X,A4,'/',A4,3X,A4,'**(-1)',5X,A4,/,1X,90('-'),                  73100
     &25(/,5X,I6,4X,3E14.5))                                             73200
 4060 FORMAT(//,3X,'TRANSPIRATION   POTENTIAL         ROOT       ACTIVIT  73300
     &Y      ACTIVITY        ROOT',                                      73400
```

```
     &/'       PERIOD        RATE            DEPTH     AT BOTTOM       A   73500
     &T TOP       PRESSURE',/,19X,A4,'/',A4,9X,A4,5X,A4,'**(-2)',4X,A4,    73600
     &'**(-2)',8X,A4,/,1X,90('-'),25(/,5X,I6,4X,5E14.5))                   73700
 4070 FORMAT(5X,47HARITHEMTIC MEAN USED FOR INTERCELL CONDUCTIVITY)        73800
 4080 FORMAT(5X,34HNUMBER OF SOIL TEXTURAL CLASSES = ,I10/                 73900
     &5X,43HNUMBER OF SOIL PARAMETERS FOR EACH CLASS = ,I10/               74000
     &5X,'NUMBER OF TRANSPORT PARAMETERS FOR EACH CLASS = ',I10/           74100
     &5X,47HMINIMUM PERMITTED NO. OF ITERATIONS/TIME STEP =,I10/           74200
     &5X,47HMAXIMUM PERMITTED NO. OF ITERATIONS/TIME STEP =,I10)           74300
 4090 FORMAT(5X,41HTEXTURAL CLASS TO BE READ IN FOR EACH ROW)              74400
 4100 FORMAT(41X,35HCONSTANTS FOR SOIL TEXTURAL CLASSES//                  74500
     &10X,10HANISOTROPY,7X,4HKSAT,5X,8HSPECIFIC,4X,8HPOROSITY,/,           74600
     &36X,7HSTORAGE)                                                       74700
 4110 FORMAT(12X,'ALPHAL',8X,'ALPHAT',6X,'DM',9X,'LAMBDA',                 74800
     &4X,'B DENSITY')                                                      74900
 4120 FORMAT(1X,7HCLASS #,I2,/9X,3(1PD12.3),14(7(1PD12.3),/))              75000
 4130 FORMAT(9X,10(1PD12.3))                                               75100
 4140 FORMAT(6X,24HTEXTURAL CLASS INDEX MAP//   )                          75200
 4150 FORMAT(1H ,5X,I4,2X,100I1)                                           75300
 4160 FORMAT(5X,24H ****** VALUE OF ITMAX =,I5,8HEXCEEDS ,                 75400
     &44HDIMENSION OF DHMX, PROGRAM TERMINATED ******)                     75500
 4170 FORMAT(5X,48HINITIAL PRESSURE HEAD OR MOISTURE CONTENT WAS SE,       75600
     & 24HT TO A CONSTANT VALUE OF,1PE12.3)                                75700
 4180 FORMAT(5X,48HINITIAL PRESSURE HEAD OR MOISTURE CONTENT WAS RE,       75800
     & 12HAD FROM UNIT,I5,                                                 75900
     & 20H A SCALING FACTOR OF,1PE12.3,9H WAS USED)                        76000
 4190 FORMAT(5X,'EQUILLIBRIUM PROFILE USED TO INITIALIZE PRESSURE',        76100
     & 27H HEADS ABOVE WATER TABLE AT,F10.2,1X,A4,1X,                      76200
     & 12HBELOW ORIGIN/5X,                                                 76300
     & 57HEQUILLIBRIUM PROFILE ONLY USED UNTIL PRESSURE HEADS EQUAL,       76400
     &  F10.2,1X,A4/5X,                                                    76500
     & 20HPRESSURE HEADS BELOW,F10.2,1X,A4,16H ARE HYDROSTATIC)            76600
 4200 FORMAT(1H ,50X,18HDEPTH FROM SURFACE)                                76700
 4210 FORMAT('     INITIAL CONCENTRATION SET TO A CONSTANT VALUE OF ',     76800
     &1PE12.3)                                                             76900
 4220 FORMAT('     INITIAL CONCENTRATION WAS READ FROM UNIT',I5,           77000
     &' A SCALING FACTOR OF, ',1PE12.3,' WAS USED')                        77100
 4230 FORMAT('  INITIAL MOISTURE CONTENT AT ROW ',I3,' COLUMN ',           77200
     &I3,' IS LESS THAN OR EQUAL TO 0.',/'  PROGRAM TERMINATED')           77300
      END                                                                  77400
      SUBROUTINE VSTMER                                                    77500
C******                                                                    77600
CVSTMER                                                                    77700
C******                                                                    77800
C                                                                          77900
C     PURPOSE: TO CONTROL THE TIME SEQUENCE OF SIMULATION                  78000
C     AND TO READ NEW BOUNDARY CONDITION DATA                              78100
C                                                                          78200
C ---------------------------------------------------------------          78300
C                                                                          78400
C   SPECIFICATIONS FOR ARRAYS AND SCALARS                                  78500
C                                                                          78600
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                  78700
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2                 78800
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                                78900
      COMMON/KCON/HX(1600),NTYP(1600)                                      79000
      COMMON/MPROP/THETA(1600),THLST(1600)                                 79100
      COMMON/PRESS/P(1600),PXXX(1600),CS1,CS2                              79200
      COMMON/DISCH/Q(1600),QQ(1600),ETOUT,ETOUT1                           79300
      COMMON/HCON/HCND(1600),HKLL(1600),HKTT(1600)                         79400
      COMMON/DUMM/DUM(1600)                                                79500
      COMMON/SPFC/JSPX(3,25,8),NFC(8),JLAST(8),NFCS                        79600
      COMMON/PTET/DPTH(1600),RT(1600),RDC(6,25),ETCYC,                     79700
     &PEVAL(25),PTVAL(25),PET,PEV,HROOT,HA,SRES,RTDPTH,                    79800
     &RTBOT,RTTOP,NPV                                                      79900
      COMMON/PND/POND                                                      80000
      COMMON/PLOTT/PLTIM(50),IJOBS(50),JPLT,NPLT,NOBS                      80100
```

```
      COMMON/SCON/DHMX(201),DELT,HMAX,TMAX,EPS,NUMT,ITMAX,MINIT,ITEST      80200
      COMMON/SCN1/TMPX,TMLT,DLTMX,DLTMIN,TRED                              80300
      COMMON/TCON/STIM,DSMAX,KTIM,NIT,NIT1,KP                              80400
      COMMON/JCON/JSTOP,JFLAG                                              80500
      COMMON/TRXX/DX1(1600),DX2(1600),DZ1(1600),DZ2(1600),VX(1600),        80600
     &VZ(1600),CC(1600),COLD(1600),CS(1600),QT(1600),NCTYP(1600),          80700
     &RET(1600)                                                            80800
      LOGICAL TRANS,TRANS1,SORP,SSTATE                                     80900
      COMMON/TRXY/MB9(72),NMB9,EPS1,TRANS,TRANS1,SORP,SSTATE               81000
      LOGICAL RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                           81100
      LOGICAL F7P,F11P,F8P,F9P,F6P,PRNT                                    81200
      COMMON/LOG1/RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                       81300
      COMMON/LOG2/F7P,F11P,F8P,F9P,F6P,PRNT                                81400
      CHARACTER*80 TITL                                                    81500
      CHARACTER*4 ZUNIT,TUNIT,CUNX                                         81600
      COMMON/SCHAR/TITL,ZUNIT,TUNIT,CUNX                                   81700
      DIMENSION IDUM(0600)                                                 81800
      SAVE STERR,KPLT                                                      81900
C-----------------------------------------------------------------------   82000
C                                                                          82100
C   ADVANCE TO NEXT TIME STEP                                              82200
C                                                                          82300
      KTIM=KTIM+1                                                          82400
      IF (KTIM.NE.1.AND.JSTOP.EQ.1) RETURN                                 82500
      JSTOP=0                                                              82600
      JPLT=0                                                               82700
      NIT=0                                                                82800
      NIT1=0                                                               82900
      IF(KTIM.EQ.1) KPLT=1                                                 83000
      IF(JFLAG.EQ.1) THEN                                                  83100
C                                                                          83200
C ..........................................................               83300
C                                                                          83400
C    READ DATA FOR NEW RECHARGE PERIOD                                     83500
C ..........................................................               83600
C                                                                          83700
      READ (05,*) TPER,DELT                                                83800
C                                                                          83900
C   CHECK FOR END OF SIMULATION                                            84000
C                                                                          84100
      IF(TPER.GE.999998.) THEN                                             84200
      WRITE (06,4070) TMAX,STIM                                            84300
      STOP                                                                 84400
      END IF                                                               84500
      READ (05,*) TMLT,DLTMX,DLTMIN,TRED                                   84600
      KP=KP+1                                                              84700
      SSTATE=.FALSE.                                                       84800
      WRITE (06,4000) KP,TPER,TUNIT,DELT,TUNIT,TMLT,DLTMX,TUNIT,DLTMIN,    84900
     *TUNIT,TRED                                                           85000
      READ (05,*) DSMAX,STERR                                              85100
      READ (05,*) POND                                                     85200
      WRITE (06,4020) DSMAX,STERR,POND                                     85300
      READ (05,*) PRNT                                                     85400
      READ (05,*) BCIT,ETSIM,SEEP                                          85500
      WRITE (06,4010) PRNT,BCIT,ETSIM,SEEP                                 85600
      DSMAX=ABS(DSMAX)                                                     85700
      ETOUT=0                                                              85800
      ETOUT1=0                                                             85900
C                                                                          86000
C    READ SEEPAGE FACE DATA                                                86100
C                                                                          86200
      IF(SEEP) THEN                                                        86300
      READ (05,*) NFCS                                                     86400
      DO 20 K=1,NFCS                                                       86500
      READ (05,*) JJ,JLAST(K)                                              86600
      NFC(K)=JJ                                                            86700
      READ (05,*) ((JSPX(L,J,K),L=2,3),J=1,JJ)                             86800
```

```
      DO 10 J=1,JJ                                                      86900
      J1=JSPX(2,J,K)                                                    87000
      N1=JSPX(3,J,K)                                                    87100
      N2=NLY*(N1-1)+J1                                                  87200
      JSPX(1,J,K)=N2                                                    87300
      Q(N2)=0.                                                          87400
      QQ(N2)=0.                                                         87500
      NCTYP(N2)=0                                                       87600
      CS(N2)=0                                                          87700
      IF(J.GT.JLAST(K)) THEN                                            87800
      NTYP(N2)=3                                                        87900
      ELSE                                                              88000
      NTYP(N2)=1                                                        88100
      IF(CS1.EQ.1.) THEN                                                88200
      Z1=DZZ(J1)                                                        88300
      ELSE                                                              88400
      Z1=DZZ(J1)*CS1+(RX(N1))*CS2                                       88500
      END IF                                                            88600
      P(N2)=-Z1                                                         88700
      END IF                                                            88800
   10 CONTINUE                                                          88900
   20 CONTINUE                                                          89000
      END IF                                                            89100
C                                                                       89200
C   READ IN NEW BOUNDARY CONDITIONS FOR RECHARGE PERIOD                 89300
C    IF IBC=0, POINT BOUNDARY CONDITIONS ARE READ IN.                   89400
C    IF IBC=1, LINE BOUNDARY CONDITIONS ARE READ IN, AND IT IS NECESSARY 89500
C     TO SPECIFY FOUR POINTS ON THE LINE--THIS ALLOWS VERTICAL OR HORI-  89600
C     ZONTAL LINES TO BE READ IN INDISCRIMINATELY.  THE SEQUENCE IS:     89700
C     TOP ROW, BOTTOM ROW, LEFT COLUMN, RIGHT COLUMN, CODE, AND FLUX OR  89800
C     PRESSURE HEAD FOR BOUNDARY CONDITION.                             89900
C                                                                       90000
      READ (05,*) IBC                                                   90100
      IF(IBC.GT.0) GO TO 40                                             90200
   30 IF (TRANS) THEN                                                   90300
      READ(05,*) JJ,NN,NTX,PFDUM,NTC,CF                                 90400
      ELSE                                                              90500
      READ (05,*) JJ,NN,NTX,PFDUM                                       90600
      CF=0                                                              90700
      NTC=0                                                             90800
      END IF                                                            90900
      IF(JJ.GE.999998) GO TO 90                                         91000
      JJT=JJ                                                            91100
      JJB=JJ                                                            91200
      NNL=NN                                                            91300
      NNR=NN                                                            91400
      GO TO 50                                                          91500
   40 IF (TRANS) THEN                                                   91600
      READ(05,*) JJT,JJB,NNL,NNR,NTX,PFDUM,NTC,CF                       91700
      ELSE                                                              91800
      READ (05,*) JJT,JJB,NNL,NNR,NTX,PFDUM                             91900
      CF=0                                                              92000
      NTC=0                                                             92100
      END IF                                                            92200
      IF(JJT.GE.999) GO TO 90                                           92300
   50 CONTINUE                                                          92400
      DO 80 JJ=JJT,JJB                                                  92500
      DO 80 NN=NNL,NNR                                                  92600
      IN=NLY*(NN-1)+JJ                                                  92700
      CS(IN)=CF                                                         92800
      IF(NTC.EQ.1) CC(IN)=CF                                            92900
      NCTYP(IN)=NTC                                                     93000
      IF(NTX.NE.6) GO TO 60                                             93100
      NTYP(IN)=2                                                        93200
      QQ(IN)=PFDUM                                                      93300
      GO TO 80                                                          93400
   60 NTYP(IN)=NTX                                                      93500
```

```
      IF(NTX .EQ. 4)NTYP(IN)=1                                          93600
      IF(NTX.EQ.0) WRITE (06,4030) JJ,NN                                93700
      IF(CS1.EQ.1.) THEN                                                93800
      Z1=DZZ(JJ)                                                        93900
      ELSE                                                              94000
      Z1=DZZ(JJ)*CS1+(RX(NN))*CS2                                       94100
      END IF                                                            94200
      IF(NTX.EQ.1) P(IN)=PFDUM-Z1                                       94300
      IF(NTX.EQ.4) P(IN)=PFDUM                                          94400
      IF(NTX.EQ.2) GO TO 70                                             94500
      QQ(IN)=0                                                          94600
      GO TO 80                                                          94700
   70 CONTINUE                                                          94800
C                                                                       94900
C    SET QQ TO RAINFALL RATE                                            95000
C                                                                       95100
      AREA=DXR(NN)                                                      95200
      IF(RAD)AREA=PI2*RX(NN)*DXR(NN)                                    95300
      QQ(IN)=PFDUM*AREA                                                 95400
   80 CONTINUE                                                          95500
      IF(IBC.EQ.0) GO TO 30                                             95600
      GO TO 40                                                          95700
   90 CONTINUE                                                          95800
C                                                                       95900
C     WRITE INITIAL BOUNDARY CONDITIONS FOR THIS PERIOD                 96000
C                                                                       96100
      WRITE (06,4040) KP                                                96200
      DO 110 J=1,NLY                                                    96300
      DO 100 N=1,NXR                                                    96400
      IN=NLY*(N-1)+J                                                    96500
      Q(IN)=0.                                                          96600
  100 IDUM(N)=NTYP(IN)                                                  96700
  110 WRITE (06,4050) J,(IDUM(I),I=1,NXR)                               96800
      TMPX=STIM+TPER                                                    96900
      IF(TMPX+0.5*DLTMIN.GT.TMAX) TMPX=TMAX                             97000
C                                                                       97100
C   CALCULATE NEW COEFFICIENTS                                          97200
C                                                                       97300
      IF(KTIM.NE.1)CALL VSCOEF                                          97400
      END IF                                                            97500
C                                                                       97600
C   INITIALIZE REQUIRED ARRAYS FOR NEW BOUNDARY CONDITION, UPDATE       97700
C   PXXX,THLST.  COMPUTE MAXIMUM HEAD CHANGE DURING LAST TIME STEP      97800
C                                                                       97900
      PDIF=0.                                                           98000
      IF(KTIM.NE.1.AND..NOT.SSTATE) THEN                                98100
      DO 120 J=2,NLYY                                                   98200
      DO 120 N=2,NXRR                                                   98300
      IN=NLY*(N-1)+J                                                    98400
      IF(HX(IN).EQ.0.) GO TO 120                                        98500
      P12=P(IN)-PXXX(IN)                                                98600
      PTMP=ABS(P12)                                                     98700
      IF(PTMP.GT.PDIF)PDIF=PTMP                                         98800
      PXXX(IN)=P(IN)                                                    98900
      THLST(IN)=THETA(IN)                                               99000
  120 CONTINUE                                                          99100
C                                                                       99200
C    CHECK FOR STEADY STATE                                             99300
C                                                                       99400
      IF(PDIF.LE.STERR) THEN                                            99500
      SSTATE=.TRUE.                                                     99600
      WRITE(6,4060) STIM,KTIM                                           99700
      END IF                                                            99800
      END IF                                                            99900
      JFLAG=0                                                          100000
C                                                                      100100
C   INITIALIZE DHMX                                                    100200
```

```
C                                                                       100300
      DO 130 K=1,201                                                    100400
  130 DHMX(K)=0.                                                        100500
C                                                                       100600
C     ADVANCE DELT AND RESET TO PROPER LENGTH IF NECESSARY              100700
C                                                                       100800
      DLTOLD=DELT                                                       100900
      DELT= TMLT*DELT                                                   101000
C                                                                       101100
C     MAXIMUM PERMISSABLE HEAD CHANGE CHECK                             101200
C                                                                       101300
      IF(KTIM.GE.2) THEN                                                101400
      IF((PDIF*DELT/DLTOLD).GT.DSMAX)DELT=DLTOLD*DSMAX*.98/PDIF         101500
      END IF                                                            101600
      IF(ABS(TMPX-PLTIM(KPLT)).LT.DLTMIN) PLTIM(KPLT)=TMPX              101700
      T1=DMIN1(TMPX,PLTIM(KPLT))                                        101800
      T2=T1-STIM                                                        101900
      IF(DELT.GT.(T2-DLTMIN)) DELT=T2                                   102000
      IF(DELT.LT.DLTMIN)DELT=DLTMIN                                     102100
      IF(DELT.GT.DLTMX)DELT=DLTMX                                       102200
      IF(T1.NE.PLTIM(KPLT).OR.T2-DELT.GT.0.5*DLTMIN) GO TO 140          102300
      KPLT=KPLT+1                                                       102400
      JPLT=1                                                            102500
  140 IF(DELT.LT.DLTMIN)DELT=DLTMIN                                     102600
      STIM=STIM+DELT                                                    102700
      IF (TMPX-STIM.LT.0.5*DLTMIN) JFLAG=1                              102800
      IF(TMAX-STIM.LT.0.5*DLTMIN.OR.KTIM.GT.NUMT) THEN                  102900
      JSTOP=1                                                           103000
      JPLT=1                                                            103100
      END IF                                                            103200
      RETURN                                                            103300
 4000 FORMAT(6X,'DATA FOR RECHARGE PERIOD ',I5//10X,                    103400
     &23HLENGTH OF THIS PERIOD =,1PE12.3,1X,A4/10X,                     103500
     &45HLENGTH OF INITIAL TIME STEP FOR THIS PERIOD =,1PE10.3,1X,A4/   103600
     &10X,27HMULTIPLIER FOR TIME STEP = ,1PE10.3,/10X,                  103700
     &25HMAXIMUM TIME STEP SIZE = ,1PE10.3,1X,A4/10X,                   103800
     &25HMINIMUM TIME STEP SIZE = ,1PE10.3,1X,A4,                       103900
     &/10X,'TIME STEP REDUCTION FACTOR = ',1PE10.3)                     104000
 4010 FORMAT(15X,37HPRINT SOLUTION AFTER EVERY TIME STEP?,1X,L1/        104100
     &15X,'SIMULATE EVAPORATION? ',L1/                                  104200
     &15X,29HSIMULATE EVAPOTRANSPIRATION? ,L1/                          104300
     &15X,24HSIMULATE SEEPAGE FACES? ,L1/)                              104400
 4020 FORMAT(                                                           104500
     &15X,55HMAXIMUM PRESSURE HEAD CHANGE ALLOWED IN ONE TIME STEP =,   104600
     &F8.3/15X,'STEADY-STATE CLOSURE CRITERION = ',1PE10.3/             104700
     &15X,'MAXIMUM DEPTH OF PONDING = ',1PE10.3)                        104800
 4030 FORMAT(1H ,1X,10(1H*),41HWARNING --- NODE TYPE OF 0 ASSIGNED TO BO 104900
     &12HUNDARY NODE ,2I4,43H SPECIFIED FLUX OR PRESSURE HEAD NOT ASSIGN 105000
     &2HED)                                                             105100
 4040 FORMAT(6X,41HNODE TYPE AND INITIAL BOUNDARY CONDITIONS,           105200
     &12H FOR PERIOD ,I4/6X,8HLEGEND: /15X,17H0 = INTERIOR CELL/        105300
     &15X,32H1 = SPECIFIED PRESSURE HEAD CELL/15X,                      105400
     &23H2 = SPECIFIED FLUX CELL/                                       105500
     & 15X,31H3 = POTENTIAL SEEPAGE FACE NODE/                          105600
     & 15X,43H5 = NODE FOR WHICH EVAPORATION IS PERMITTED//)            105700
 4050 FORMAT(1H ,I5,5X,80I1)                                            105800
 4060 FORMAT(6X,100(1H*)/5X,                                            105900
     &'STEADY STATE REACHED AT TIME = ',E12.4,'   TIME STEP NUMBER = '  106000
     &,I5//)                                                            106100
 4070 FORMAT(6X,100(1H*),/,5X,17HEND OF SIMULATION/,                    106200
     &5X,33HMAXIMUM SIMULATION TIME (TMAX) = ,E15.4/,                   106300
     &5X,33HELAPSED SIMULATION TIME (STIM) = ,E15.4/,                   106400
     &6X,100(1H*))                                                      106500
      END                                                               106600
      SUBROUTINE VSMGEN                                                 106700
C******                                                                 106800
CVSMGEN                                                                 106900
```

```
C******                                                                 107000
C                                                                       107100
C    PURPOSE:  TO SET UP COEFFICIENT MATRICES AND CALL                  107200
C          SOLUTION ALGORITHM                                           107300
C                                                                       107400
C-----------------------------------------------------------------------107500
C                                                                       107600
C   SPECIFICATIONS FOR ARRAYS AND SCALARS                               107700
C                                                                       107800
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               107900
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2              108000
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                             108100
      COMMON/KCON/HX(1600),NTYP(1600)                                   108200
      COMMON/RPROP/HK(10,100),HT(10,20),ANIZ(10)                        108300
      COMMON/MPROP/THETA(1600),THLST(1600)                              108400
      COMMON/PRESS/P(1600),PXXX(1600),CS1,CS2                           108500
      COMMON/DISCH/Q(1600),QQ(1600),ETOUT,ETOUT1                        108600
      COMMON/HCON/HCND(1600),HKLL(1600),HKTT(1600)                      108700
      COMMON/EQUAT/A(1600),B(1600),C(1600),D(1600),E(1600),RHS(1600),   108800
     &XI(1600)                                                          108900
      COMMON/JTXX/JTEX(1600)                                            109000
      COMMON/PTET/DPTH(1600),RT(1600),RDC(6,25),ETCYC,                  109100
     &PEVAL(25),PTVAL(25),PET,PEV,HROOT,HA,SRES,RTDPTH,                 109200
     &RTBOT,RTTOP,NPV                                                   109300
      COMMON/WGT/WUS,WDS                                                109400
      COMMON/SCON/DHMX(201),DELT,HMAX,TMAX,EPS,NUMT,ITMAX,MINIT,ITEST   109500
      COMMON/SCN1/TMPX,TMLT,DLTMX,DLTMIN,TRED                           109600
      COMMON/TCON/STIM,DSMAX,KTIM,NIT,NIT1,KP                           109700
      COMMON/JCON/JSTOP,JFLAG                                           109800
      LOGICAL RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                        109900
      COMMON/LOG1/RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                    110000
      CHARACTER*80 TITL                                                 110100
      CHARACTER*4 ZUNIT,TUNIT,CUNX                                      110200
      COMMON/SCHAR/TITL,ZUNIT,TUNIT,CUNX                                110300
      DIMENSION PITT(1600)                                              110400
      SAVE PITT                                                         110500
C                                                                       110600
C .....................................................................  110700
C  START OF LINEARIZATION ITERATION LOOP                                110800
C. .....................................................................  110900
C                                                                       111000
C    UPDATE COEFFICIENTS                                                111100
C                                                                       111200
      I13=0                                                             111300
C                                                                       111400
C     ESTABLISH TIME-DEPENDENT PARAMETERS GOVERNING EVAPORATION AND     111500
C     TRANSPIRATION.  DETERMINE ROOT ACTIVITY.                          111600
C                                                                       111700
   10 IF ( BCIT.OR. ETSIM) THEN                                         111800
      CALL VSPET                                                        111900
      DO 20 J=2,NLYY                                                    112000
      DO 20 I=2,NXRR                                                    112100
      N=NLY*(I-1)+J                                                     112200
      IF(HX(N).GT.0) THEN                                               112300
      IF(ETSIM) RT(N)=VSRDF(DPTH(N),DELZ(J))                            112400
      Q(N)=0.0                                                          112500
      END IF                                                            112600
   20 CONTINUE                                                          112700
      END IF                                                            112800
   30 IF (NIT.NE.0) CALL VSCOEF                                         112900
C                                                                       113000
C -------------------  UPDATE BOUNDARY AND FLUX CONDITIONS ------------ 113100
C                                                                       113200
      IF(BCIT)CALL VSEVAP                                               113300
      IF (ETSIM)CALL VSPLNT                                             113400
      IF(SEEP) CALL VSSFAC                                              113500
C                                                                       113600
```

```
C ..................................................................   113700
C                                                                      113800
C         LOOP TO CALCULATE COEFFICIENT MATRIX                         113900
C ..................................................................   114000
C                                                                      114100
      DO 40 J=2,NLYY                                                   114200
      DO 40 I=2,NXRR                                                   114300
      N=NLY*(I-1)+J                                                    114400
      IF(HX(N).GT.0.) THEN                                             114500
      JM1=N-1                                                          114600
      JP1=N+1                                                          114700
      IM1=N-NLY                                                        114800
      IP1=N+NLY                                                        114900
      VOL=DXR(I)*DELZ(J)                                               115000
      IF(RAD)VOL=PI2*RX(I)*DXR(I)*DELZ(J)                              115100
      JJ=JTEX(N)                                                       115200
C                                                                      115300
C   CALCULATE STORAGE TERMS                                            115400
C                                                                      115500
      IF(CS1.EQ.1.) THEN                                               115600
      Z1=DZZ(J)                                                        115700
      ELSE                                                             115800
      Z1=DZZ(J)*CS1+(RX(I))*CS2                                        115900
      END IF                                                           116000
      PTMP=P(N)+Z1                                                     116100
      SCAP=VSDTHU(PTMP,JJ,HK)                                          116200
      GSF=VOL*SCAP                                                     116300
      SS=HK(JJ,2)/HK(JJ,3)                                             116400
      GSS=VOL*THETA(N)*SS                                              116500
      G1=0                                                             116600
C                                                                      116700
C   APPLY NEWTON-RAPHSON LINEARIZATION TO STORAGE TERM.                116800
C   PITT HOLDS STORAGE TERMS FROM PREVIOUS ITERATION.                  116900
C                                                                      117000
      IF(NIT.GT.0.AND.XI(N).NE.0)G1=(P(N)-PXXX(N))*(GSF+GSS-PITT(N))/  117100
     *XI(N)                                                           117200
      PITT(N)=GSF+GSS                                                  117300
      G1=-G1/DELT                                                      117400
      GSF=-GSF/DELT                                                    117500
      GSS=-GSS/DELT                                                    117600
      IF(WUS.EQ.0.) THEN                                               117700
C                                                                      117800
C   USE GEOMETRIC MEAN OR WEIGHTS FOR INTERCELL K                      117900
C                                                                      118000
      A(N)=HKLL(N)*DSQRT(HCND(IM1)*HCND(N))                            118100
      B(N)=HKTT(N)*DSQRT(HCND(JM1)*HCND(N))                            118200
      C(N)=HKLL(IP1)*DSQRT(HCND(IP1)*HCND(N))                          118300
      D(N)=HKTT(JP1)*DSQRT(HCND(JP1)*HCND(N))                          118400
      ELSE                                                             118500
C                                                                      118600
C  CHOOSE UPSTREAM WEIGHTING COEFFICIENTS                              118700
C                                                                      118800
      IF(P(IM1).LE.P(N).OR.HX(IM1).EQ.0.) THEN                         118900
      ALA=WDS                                                          119000
      BTA=WUS                                                          119100
      ELSE                                                             119200
      ALA=WUS                                                          119300
      BTA=WDS                                                          119400
      END IF                                                           119500
      IF(P(JM1).LE.P(N).OR.HX(JM1).EQ.0.) THEN                         119600
      ALB=WDS                                                          119700
      BTB=WUS                                                          119800
      ELSE                                                             119900
      ALB=WUS                                                          120000
      BTB=WDS                                                          120100
      END IF                                                           120200
      IF(P(IP1).LE.P(N).OR.HX(IP1).EQ.0.) THEN                         120300
```

```
      ALC=WDS                                                          120400
      BTC=WUS                                                          120500
      ELSE                                                             120600
      ALC=WUS                                                          120700
      BTC=WDS                                                          120800
      END IF                                                           120900
      IF(P(JP1).LE.P(N).OR.HX(JP1).EQ.0.) THEN                         121000
      ALD=WDS                                                          121100
      BTD=WUS                                                          121200
      ELSE                                                             121300
      ALD=WUS                                                          121400
      BTD=WDS                                                          121500
      END IF                                                           121600
C                                                                      121700
C   SET THE PENTA-DIAGNOL COEFFICIENT MATRIX (E IS MAIN DIAGNOL)       121800
C   AND RIGHT HAND SIDE                                                121900
C                                                                      122000
      A(N)=(ALA*HCND(IM1)+BTA*HCND(N))*HKLL(N)                         122100
      B(N)=(ALB*HCND(JM1)+BTB*HCND(N))*HKTT(N)                         122200
      C(N)=(ALC*HCND(IP1)+BTC*HCND(N))*HKLL(IP1)                       122300
      D(N)=(ALD*HCND(JP1)+BTD*HCND(N))*HKTT(JP1)                       122400
      END IF                                                           122500
      E(N)=-A(N)-B(N)-C(N)-D(N)                                        122600
      RHS(N)=VOL*(THETA(N)-THLST(N))/DELT-(Q(N)+QQ(N))-(A(N)*P(IM1)+B(N) 122700
     &*P(JM1)+C(N)*P(IP1)+D(N)*P(JP1)+(E(N)+GSS)*P(N))+GSS*PXXX(N)     122800
      E(N)=E(N)+GSF+GSS+G1                                             122900
      END IF                                                           123000
   40 CONTINUE                                                         123100
C                                                                      123200
C    CALL SOLUTION ALGORITHM                                           123300
C                                                                      123400
      NIT=NIT+1                                                        123500
      CALL SLVSIP                                                      123600
      IF(NIT.LT.MINIT) GO TO 30                                        123700
C                                                                      123800
C   IF SOLUTION HAS BEEN FOUND THEN RETURN                             123900
C                                                                      124000
      IF(ITEST.EQ.0) RETURN                                            124100
      IF(NIT.LE.ITMAX) GO TO 30                                        124200
C                                                                      124300
C   MAXIMUM NUMBER OF ITERATIONS EXCEEDED                              124400
C                                                                      124500
      WRITE (6,4000) NIT,KTIM,STIM,TUNIT                               124600
C                                                                      124700
C   AUTOMATICALLY REDUCE TIME STEP SIZE, BUT NOT MORE                  124800
C   THAN TWICE.                                                        124900
C                                                                      125000
      IF(DELT.LE.DLTMIN.OR.I13.GT.2.OR.TRED.LE.0) THEN                 125100
      IF(.NOT.ITSTOP)RETURN                                            125200
C                                                                      125300
C   TERMINATE SIMULATION.                                              125400
C                                                                      125500
      JSTOP=1                                                          125600
      JFLAG=1                                                          125700
      RETURN                                                           125800
      ELSE                                                             125900
      I13=I13+1                                                        126000
      DELTT=DELT*TRED                                                  126100
      IF(DELTT.LT.DLTMIN) DELTT=DLTMIN                                 126200
      WRITE(6,4010) DELTT                                              126300
      STIM=STIM-DELT+DELTT                                             126400
      DELT=DELTT                                                       126500
C                                                                      126600
C   RESET HEADS TO VALUES AT END OF PREVIOUS TIME STEP.                126700
C                                                                      126800
      DO 50 II=1,NNODES                                                126900
      IF(NTYP(II).NE.1.AND.HX(II).GT.0) P(II)=PXXX(II)                 127000
```

```
   50 CONTINUE                                                          127100
      NIT=1                                                             127200
      GO TO 10                                                          127300
      END IF                                                            127400
 4000 FORMAT(5X,100(1H*)/5X,'EXCEEDED PERMITTED NUMBER OF ITERATIONS',  127500
     &'  ( =',I4,')'                                                   127600
     & /5X,'TIME STEP NUMBER',I4/5X,'ELAPSED TIME = ',                  127700
     & 1PE12.3,1X,A4  /5X,100(1H*))                                     127800
 4010 FORMAT(5X,'TIME STEP SIZE REDUCED TO ',E12.4)                     127900
      END                                                               128000
      SUBROUTINE VSSIP                                                  128100
C                                                                       128200
C*****                                                                  128300
CVSSIP                                                                  128400
C*****                                                                  128500
C                                                                       128600
C     PURPOSE: TO SOLVE THE  MATRIX EQUATIONS USING THE                 128700
C     STRONGLY IMPLICIT METHOD                                          128800
C                                                                       128900
C -------------------------------------------------------------------   129000
C                                                                       129100
C   SPECIFICATIONS FOR ARRAYS AND SCALARS                               129200
C                                                                       129300
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               129400
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2              129500
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                             129600
      COMMON/KCON/HX(1600),NTYP(1600)                                   129700
      COMMON/RPROP/HK(10,100),HT(10,20),ANIZ(10)                        129800
      COMMON/PRESS/P(1600),PXXX(1600),CS1,CS2                           129900
      COMMON/EQUAT/A(1600),B(1600),C(1600),D(1600),E(1600),RHS(1600),   130000
     &XI(1600)                                                          130100
      COMMON/JTXX/JTEX(1600)                                            130200
      COMMON/SCON/DHMX(201),DELT,HMAX,TMAX,EPS,NUMT,ITMAX,MINIT,ITEST   130300
      COMMON/SCN1/TMPX,TMLT,DLTMX,DLTMIN,TRED                           130400
      COMMON/TCON/STIM,DSMAX,KTIM,NIT,NIT1,KP                           130500
      COMMON/TRXX/DX1(1600),DX2(1600),DZ1(1600),DZ2(1600),VX(1600),     130600
     &VZ(1600),CC(1600),COLD(1600),CS(1600),QT(1600),NCTYP(1600),       130700
     &RET(1600)                                                         130800
      LOGICAL TRANS,TRANS1,SORP,SSTATE                                  130900
      COMMON/TRXY/MB9(72),NMB9,EPS1,TRANS,TRANS1,SORP,SSTATE            131000
      DIMENSION IORDER(21)                                              131100
      DIMENSION DEL(1600),ETA(1600),V(1600),TEMP(100),HM(30)            131200
      SAVE HM,W1,W9,L2                                                  131300
C                                                                       131400
C-------------------------------------------------------------------    131500
C                                                                       131600
      DATA IORDER/1,2,3,4,5,1,2,3,4,5,11*1/                             131700
C                                                                       131800
C     COMPUTE ITERATION PARAMETERS                                      131900
C                                                                       132000
      J2=NXR-2                                                          132100
      I2=NLY-2                                                          132200
      L2=5                                                              132300
      PL2=L2-1                                                          132400
      W=0.                                                              132500
      PIE=0.                                                            132600
      W9=100.                                                           132700
C                                                                       132800
C  COMPUTE MAXIMUM PARAMETER                                            132900
C                                                                       133000
      DO 10 I=2,NLYY                                                    133100
      DO 10 J=2,NXRR                                                    133200
      N=NLY*(J-1)+I                                                     133300
      IF(HX(N).GT.0.) THEN                                              133400
      IM1=JTEX(N)                                                       133500
      PIE=PIE+1.                                                        133600
      DX=DXR(J)/RX(NXR)                                                 133700
```

```
      DY=DELZ(I)/DZZ(NLY)                                               133800
      DX3=DX*DX                                                         133900
      DY2=DY*DY                                                         134000
      W=W+1-DMIN1((DX3+DX3)/(1.+ANIZ(IM1)*DX3/DY2),(DY2+DY2)/(1+DY2/    134100
     &(ANIZ(IM1)*DX3)))                                                 134200
      END IF                                                            134300
   10 CONTINUE                                                          134400
      W=W/PIE                                                           134500
C                                                                       134600
C COMPUTE PARAMETERS IN GEOMETRIC SEQUENCE                              134700
C                                                                       134800
      PJ=-1.                                                            134900
      DO 20 I=1,L2                                                      135000
      PJ=PJ+1.                                                          135100
   20 TEMP(I)=1. -(1. -W)**(PJ/PL2)                                     135200
C                                                                       135300
C ORDER SEQUENCE OF PARAMETERS                                          135400
C                                                                       135500
      DO 30 J=1,L2                                                      135600
   30 HM(J)=TEMP(IORDER(J))                                             135700
      WRITE (06,4000) L2,(HM(J),J=1,L2)                                 135800
      RETURN                                                            135900
C                                                                       136000
C STRONGLY IMPLICIT ALGORITHM                                           136100
C                                                                       136200
      ENTRY SLVSIP                                                      136300
      I2=NLY-2                                                          136400
      J2=NXR-2                                                          136500
C                                                                       136600
C      SELECT ITERATION PARAMETER.  INITIALIZE ARRAYS                   136700
C                                                                       136800
      IF(TRANS1) THEN                                                   136900
C                                                                       137000
C  IF TRANS1=T  TRANSPORT EQUATION IS SOLVED                            137100
C           =F  FLOW EQUATION IS SOLVED                                 137200
C                                                                       137300
      NT=NIT1                                                           137400
      ELSE                                                              137500
      NT=NIT                                                            137600
      END IF                                                            137700
      IF(MOD(NT,L2).EQ.0.OR.NT.EQ.1)NTH=0                               137800
      NTH=NTH+1                                                         137900
      W=HM(NTH)                                                         138000
      ITEST=0                                                           138100
      DO 40 I=1,NNODES                                                  138200
      DEL(I)=0.                                                         138300
      ETA(I)=0.                                                         138400
      V(I)=0.                                                           138500
   40 XI(I)=0.                                                          138600
      BIGI=0.                                                           138700
      BIGI1=0.                                                          138800
C                                                                       138900
C CHOOSE SIP NORMAL OR REVERSE ALGORITHM                                139000
C                                                                       139100
      IF(MOD(NT,2)) 50,80,50                                            139200
C .....................................................................  139300
C ORDER EQUATIONS WITH ROW 1 FIRST  -  3X3 EXAMPLE:                     139400
C    1 2 3                                                              139500
C    4 5 6                                                              139600
C    7 8 9                                                              139700
C .....................................................................  139800
   50 DO 60 I=2,NLYY                                                    139900
      DO 60 J=2,NXRR                                                    140000
      N=I+NLY*(J-1)                                                     140100
C                                                                       140200
C   ---- SKIP COMPUTATIONS OF NODE IS OUTSIDE OF SOLUTION DOMAIN        140300
C                                                                       140400
```

```
      IF(HX(N).EQ.0.) GO TO 60                                          140500
      IF((NTYP(N).EQ.1.AND.(.NOT.TRANS1)).OR.(TRANS1.AND.(NCTYP(N).EQ.1 140600
     *)))GO TO 60                                                       140700
      NL=N-NLY                                                          140800
      NA=N-1                                                            140900
      NB=N+1                                                            141000
C                                                                       141100
C     --- SIP "NORMAL" ALGORITHM-----                                   141200
C     --- FORWARD SUBSTITUTE, COMPUTING INTERMEDIATE VECTOR V --        141300
C                                                                       141400
      CH=DEL(NA)*B(N)/(1. +W*DEL(NA))                                   141500
      GH=ETA(NL)*A(N)/(1. +W*ETA(NL))                                   141600
      BH=B(N)-W*CH                                                      141700
      DH=A(N)-W*GH                                                      141800
      EH=E(N)+W*CH+W*GH                                                 141900
      FH=C(N)-W*CH                                                      142000
      HH=D(N)-W*GH                                                      142100
      ALFA=BH                                                           142200
      BETA=DH                                                           142300
      GAMA=EH-ALFA*ETA(NA)-BETA*DEL(NL)                                 142400
      DEL(N)=FH/GAMA                                                    142500
      ETA(N)=HH/GAMA                                                    142600
      RES=RHS(N)                                                        142700
      V(N)=(HMAX*RES-ALFA*V(NA)-BETA*V(NL))/GAMA                        142800
   60 CONTINUE                                                          142900
C                                                                       143000
C  ---BACK SUBSTITUTE FOR VECTOR XI                                     143100
C                                                                       143200
      DO 70 I=1,I2                                                      143300
      I3=NLY-I                                                          143400
      DO 70 J=1,J2                                                      143500
      J3=NXR-J                                                          143600
      N=I3+NLY*(J3-1)                                                   143700
      IF(HX(N).EQ.0.) GO TO 70                                          143800
      IF((NTYP(N).EQ.1.AND.(.NOT.TRANS1)).OR.(TRANS1.AND.(NCTYP(N).EQ.1 143900
     *)))GO TO 70                                                       144000
      XI(N)=V(N)-DEL(N)*XI(N+NLY)-ETA(N)*XI(N+1)                        144100
C                                                                       144200
C      FIND MAXIMUM HEAD CHANGE                                         144300
C                                                                       144400
      TCHK=ABS(XI(N))                                                   144500
      IF(TCHK.GE.BIGI) THEN                                             144600
      BIGI=TCHK                                                         144700
      BIGI1=XI(N)                                                       144800
      END IF                                                            144900
   70 CONTINUE                                                          145000
      GO TO 110                                                         145100
C                                                                       145200
C.....................................................................  145300
C  ---ORDER EQUATIONS WITH THE LAST ROW FIRST  -  3X3 EXAMPLE          145400
C           7 8 9                                                       145500
C           4 5 6                                                       145600
C           1 2 3                                                       145700
C.....................................................................  145800
C                                                                       145900
   80 DO 90 II=1,I2                                                     146000
      I=NLY-II                                                          146100
      DO 90 J=2,NXRR                                                    146200
      N=I+NLY*(J-1)                                                     146300
      NL=N-NLY                                                          146400
      NA=N-1                                                            146500
      NB=N+1                                                            146600
C                                                                       146700
C  -- SKIP COMPUTATIONS IF NODE IS OUTSIDE OF SOLUTION DOMAIN           146800
C                                                                       146900
      IF(HX(N).EQ.0.) GO TO 90                                          147000
      IF((NTYP(N).EQ.1.AND.(.NOT.TRANS1)).OR.(TRANS1.AND.(NCTYP(N).EQ.1 147100
```

```
     *)))GO TO 90                                                          147200
C                                                                          147300
C ------ SIP "REVERSE" ALGORITHM                                          147400
C --- FORWARD SUBSTITUTE, COMPUTING INTERMEDIATE VECTOR V                  147500
C                                                                          147600
      CH=DEL(NB)*D(N)/(1. +W*DEL(NB))                                      147700
      GH=ETA(NL)*A(N)/(1. +W*ETA(NL))                                      147800
      BH=D(N)-W*CH                                                         147900
      DH=A(N)-W*GH                                                         148000
      EH=E(N)+W*CH+W*GH                                                    148100
      FH=C(N)-W*CH                                                         148200
      HH=B(N)-W*GH                                                         148300
      ALFA=BH                                                              148400
      BETA=DH                                                              148500
      GAMA=EH-ALFA*ETA(NB)-BETA*DEL(NL)                                    148600
      DEL(N)=FH/GAMA                                                       148700
      ETA(N)=HH/GAMA                                                       148800
      RES=RHS(N)                                                           148900
      V(N)=(HMAX*RES-ALFA*V(NB)-BETA*V(NL))/GAMA                           149000
   90 CONTINUE                                                             149100
C                                                                          149200
C --- BACK SUBSTITUTE FOR VECTOR XI                                        149300
C                                                                          149400
      DO 100 I3=2,NLYY                                                     149500
      DO 100 J=1,J2                                                        149600
      J3=NXR-J                                                             149700
      N=I3+NLY*(J3-1)                                                      149800
      IF(HX(N).EQ.0.) GO TO 100                                            149900
      IF((NTYP(N).EQ.1.AND.(.NOT.TRANS1)).OR.(TRANS1.AND.(NCTYP(N).EQ.1    150000
     *)))GO TO 100                                                         150100
      XI(N)=V(N)-DEL(N)*XI(N+NLY)-ETA(N)*XI(N-1)                           150200
C                                                                          150300
C      FIND MAXIMUM HEAD CHANGE                                            150400
C                                                                          150500
      TCHK=ABS(XI(N))                                                      150600
      IF(TCHK.GE.BIGI) THEN                                                150700
      BIGI=TCHK                                                            150800
      BIGI1=XI(N)                                                          150900
      END IF                                                               151000
  100 CONTINUE                                                             151100
C                                                                          151200
C      COMPUTE RELAXATION PARAMETER W FOR HEAD CHANGES.  ALGORITHM         151300
C      IS FROM COOLEY (1983)                                               151400
C                                                                          151500
  110 S=1.                                                                 151600
      IF(NT.GT.1.AND.W1.NE.0.0) S=BIGI1/W1                                 151700
      S1=ABS(S)                                                            151800
      IF(S.LT.-1.) THEN                                                    151900
      W=1/(S1+S1)                                                          152000
      ELSE                                                                 152100
      W=(3+S)/(3+S1)                                                       152200
      END IF                                                               152300
      IF(W.EQ.W9) W=.9*W                                                   152400
      W1=W*BIGI                                                            152500
      IF(W1.GT.DSMAX) W=DSMAX/BIGI                                         152600
      IF(BIGI1.LT.0.) W1=-W1                                               152700
C                                                                          152800
C      ADD CHANGES TO MATRIX.                                              152900
C                                                                          153000
      W9=W                                                                 153100
      IF(TRANS1) THEN                                                      153200
      DO 120 N=NLY+1,NNODES                                                153300
      IF(NCTYP(N).NE.1.AND.HX(N).GT.0.) CC(N)=CC(N)+W*XI(N)                153400
  120 CONTINUE                                                             153500
      IF(BIGI.GT.EPS1) ITEST=1                                             153600
      ELSE                                                                 153700
      DO 130 N=NLY+1,NNODES                                                153800
```

```
      IF(HX(N).GT.0.AND.NTYP(N).NE.1) P(N)=P(N)+W*XI(N)                  153900
  130 CONTINUE                                                          154000
C                                                                       154100
C      COMPARE MAXIMUM HEAD CHANGE TO CLOSURE CRITERION.                154200
C                                                                       154300
      IF(BIGI.GT.EPS) ITEST=1                                           154400
      DHMX(NIT)=BIGI                                                    154500
      END IF                                                            154600
      RETURN                                                            154700
 4000 FORMAT(1X,I5,25HSIP ITERATION PARAMETERS:,6D15.7/(28X,6D15.7/))   154800
      END                                                               154900
      SUBROUTINE VSCOEF                                                 155000
C******                                                                 155100
CVSCOEF                                                                 155200
C******                                                                 155300
C     PURPOSE: TO COMPUTE ALL VALUES OF NONLINEAR COEFFICIENTS          155400
C              USING THE MOST RECENT VALUES OF PRESSURE HEAD            155500
C -------------------------------------------------------------------   155600
C                                                                       155700
C   SPECIFICATIONS FOR ARRAYS AND SCALARS                               155800
C                                                                       155900
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               156000
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2              156100
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                             156200
      COMMON/KCON/HX(1600),NTYP(1600)                                   156300
      COMMON/RPROP/HK(10,100),HT(10,20),ANIZ(10)                        156400
      COMMON/MPROP/THETA(1600),THLST(1600)                              156500
      COMMON/PRESS/P(1600),PXXX(1600),CS1,CS2                           156600
      COMMON/HCON/HCND(1600),HKLL(1600),HKTT(1600)                      156700
      COMMON/JTXX/JTEX(1600)                                            156800
C                                                                       156900
C-------------------------------------------------------------------    157000
      DO 10 J=2,NLYY                                                    157100
      DO 10 N=2,NXRR                                                    157200
      IN=NLY*(N-1)+J                                                    157300
      IF(HX(IN).GT.0.) THEN                                             157400
      J1=JTEX(IN)                                                       157500
      HCND(IN)=0.D0                                                     157600
C                                                                       157700
C       COMPUTE PRESSURE HEADS TO USE IN FUNCTIONS                      157800
C                                                                       157900
      IF(CS1.EQ.1.) THEN                                                158000
      Z1=DZZ(J)                                                         158100
      ELSE                                                              158200
      Z1=DZZ(J)*CS1+(RX(N))*CS2                                         158300
      END IF                                                            158400
      PTMP=P(IN)+Z1                                                     158500
      HCND(IN)=VSHKU(PTMP,J1,HK)                                        158600
      THETA(IN)=VSTHU(PTMP,J1,HK)                                       158700
      END IF                                                            158800
   10 CONTINUE                                                          158900
      RETURN                                                            159000
      END                                                               159100
      SUBROUTINE VSHCMP                                                 159200
C******                                                                 159300
CVSHCMP                                                                 159400
C******                                                                 159500
C                                                                       159600
C   PURPOSE: TO COMPUTE INTERCELL CONDUCTANCES                          159700
C                                                                       159800
C -------------------------------------------------------------------   159900
C                                                                       160000
C   SPECIFICATIONS FOR ARRAYS AND SCALARS                               160100
C                                                                       160200
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               160300
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2              160400
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                             160500
```

```
      COMMON/KCON/HX(1600),NTYP(1600)                                   160600
      COMMON/RPROP/HK(10,100),HT(10,20),ANIZ(10)                        160700
      COMMON/HCON/HCND(1600),HKLL(1600),HKTT(1600)                      160800
      COMMON/JTXX/JTEX(1600)                                            160900
      LOGICAL RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                        161000
      COMMON/LOG1/RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                    161100
C                                                                       161200
C-----------------------------------------------------------------------161300
C                                                                       161400
C    COMPUTE HARMONIC MEANS OF KSAT AND GRID SPACING                    161500
C                                                                       161600
      DO 10 J=2,NLY                                                     161700
      DO 10 N=2,NXR                                                     161800
      IN=NLY*(N-1)+J                                                    161900
      JM1=IN-1                                                          162000
      NM1=IN-NLY                                                        162100
      A1=ANIZ(JTEX(IN))                                                 162200
      A2=ANIZ(JTEX(JM1))                                                162300
      IF(HX(IN).EQ.0.) GO TO 10                                         162400
      AREA=DXR(N)                                                       162500
      IF(RAD)AREA=PI2*RX(N)*DXR(N)                                      162600
C                                                                       162700
C   VERTICAL CONDUCTANCE                                                162800
C   THROUGH TOP                                                         162900
C                                                                       163000
      HKTT(IN)=2.0*A1*A2*AREA*HX(IN)*HX(JM1)/(A2*HX(JM1)*DELZ(J)+       163100
     &A1*HX(IN)*DELZ(J-1))                                              163200
      AREA=DELZ(J)                                                      163300
      IF(RAD)AREA=PI2*DELZ(J)*(RX(N)-.5 *DXR(N))                        163400
C                                                                       163500
C   HORIZONTAL OR RADIAL CONDUCTANCE                                    163600
C   THROUGH LEFT-HAND SIDE                                              163700
C                                                                       163800
      HKLL(IN)=2.0*AREA*HX(IN)*HX(NM1)/(HX(NM1)*DXR(N)+HX(IN)*DXR(N-1)) 163900
   10 CONTINUE                                                          164000
      RETURN                                                            164100
      END                                                               164200
      SUBROUTINE VSFLUX                                                 164300
C******                                                                 164400
CVSFLUX                                                                 164500
C******                                                                 164600
C                                                                       164700
C      PURPOSE: TO COMPUTE FLUXES AND MASS BALANCE                      164800
C                                                                       164900
C -------------------------------------------------------------        165000
C                                                                       165100
C   SPECIFICATIONS FOR ARRAYS AND SCALARS                               165200
C                                                                       165300
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               165400
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2              165500
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                             165600
      COMMON/KCON/HX(1600),NTYP(1600)                                   165700
      COMMON/RPROP/HK(10,100),HT(10,20),ANIZ(10)                        165800
      COMMON/MPROP/THETA(1600),THLST(1600)                              165900
      COMMON/PLOTT/PLTIM(50),IJOBS(50),JPLT,NPLT,NOBS                   166000
      COMMON/PRESS/P(1600),PXXX(1600),CS1,CS2                           166100
      COMMON/DISCH/Q(1600),QQ(1600),ETOUT,ETOUT1                        166200
      COMMON/JTXX/JTEX(1600)                                            166300
      COMMON/SCON/DHMX(201),DELT,HMAX,TMAX,EPS,NUMT,ITMAX,MINIT,ITEST   166400
      COMMON/SCN1/TMPX,TMLT,DLTMX,DLTMIN,TRED                           166500
      COMMON/TCON/STIM,DSMAX,KTIM,NIT,NIT1,KP                           166600
      COMMON/JCON/JSTOP,JFLAG                                           166700
      COMMON/TRXX/DX1(1600),DX2(1600),DZ1(1600),DZ2(1600),VX(1600),     166800
     &VZ(1600),CC(1600),COLD(1600),CS(1600),QT(1600),NCTYP(1600),       166900
     &RET(1600)                                                         167000
      LOGICAL TRANS,TRANS1,SORP,SSTATE                                  167100
      COMMON/TRXY/MB9(72),NMB9,EPS1,TRANS,TRANS1,SORP,SSTATE            167200
```

```
      LOGICAL RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                        167300
      LOGICAL F7P,F11P,F8P,F9P,F6P,PRNT                                 167400
      COMMON/LOG1/RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                    167500
      COMMON/LOG2/F7P,F11P,F8P,F9P,F6P,PRNT                             167600
      CHARACTER*80 TITL                                                 167700
      CHARACTER*4 ZUNIT,TUNIT,CUNX                                      167800
      COMMON/SCHAR/TITL,ZUNIT,TUNIT,CUNX                                167900
      DIMENSION BL(72)                                                  168000
      SAVE BL                                                           168100
C-----------------------------------------------------------------      168200
C                                                                       168300
C   INITIALIZE MASS BALANCE VARIABLES USED FOR                          168400
C   ENTIRE SIMULATION.                                                  168500
C                                                                       168600
      IF(KTIM.EQ.1) THEN                                                168700
      DO 10 I=1,72                                                      168800
      BL(I)=0.                                                          168900
   10 CONTINUE                                                          169000
      END IF                                                            169100
C                                                                       169200
C   INITIALIZE MASS BALANCE VARIABLES USED FOR CURRENT                  169300
C   TIME STEP                                                           169400
C                                                                       169500
      BLTEMP=0                                                          169600
      BL(3)=0.                                                          169700
      BL(6)=0.                                                          169800
      BL(9)=0.                                                          169900
      BL(12)=0.                                                         170000
      BL(27)=0.                                                         170100
      BL(29)=0                                                          170200
      BL(36)=0.                                                         170300
      BL(39)=0.                                                         170400
      BL(42)=0.                                                         170500
      BL(45)=0.                                                         170600
      BL(60)=0.                                                         170700
      BL(68)=0                                                          170800
      BL(62)=0.                                                         170900
      BL(51)=0.                                                         171000
      BL(48)=0.                                                         171100
      DO 20 J=2,NLYY                                                    171200
      DO 20 N=2,NXRR                                                    171300
      IN=NLY*(N-1)+J                                                    171400
      IF(HX(IN).EQ.0.) GO TO 20                                         171500
      JM1=IN-1                                                          171600
      JP1=IN+1                                                          171700
      NM1=IN-NLY                                                        171800
      NP1=IN+NLY                                                        171900
      VOL=DXR(N)*DELZ(J)                                                172000
      IF(RAD)VOL=PI2*RX(N)*DXR(N)*DELZ(J)                               172100
C                                                                       172200
C     SUM CHANGE IN STORAGE                                             172300
C                                                                       172400
      GSF=VOL*(THETA(IN)-THLST(IN))                                     172500
      JJ=JTEX(IN)                                                       172600
      SS=HK(JJ,2)/HK(JJ,3)                                              172700
      GSS=VOL*THETA(IN)*SS                                              172800
      BL(29)=BL(29)+(GSF+GSS*(P(IN)-PXXX(IN)))                          172900
      IF(TRANS) THEN                                                    173000
C                                                                       173100
C   FOR TRANSPORT SUM CHANGE IN STORAGE AND DIFFUSIVE/DISPERSIVE        173200
C   FLUXES                                                              173300
C                                                                       173400
      IF(NCTYP(IN).NE.1) BL(68)=BL(68)+VOL*(                            173500
     *CC(IN)*THETA(IN)*(1+SS*P(IN))-COLD(IN)*THLST(IN)*(1+SS*PXXX(IN)))  173600
      SS=-HT(JJ,4)*(THETA(IN)+THETA(IN)*P(IN)*SS+RET(IN))*DELT          173700
      BL(62)=BL(62)+VOL*SS*CC(IN)                                       173800
      BLTEMP=BLTEMP-RET(IN)*CC(IN)*VOL                                  173900
```

```
      IF(NCTYP(IN).EQ.2) THEN                                          174000
      IF(CS(IN).LT.0) THEN                                             174100
      BL(51)=BL(51)+CS(IN)                                             174200
      ELSE                                                             174300
      BL(48)=BL(48)+CS(IN)                                             174400
      END IF                                                           174500
      END IF                                                           174600
      IF(NCTYP(IN).EQ.1) THEN                                          174700
      IP2=NP1-1                                                        174800
      IM2=NM1+1                                                        174900
      IM3=NM1-1                                                        175000
      IP3=NP1+1                                                        175100
      T5=(DX1(NP1)*(CC(IN)-CC(NP1))-DX2(NP1)*(0.5)*(                   175200
     &CC(JP1)-CC(JM1)+CC(IP3)-CC(IP2)))                                175300
     &+(DX1(IN)*(CC(IN)-CC(NM1))+DX2(IN)*(0.5)*(                       175400
     &CC(JP1)-CC(JM1)+CC(IM2)-CC(IM3)))                                175500
     &+(DZ1(JP1)*(CC(IN)-CC(JP1))-DZ2(JP1)*(0.5)*(                     175600
     &CC(NP1)-CC(NM1)+CC(IP3)-CC(IM2)))                                175700
     &+(DZ1(IN)*(CC(IN)-CC(JM1))+DZ2(IN)*(0.5)*(                       175800
     &CC(NP1)-CC(NM1)+CC(IP2)-CC(IM3)))                                175900
      IF(T5.LT.0) THEN                                                 176000
      BL(51)=BL(51)+T5                                                 176100
      ELSE                                                             176200
      BL(48)=BL(48)+T5                                                 176300
      END IF                                                           176400
      END IF                                                           176500
      END IF                                                           176600
C                                                                      176700
C  FLUX FOR NEUMANN CELLS                                              176800
C                                                                      176900
      IF(NTYP(IN).EQ.2) THEN                                           177000
      IF(QQ(IN).LE.0) THEN                                             177100
      BL(12)=BL(12)+QQ(IN)                                             177200
      IF(TRANS) BL(45)=BL(45)+QQ(IN)*CC(IN)                            177300
      ELSE                                                             177400
      BL(9)=BL(9)+QQ(IN)                                               177500
      IF(TRANS) BL(42)=BL(42)+QQ(IN)*CS(IN)                            177600
      END IF                                                           177700
      ELSE                                                             177800
C                                                                      177900
C   FLUX FOR DIRICHLET CELLS                                           178000
C                                                                      178100
      IF(NTYP(IN).EQ.1) THEN                                           178200
      IF(TRANS) THEN                                                   178300
      QX=QT(IN)                                                        178400
      ELSE                                                             178500
      QX=VSFLX1(IN)                                                    178600
      END IF                                                           178700
      IF(QX.LT.0) THEN                                                 178800
      BL(3)=BL(3)-QX                                                   178900
      IF(TRANS) BL(36)=BL(36)-QX*CS(IN)                                179000
      ELSE                                                             179100
      BL(6)=BL(6)-QX                                                   179200
      IF(TRANS) BL(39)=BL(39)-QX*CC(IN)                                179300
      END IF                                                           179400
      ELSE                                                             179500
C                                                                      179600
C    SUM SOURCES AND SINKS                                             179700
C                                                                      179800
      BL(27)=BL(27)+Q(IN)                                              179900
      IF(TRANS) BL(60)=BL(60)+ETOUT*CC(IN)                             180000
      END IF                                                           180100
      END IF                                                           180200
   20 CONTINUE                                                         180300
C                                                                      180400
C   ACCUMULATE VALUES FOR TOTAL ELAPSED SIMULATION TIME                180500
C                                                                      180600
```

```
      BL(24)=ETOUT                                                      180700
      BL(21)=ETOUT1                                                     180800
      BL(30)=BL(29)/DELT                                                180900
      BL(15)=BL(3)+BL(9)                                                181000
      BL(18)=BL(6)+BL(12)                                               181100
      DO 30 I=2,26,3                                                    181200
      BL(I)=DELT*BL(I+1)                                                181300
   30 CONTINUE                                                          181400
      BL(19)=BL(19)+BL(20)                                              181500
      BL(22)=BL(22)+BL(23)                                              181600
      BL(1)=BL(1)+BL(2)                                                 181700
      BL(4)=BL(4)+BL(5)                                                 181800
      BL(10)=BL(10)+BL(11)                                              181900
      BL(13)=BL(13)+BL(14)                                              182000
      BL(7)=BL(7)+BL(8)                                                 182100
      BL(16)=BL(16)+BL(17)                                              182200
      BL(25)=BL(25)+BL(26)                                              182300
      BL(28)=BL(28)+BL(29)                                              182400
      BL(32)=BL(14)+BL(17)+BL(26)-BL(29)                                182500
      BL(33)=BL(32)/DELT                                                182600
      BL(31)=BL(31)+BL(32)                                              182700
      IF(TRANS) THEN                                                    182800
C                                                                       182900
C   TRANSPORT MASS BALANCE COMPONENTS                                   183000
C                                                                       183100
      BL(67)=BL(67)+BL(68)                                              183200
      BL(69)=BL(68)/DELT                                                183300
      BL(61)=BL(61)+BL(62)                                              183400
      BL(65)=BLTEMP-BL(64)                                              183500
      BL(64)=BLTEMP                                                     183600
      BL(63)=BL(62)/DELT                                                183700
      BL(66)=BL(65)/DELT                                                183800
      BL(54)=BL(36)+BL(42)+BL(48)                                       183900
      DO 40 I=35,59,3                                                   184000
      BL(I)=DELT*BL(I+1)                                                184100
   40 CONTINUE                                                          184200
      BL(49)=BL(49)+BL(50)                                              184300
      BL(46)=BL(46)+BL(47)                                              184400
      BL(57)=BL(39)+BL(45)+BL(51)                                       184500
      BL(58)=BL(58)+BL(59)                                              184600
      BL(34)=BL(34)+BL(35)                                              184700
      BL(37)=BL(37)+BL(38)                                              184800
      BL(43)=BL(43)+BL(44)                                              184900
      BL(52)=BL(52)+BL(53)                                              185000
      BL(40)=BL(40)+BL(41)                                              185100
      BL(55)=BL(55)+BL(56)                                              185200
      BL(71)=BL(53)+BL(56)+BL(59)-BL(68)+BL(62)+BL(65)                  185300
      BL(72)=BL(71)/DELT                                                185400
      BL(70)=BL(70)+BL(71)                                              185500
      END IF                                                            185600
C                                                                       185700
C   WRITE RESULTS TO FILE 9                                             185800
C                                                                       185900
      IF(F9P) WRITE(09,4000) STIM,(BL(MB9(IM)),IM=1,NMB9)               186000
      IF(.NOT.F6P.AND.JPLT.NE.1.AND.JSTOP.NE.1.AND.JFLAG.NE.1) GO TO 50 186100
C                                                                       186200
C    WRITE RESULTS OF MASS BALANCE TO FILE 6                            186300
C                                                                       186400
      WRITE (06,4010) KTIM,KP,STIM,TUNIT,ZUNIT,ZUNIT,ZUNIT,TUNIT,(BL(M), 186500
     *M=1,12)                                                           186600
      WRITE(06,4020) (BL(M),M=13,33)                                    186700
      IF(TRANS) WRITE(06,4030) CUNX,CUNX,CUNX,TUNIT,(BL(M),M=34,72)     186800
      WRITE(06,4040)                                                    186900
   50 CONTINUE                                                          187000
      RETURN                                                            187100
 4000 FORMAT(11(1PE11.3))                                               187200
 4010 FORMAT(21X,10(1H-),1X,'MASS BALANCE SUMMARY FOR TIME STEP',       187300
```

```
     & I4,1X,10(1H-)/25X,'PUMPING PERIOD NUMBER ',I4/25X,                  187400
     &'TOTAL ELAPSED SIMULATION TIME = ',1PE10.3,1X,A4//2X,128(1H+)/       187500
     & 2X,'+',126X,'+'/                                                    187600
     &2X,'+',90X,' TOTAL THIS',11X,'RATE THIS',5X,'+'/2X,'+',              187700
     &33X,'VOLUMETRIC FLOW BALANCE',                                       187800
     &18X,'TOTAL   ',9X,'TIME STEP',11X,'TIME STEP',6X,'+'/                187900
     &2X,'+',72X,A4,'**3',13X,A4,'**3',11X,A4,'**3/',A4,4X,'+'/            188000
     &2X,'+',4X,'FLUX INTO DOMAIN ACROSS SPECIFIED PRESSURE HEAD',         188100
     &1X,'BOUNDARIES --  ',2(1PE15.5,5X),1PE15.5,4X,'+'/                   188200
     &2X,'+',2X,'FLUX OUT OF DOMAIN ACROSS SPECIFIED PRESSURE HEAD',       188300
     &1X,'BOUNDARIES --  ',2(1PE15.5,5X),1PE15.5,4X,'+'/                   188400
     &2X,'+',13X,'FLUX INTO DOMAIN ACROSS SPECIFIED FLUX BOUNDARIES',      188500
     &1X,'--  ',2(1PE15.5,5X),1PE15.5,4X,'+'/                              188600
     &2X,'+',11X,'FLUX OUT OF DOMAIN ACROSS SPECIFIED FLUX',               188700
     &1X,'BOUNDARIES --  ',2(1PE15.5,5X),1PE15.5,4X,'+')                   188800
 4020 FORMAT(1H ,1X,'+',40X,'TOTAL FLUX INTO DOMAIN --  ',2(1PE15.5,5X),   188900
     & 1PE15.5,4X,'+'/2X,'+',38X,'TOTAL FLUX OUT OF DOMAIN --  ',          189000
     &2(1PE15.5,5X),1PE15.5,4X,'+'/                                        189100
     &2X,'+',51X,'EVAPORATION --  ',2(1PE15.5,5X),1PE15.5,4X,'+'/          189200
     &2X,'+',49X,'TRANSPIRATION --  ',2(1PE15.5,5X),1PE15.5,4X,'+'/        189300
     &2X,'+',38X,'TOTAL EVAPOTRANSPIRATION',                               189400
     &1X,'--  ',2(1PE15.5,5X),1PE15.5,4X,'+'/                              189500
     &2X,'+',30X,'CHANGE IN FLUID STORED IN DOMAIN --  ',                  189600
     &2(1PE15.5,5X),1PE15.5,4X,'+'/2X,'+',42X,'FLUID VOLUME BALANCE'       189700
     &,1X,'--  ',2(1PE15.5,5X),1PE15.5,4X,'+'/2X,'+',126X,'+')             189800
 4030 FORMAT(2X,'+'126X,'+',/,2X,'+',35X,'SOLUTE MASS BALANCE',            189900
     &72X,'+',/,2X,'+',74X,A4,16X,A4,14X,A4,'/',A4,5X,'+',/,               190000
     &2X,'+',4X,'FLUX INTO DOMAIN ACROSS SPECIFIED PRESSURE HEAD',         190100
     &1X,'BOUNDARIES --  ',2(1PE15.5,5X),1PE15.5,4X,'+'/                   190200
     &2X,'+',2X,'FLUX OUT OF DOMAIN ACROSS SPECIFIED PRESSURE HEAD',       190300
     &1X,'BOUNDARIES --  ',2(1PE15.5,5X),1PE15.5,4X,'+'/                   190400
     &2X,'+',13X,'FLUX INTO DOMAIN ACROSS SPECIFIED FLUX BOUNDARIES',      190500
     &1X,'--  ',2(1PE15.5,5X),1PE15.5,4X,'+'/                              190600
     &2X,'+',11X,'FLUX OUT OF DOMAIN ACROSS SPECIFIED FLUX',               190700
     &1X,'BOUNDARIES --  ',2(1PE15.5,5X),1PE15.5,4X,'+'/,                  190800
     &2X,'+',25X,'DIFFUSIVE/DISPERSIVE FLUX INTO DOMAIN --  ',             190900
     &2(1PE15.5,5X),1PE15.5,4X,'+'/2X,                                     191000
     &'+',23X,'DIFFUSIVE/DISPERSIVE FLUX OUT OF DOMAIN --  ',              191100
     &2(1PE15.5,5X),1PE15.5,4X,'+'/,                                       191200
     &1H ,1X,'+',40X,'TOTAL FLUX INTO DOMAIN --  ',2(1PE15.5,5X),          191300
     & 1PE15.5,4X,'+'/2X,'+',38X,'TOTAL FLUX OUT OF DOMAIN --  ',          191400
     &2(1PE15.5,5X),1PE15.5,4X,'+'/                                        191500
     &2X,'+',38X,'TOTAL EVAPOTRANSPIRATION',                               191600
     &1X,'--  ',2(1PE15.5,5X),1PE15.5,4X,'+'/                              191700
     &2X,'+',45X,'FIRST ORDER DECAY --  ',2(1PE15.5,5X),                   191800
     &1PE15.5,4X,'+'/                                                      191900
     &2X,'+',39X,'ADSORPTION/ION EXCHANGE --  ',2(1PE15.5,5X),             192000
     &1PE15.5,4X,'+'/                                                      192100
     &2X,'+',29X,'CHANGE IN SOLUTE STORED IN DOMAIN --  ',                 192200
     &2(1PE15.5,5X),1PE15.5,4X,'+'/2X,'+',43X,'SOLUTE MASS BALANCE'        192300
     &,1X,'--  ',2(1PE15.5,5X),1PE15.5,4X,'+'/2X,'+',126X,'+')             192400
 4040 FORMAT( 2X,128(1H+))                                                 192500
      END                                                                  192600
      DOUBLE PRECISION FUNCTION VSFLX1(IN)                                 192700
C******                                                                    192800
CVSFLX1                                                                    192900
C******                                                                    193000
C   PURPOSE: TO COMPUTE INTERCELL MASS FLUX RATES FOR DIRICHLET            193100
C   BOUNDARY NODES                                                         193200
C ---------------------------------------------------------------          193300
C                                                                          193400
C   SPECIFICATIONS FOR ARRAYS AND SCALARS                                  193500
C                                                                          193600
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                  193700
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2                 193800
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                                193900
      COMMON/KCON/HX(1600),NTYP(1600)                                      194000
```

```
      COMMON/PRESS/P(1600),PXXX(1600),CS1,CS2                          194100
      COMMON/DISCH/Q(1600),QQ(1600),ETOUT,ETOUT1                       194200
      COMMON/HCON/HCND(1600),HKLL(1600),HKTT(1600)                     194300
      COMMON/EQUAT/A(1600),B(1600),C(1600),D(1600),E(1600),RHS(1600),  194400
     &XI(1600)                                                         194500
      COMMON/WGT/WUS,WDS                                               194600
      LOGICAL RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                       194700
      COMMON/LOG1/RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                   194800
C-----------------------------------------------------------------------194900
C                                                                      195000
C                                                                      195100
C   COMPUTE FLUXES ON ALL FOUR SIDES OF EACH CONSTANT HEAD NODE        195200
C                                                                      195300
      JM1=IN-1                                                         195400
      JP1=IN+1                                                         195500
      NP1=IN+NLY                                                       195600
      NM1=IN-NLY                                                       195700
C                                                                      195800
C   COMPUTE A,B,C,D                                                    195900
C                                                                      196000
      IF(WUS.EQ.0.) THEN                                               196100
      A(IN)=HKLL(IN)*DSQRT(HCND(NM1)*HCND(IN))                         196200
      B(IN)=HKTT(IN)*DSQRT(HCND(JM1)*HCND(IN))                         196300
      C(IN)=HKLL(NP1)*DSQRT(HCND(NP1)*HCND(IN))                        196400
      D(IN)=HKTT(JP1)*DSQRT(HCND(JP1)*HCND(IN))                        196500
      ELSE                                                             196600
      IF(P(NM1).GT.P(IN).AND.HX(NM1).NE.0.) THEN                       196700
      ALA=WDS                                                          196800
      BTA=WUS                                                          196900
      ELSE                                                             197000
      ALA=WUS                                                          197100
      BTA=WDS                                                          197200
      END IF                                                           197300
      IF(P(JM1).GT.P(IN).AND.HX(JM1).NE.0.) THEN                       197400
      ALB=WDS                                                          197500
      BTB=WUS                                                          197600
      ELSE                                                             197700
      ALB=WUS                                                          197800
      BTB=WDS                                                          197900
      END IF                                                           198000
      IF(P(NP1).GT.P(IN).AND.HX(NP1).NE.0.) THEN                       198100
      ALC=WDS                                                          198200
      BTC=WUS                                                          198300
      ELSE                                                             198400
      ALC=WUS                                                          198500
      BTC=WDS                                                          198600
      END IF                                                           198700
      IF(P(JP1).GT.P(IN).AND.HX(JP1).NE.0.) THEN                       198800
      ALD=WDS                                                          198900
      BTD=WUS                                                          199000
      ELSE                                                             199100
      ALD=WUS                                                          199200
      BTD=WDS                                                          199300
      END IF                                                           199400
C                                                                      199500
C   DETERMINE FLUXES                                                   199600
C                                                                      199700
      A(IN)=(ALA*HCND(NM1)+BTA*HCND(IN))*HKLL(IN)                      199800
      B(IN)=(ALB*HCND(JM1)+BTB*HCND(IN))*HKTT(IN)                      199900
      C(IN)=(ALC*HCND(NP1)+BTC*HCND(IN))*HKLL(NP1)                     200000
      D(IN)=(ALD*HCND(JP1)+BTD*HCND(IN))*HKTT(JP1)                     200100
      END IF                                                           200200
   10 QL=-A(IN)*(P(IN)-P(NM1))                                         200300
      QA=-B(IN)*(P(IN)-P(JM1))                                         200400
      QR=-C(IN)*(P(IN)-P(NP1))                                         200500
      QB=-D(IN)*(P(IN)-P(JP1))                                         200600
C                                                                      200700
```

```
C     COMPUTE NET FLUX IN (+) OR OUT (-)                                200800
C                                                                       200900
      VSFLX1=QL+QR+QA+QB                                                201000
      RETURN                                                            201100
      END                                                               201200
      SUBROUTINE VSOUTP                                                 201300
C******                                                                 201400
CVSOUTP                                                                 201500
C******                                                                 201600
C                                                                       201700
C   PURPOSE: TO OUTPUT RESULTS AFTER EACH TIME STEP.                    201800
C                                                                       201900
C-----------------------------------------------------------------------202000
C                                                                       202100
C        SPECIFICATIONS FOR ARRAYS AND SCALARS                          202200
C                                                                       202300
      IMPLICIT DOUBLE PRECISION(A-H,P-Z)                                202400
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2              202500
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                             202600
      COMMON/KCON/HX(1600),NTYP(1600)                                   202700
      COMMON/RPROP/HK(10,100),HT(10,20),ANIZ(10)                        202800
      COMMON/MPROP/THETA(1600),THLST(1600)                              202900
      COMMON/PRESS/P(1600),PXXX(1600),CS1,CS2                           203000
      COMMON/DISCH/Q(1600),QQ(1600),ETOUT,ETOUT1                        203100
      COMMON/JTXX/JTEX(1600)                                            203200
      COMMON/DUMM/DUM(1600)                                             203300
      COMMON/PLOTT/PLTIM(50),IJOBS(50),JPLT,NPLT,NOBS                   203400
      COMMON/SCON/DHMX(201),DELT,HMAX,TMAX,EPS,NUMT,ITMAX,MINIT,ITEST   203500
      COMMON/SCN1/TMPX,TMLT,DLTMX,DLTMIN,TRED                           203600
      COMMON/TCON/STIM,DSMAX,KTIM,NIT,NIT1,KP                           203700
      COMMON/JCON/JSTOP,JFLAG                                           203800
      COMMON/TRXX/DX1(1600),DX2(1600),DZ1(1600),DZ2(1600),VX(1600),     203900
     &VZ(1600),CC(1600),COLD(1600),CS(1600),QT(1600),NCTYP(1600),       204000
     &RET(1600)                                                         204100
      LOGICAL TRANS,TRANS1,SORP,SSTATE                                  204200
      COMMON/TRXY/MB9(72),NMB9,EPS1,TRANS,TRANS1,SORP,SSTATE            204300
      LOGICAL F7P,F11P,F8P,F9P,F6P,PRNT                                 204400
      LOGICAL THPT,SPNT,PPNT,HPNT,VPNT                                  204500
      COMMON/LOG2/F7P,F11P,F8P,F9P,F6P,PRNT                             204600
      COMMON/LOG4/THPT,SPNT,PPNT,HPNT,VPNT                              204700
      CHARACTER*80 TITL                                                 204800
      CHARACTER*4 ZUNIT,TUNIT,CUNX                                      204900
      COMMON/SCHAR/TITL,ZUNIT,TUNIT,CUNX                                205000
C                                                                       205100
C-----------------------------------------------------------------      205200
C                                                                       205300
C   OUTPUT RESULTS TO FILE 11 AT EACH TIME STEP                         205400
C                                                                       205500
      IF(F11P) THEN                                                     205600
      DO 10 J=1,NOBS                                                    205700
      N=IJOBS(J)                                                        205800
      I=N/NLY+1                                                         205900
      J1=MOD(N,NLY)                                                     206000
      IF(HX(N).NE.0.) THEN                                              206100
      PPR=HK(JTEX(N),3)                                                 206200
      IF(PPR.EQ.0.)PPR=1.                                               206300
      SAT=THETA(N)/PPR                                                  206400
      IF(CS1.EQ.1.) THEN                                                206500
      Z1=DZZ(J1)                                                        206600
      ELSE                                                              206700
      Z1=DZZ(J1)*CS1+(RX(I))*CS2                                        206800
      END IF                                                            206900
      PHD=P(N)+Z1                                                       207000
      IF(TRANS) THEN                                                    207100
      WRITE(11,4020) STIM,RX(I),DZZ(J1),P(N),PHD,THETA(N),SAT,CC(N)     207200
      ELSE                                                              207300
      WRITE (11,4020) STIM,RX(I),DZZ(J1),P(N),PHD,THETA(N),SAT          207400
```

```
      END IF                                                            207500
      END IF                                                            207600
   10 CONTINUE                                                          207700
      END IF                                                            207800
      IF(KTIM.EQ.0)  GO TO 20                                           207900
C                                                                       208000
C    WRITE TIME STEP HEADER TO FILE 6                                   208100
C                                                                       208200
C    WRITE MAXIMUM HEAD CHANGE EACH TIME STEP TO FILE 7                 208300
C                                                                       208400
      IF(F7P) THEN                                                      208500
      WRITE(07,4040) KTIM,STIM,NIT,NIT1                                 208600
      WRITE(07,4030) (DHMX(M2),M2=1,NIT)                                208700
      END IF                                                            208800
      WRITE(06,4040) KTIM,STIM,NIT,NIT1                                 208900
      IF(JSTOP.EQ.1.OR.JPLT.EQ.1) GO TO 20                              209000
      IF(.NOT.PRNT.AND.JFLAG.EQ.0) RETURN                               209100
   20 WRITE (6,4050) TITL,STIM,TUNIT,KTIM                               209200
C                                                                       209300
C    PRINT SOLUTION FOR CURRENT TIME STEP                               209400
C                                                                       209500
      IF(JPLT.EQ.1) THEN                                                209600
C                                                                       209700
C   WRITE PRESSURE HEADS TO FILE 8 AT OBSERVATION TIMES.                209800
C                                                                       209900
      WRITE (8,4000) STIM,TUNIT                                         210000
      DO 40 J=1,NLY                                                     210100
      DO 30 N=1,NXR                                                     210200
      IN=NLY*(N-1)+J                                                    210300
      IF(CS1.EQ.1.) THEN                                                210400
      Z1=DZZ(J)                                                         210500
      ELSE                                                              210600
      Z1=DZZ(J)*CS1+(RX(N))*CS2                                         210700
      END IF                                                            210800
   30 DUM(IN)=P(IN)+Z1                                                  210900
   40 WRITE(8,4010) (DUM(N),N=J,NNODES-NLY+J,NLY)                       211000
C                                                                       211100
C  WRITE CONCENTRATIONS TO FILE 8                                       211200
C                                                                       211300
      IF(TRANS) THEN                                                    211400
      DO 50 J=1,NLY                                                     211500
      WRITE(08,4010) (CC(N),N=J,NNODES-NLY+J,NLY)                       211600
   50 CONTINUE                                                          211700
      END IF                                                            211800
      END IF                                                            211900
C                                                                       212000
C    PRINT TOTAL HEADS                                                  212100
C                                                                       212200
      IF(HPNT) THEN                                                     212300
      WRITE (6,4060)                                                    212400
      CALL VSOUT(1,P)                                                   212500
      END IF                                                            212600
C                                                                       212700
C  PRINT PRESSURE HEADS                                                 212800
C                                                                       212900
      IF(PPNT) THEN                                                     213000
      IF(JPLT.NE.1) THEN                                                213100
      DO 60 J=2,NLYY                                                    213200
      DO 60 N=2,NXRR                                                    213300
      IN=NLY*(N-1)+J                                                    213400
      IF(CS1.EQ.1.) THEN                                                213500
      Z1=DZZ(J)                                                         213600
      ELSE                                                              213700
      Z1=DZZ(J)*CS1+(RX(N))*CS2                                         213800
      END IF                                                            213900
      DUM(IN)=P(IN)+Z1                                                  214000
      IF(HX(IN).EQ.0.)DUM(IN)=0.                                        214100
```

```
   60 CONTINUE                                                          214200
      END IF                                                            214300
      WRITE (6,4070)                                                    214400
      CALL VSOUT(1,DUM)                                                 214500
      END IF                                                            214600
C                                                                       214700
C  PRINT SATURATIONS                                                    214800
C                                                                       214900
      IF(SPNT) THEN                                                     215000
      DO 70 J=2,NLYY                                                    215100
      DO 70 N=2,NXRR                                                    215200
      IN=NLY*(N-1)+J                                                    215300
      TTX=HK(JTEX(IN),3)                                                215400
      IF(TTX.EQ.0.) THEN                                                215500
      DUM(IN)=0.                                                        215600
      ELSE                                                              215700
      DUM(IN)=THETA(IN)/TTX                                             215800
      END IF                                                            215900
   70 CONTINUE                                                          216000
      WRITE (6,4080)                                                    216100
      CALL VSOUT(2,DUM)                                                 216200
      END IF                                                            216300
C                                                                       216400
C  PRINT MOISTURE CONTENTS                                              216500
C                                                                       216600
      IF(THPT) THEN                                                     216700
      WRITE (6,4090)                                                    216800
      CALL VSOUT(2,THETA)                                               216900
      END IF                                                            217000
C                                                                       217100
C  PRINT VELOCITIES                                                     217200
C                                                                       217300
      IF(VPNT.AND.KTIM.GT.0) THEN                                       217400
      WRITE(06,4100)                                                    217500
      CALL VSOUT(1,VX)                                                  217600
      WRITE(06,4110)                                                    217700
      CALL VSOUT(1,VZ)                                                  217800
      END IF                                                            217900
C                                                                       218000
C  PRINT CONCENTRATIONS                                                 218100
C                                                                       218200
      IF(TRANS) THEN                                                    218300
      WRITE(6,4120)                                                     218400
      CALL VSOUT(1,CC)                                                  218500
      END IF                                                            218600
      CONTINUE                                                          218700
      RETURN                                                            218800
 4000 FORMAT(/,8H TIME = ,E14.4,1X,A4/)                                 218900
 4010 FORMAT(8(1PE10.3))                                                219000
 4020 FORMAT(8(1PE12.3))                                                219100
 4030 FORMAT(7E11.4)                                                    219200
 4040 FORMAT(' TIME STEP ',I5,'  TIME = ',E12.4,'  NIT = ',I3,          219300
     &'  NIT1 = ',I3)                                                   219400
 4050 FORMAT(6X,A80/5X,20HTOTAL ELAPSED TIME =,1PE12.3,1X,A4/5X,        219500
     &10HTIME STEP ,I5,//)                                              219600
 4060 FORMAT(1H ,50X,10HTOTAL HEAD)                                     219700
 4070 FORMAT(1H ,50X,13HPRESSURE HEAD)                                  219800
 4080 FORMAT(1H ,50X,10HSATURATION)                                     219900
 4090 FORMAT(1H ,50X,16HMOISTURE CONTENT)                               220000
 4100 FORMAT(51X,'X-VELOCITY')                                          220100
 4110 FORMAT(51X,'Z-VELOCITY')                                          220200
 4120 FORMAT(51X,'CONCENTRATION')                                       220300
      END                                                               220400
      SUBROUTINE VSOUT(IV,VPRNT)                                        220500
C*****                                                                  220600
CVSOUT                                                                  220700
C*****                                                                  220800
```

```
C                                                                       220900
C    PURPOSE: TO PRINT TWO DIMENSIONAL ARRAYS                           221000
C                                                                       221100
C                                                                       221200
C-----------------------------------------------------------------------221300
C                                                                       221400
C   SPECIFICATIONS FOR ARRAYS AND SCALARS                               221500
C                                                                       221600
C                                                                       221700
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               221800
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2              221900
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                             222000
      COMMON/KCON/HX(1600),NTYP(1600)                                   222100
      COMMON/DUMM/DUM(1600)                                             222200
      COMMON/PLOTT/PLTIM(50),IJOBS(50),JPLT,NPLT,NOBS                   222300
      LOGICAL F7P,F11P,F8P,F9P,F6P,PRNT                                 222400
      COMMON/LOG2/F7P,F11P,F8P,F9P,F6P,PRNT                             222500
      CHARACTER*80 TITL                                                 222600
      CHARACTER*4 ZUNIT,TUNIT,CUNX                                      222700
      COMMON/SCHAR/TITL,ZUNIT,TUNIT,CUNX                                222800
      DIMENSION VPRNT(1),DUM1(600)                                      222900
C                                                                       223000
C-----------------------------------------------------------------------223100
C                                                                       223200
      WRITE (06,4000) ZUNIT,ZUNIT                                       223300
      WRITE (06,4010) (RX(K),K=2,NXRR)                                  223400
      DO 30 J=2,NLYY                                                    223500
      DO 10 N=2,NXRR                                                    223600
      IN=NLY*(N-1)+J                                                    223700
      DUM1(N)=VPRNT(IN)                                                 223800
      IF(HX(IN).EQ.0. ) DUM1(N)=0.                                      223900
   10 CONTINUE                                                          224000
      IF(IV.GT.1) GO TO 20                                              224100
      WRITE (06,4020) DZZ(J),(DUM1(N),N=2,NXRR)                         224200
      GO TO 30                                                          224300
   20 WRITE (06,4030) DZZ(J),(DUM1(N),N=2,NXRR)                         224400
   30 CONTINUE                                                          224500
      RETURN                                                            224600
 4000 FORMAT(1H ,1X,5HZ, IN/2X,A4,20X,20HX OR R DISTANCE, IN ,A4)       224700
 4010 FORMAT(1H ,8X,13(F9.2)/(9X,13(F9.2)))                             224800
 4020 FORMAT(1X,F8.2,13(1PE9.2)/(9X,13(1PE9.2)))                        224900
 4030 FORMAT(1X,F8.2,13F9.3/(9X,13F9.3))                                225000
      END                                                               225100
      SUBROUTINE VSPOND(IFET,IFET1,IFET2)                               225200
C******                                                                 225300
CVSPOND                                                                 225400
C******                                                                 225500
C                                                                       225600
C  UPDATED 10-88                                                        225700
C                                                                       225800
C    PURPOSE:  TO DETERMINE IF PONDING OR UNPONDING HAS OCCURRED, AND   225900
C              IF SO TO CHANGE BOUNDARY CONDITIONS AT THOSE NODES FROM  226000
C              NEUMAN TO DIRICHLET OR VICE VERSA                        226100
C                                                                       226200
C -------------------------------------------------------------         226300
C                                                                       226400
C   SPECIFICATIONS FOR ARRAYS AND SCALARS                               226500
C                                                                       226600
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               226700
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2              226800
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                             226900
      COMMON/KCON/HX(1600),NTYP(1600)                                   227000
      COMMON/PRESS/P(1600),PXXX(1600),CS1,CS2                           227100
      COMMON/DISCH/Q(1600),QQ(1600),ETOUT,ETOUT1                        227200
      COMMON/EQUAT/A(1600),B(1600),C(1600),D(1600),E(1600),RHS(1600),   227300
     &XI(1600)                                                          227400
      COMMON/PND/POND                                                   227500
```

```
      COMMON/TCON/STIM,DSMAX,KTIM,NIT,NIT1,KP                          227600
C                                                                      227700
C-----------------------------------------------------------------     227800
C                                                                      227900
C   IFET1 INDICATES WHETHER THERE ARE ANY NEUMAN BOUNDARIES REMAINING  228000
C   IFET2 INDICATES WHETHER ANY SPECIFIC FLUX NODES HAVE BEEN CONVERTED 228100
C     TO SPECIFIED HEAD NODES.  BECAUSE OF THE CAPILLARY BARRIER       228200
C     EFFECT, THESE NODES MAY NEED TO REVERT TO SPECIFIED FLUX NODES.  228300
C   IFET INDICATES WHETHER PONDING OCCURRED OR DISAPPEARED             228400
C                                                                      228500
      IF(IFET1.EQ.0 .AND. IFET2 .EQ. 0) RETURN                         228600
      IFET=0                                                           228700
      IFET1=0                                                          228800
      IFET2=0                                                          228900
      IF(CS1.EQ.1.) THEN                                               229000
      DZ1=DZZ(2)                                                       229100
      ELSE                                                             229200
      IF(CS2.LT.0) THEN                                                229300
      DZ1=DZZ(2)*CS1+RX(NXRR)*CS2                                      229400
      ELSE                                                             229500
      DZ1=DZZ(2)*CS1+RX(2)*CS2                                         229600
      END IF                                                           229700
      END IF                                                           229800
      DO 20 I=2,NXRR                                                   229900
      DO 10 J=2,NLYY                                                   230000
      IN=NLY*(I-1)+J                                                   230100
      IF(HX(IN).NE.0.) THEN                                            230200
      IF(NTYP(IN).EQ.2.AND.QQ(IN).GT.0.) THEN                          230300
      IFET1=1                                                          230400
      IF(CS1.EQ.1.) THEN                                               230500
      Z1=DZZ(J)                                                        230600
      ELSE                                                             230700
      Z1=DZZ(J)*CS1+RX(I)*CS2                                          230800
      END IF                                                           230900
      IF(POND.GE.0.) THEN                                              231000
C                                                                      231100
C    DZ2 IS MAXIMUM ALLOWABLE TOTAL HEAD                               231200
C                                                                      231300
      DZ2=POND-Z1                                                      231400
      ELSE                                                             231500
      DZ2=-DMIN1(Z1,DZ1-POND)                                          231600
      END IF                                                           231700
      IF(P(IN).GT.DZ2) THEN                                            231800
C                                                                      231900
C    IF COMPUTED HEAD EXCEEDS MAXIMUM THEN SET P=DZ2                   232000
C    AND CHANGE BOUNDARY TYPE TO CONSTANT HEAD                         232100
C                                                                      232200
      P(IN)=DZ2                                                        232300
      NTYP(IN)=1                                                       232400
      IFET=1                                                           232500
      IFET2=1                                                          232600
      WRITE(6,4000) J,I,KTIM,NIT                                       232700
      END IF                                                           232800
      ELSE                                                             232900
      IF(NTYP(IN).EQ.1.AND.QQ(IN).GT.0.) THEN                          233000
      IFET2=1                                                          233100
      JP1=IN+1                                                         233200
      IM1=IN+NLY                                                       233300
      IP1=IN-NLY                                                       233400
      TEST=(P(IN)-P(JP1))*D(IN)                                        233500
      IF(HX(IM1).NE.0) TEST=TEST+(P(IN)-P(IM1))*C(IN)                  233600
      IF(HX(IP1).NE.0)TEST=TEST+(P(IN)-P(IP1))*A(IN)                   233700
      TEST=TEST/QQ(IN)                                                 233800
      IF (TEST .GE. 1.01)THEN                                          233900
C                                                                      234000
C  IF FLUX FROM THE CONVERTED NODE IS GREATER THAN THE SPECIFIED       234100
C  FLUX RATE, THE NODE IS RECONVERTED TO A SPECIFIED FLUX NODE.        234200
```

```
C                                                                       234300
      NTYP(IN)=2                                                        234400
      IFET=1                                                            234500
      IFET1=1                                                           234600
      WRITE(06,4010)J,I,KTIM,NIT                                        234700
      END IF                                                            234800
      END IF                                                            234900
      END IF                                                            235000
      GO TO 20                                                          235100
      END IF                                                            235200
   10 CONTINUE                                                          235300
   20 CONTINUE                                                          235400
      RETURN                                                            235500
 4000 FORMAT(//,6X,17H PONDING AT NODE ,2I4,17H DURING TIME STEP,       235600
     &I4,'  ITERATION ',I4)                                             235700
 4010 FORMAT(//,6X,' PONDING ENDED AT NODE ',2I4,                       235800
     &' DURING TIME STEP ',I4,'  ITERATION ',I4)                        235900
      END                                                               236000
      SUBROUTINE VSSFAC                                                 236100
C******                                                                 236200
CVSSFAC                                                                 236300
C******                                                                 236400
C                                                                       236500
C  REVISED 10-88                                                        236600
C                                                                       236700
C    PURPOSE:  TO COMPUTE POSITION OF SEEPAGE FACE BOUNDARIES           236800
C                                                                       236900
C     HEIGHT OF SEEPAGE FACE IS LOWERED IF THERE IS FLUX INTO SYSTEM    237000
C     THRU FACE.                                                        237100
C     HEIGHT IS RAISED IF PRESSURE HEADS ARE POSITIVE ABOVE FACE.       237200
C                                                                       237300
C ---------------------------------------------------------------       237400
C                                                                       237500
C   SPECIFICATIONS FOR ARRAYS AND SCALARS                               237600
C                                                                       237700
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               237800
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2              237900
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                             238000
      COMMON/KCON/HX(1600),NTYP(1600)                                   238100
      COMMON/PRESS/P(1600),PXXX(1600),CS1,CS2                           238200
      COMMON/HCON/HCND(1600),HKLL(1600),HKTT(1600)                      238300
      COMMON/SPFC/JSPX(3,25,8),NFC(8),JLAST(8),NFCS                     238400
      COMMON/TCON/STIM,DSMAX,KTIM,NIT,NIT1,KP                           238500
C                                                                       238600
C-----------------------------------------------------------------      238700
C                                                                       238800
      DO 90 K=1,NFCS                                                    238900
      NFX=NFC(K)                                                        239000
      JFST=0                                                            239100
      JLST=JLAST(K)                                                     239200
C                                                                       239300
C   CHECK FOR POSITIVE PRESSURES ABOVE SEEPAGE FACE                     239400
C                                                                       239500
      DO 10 J=NFX,1,-1                                                  239600
      IN=JSPX(1,J,K)                                                    239700
      JJ=JSPX(2,J,K)                                                    239800
      NN=JSPX(3,J,K)                                                    239900
      IF(CS1.EQ.1) THEN                                                 240000
      Z1=DZZ(JJ)                                                        240100
      ELSE                                                              240200
      Z1=DZZ(JJ)*CS1+RX(NN)*CS2                                         240300
      END IF                                                            240400
      PTMP=P(IN)+Z1                                                     240500
      IF(PTMP.LT.0.) GO TO 10                                           240600
      JFST=J                                                            240700
      GO TO 20                                                          240800
   10 CONTINUE                                                          240900
```

```
   20 CONTINUE                                                          241000
C                                                                       241100
C   CHECK FOR FLOW INTO DOMAIN THROUGH SEEPAGE FACE                     241200
C                                                                       241300
      IF(JFST.GT.JLST) GO TO 60                                         241400
      DO 40 I=JLST,1,-1                                                 241500
      IN=JSPX(1,I,K)                                                    241600
      IM1=IN-NLY                                                        241700
      JM1=IN-1                                                          241800
      IP1=IN+NLY                                                        241900
      JP1=IN+1                                                          242000
      IF(HX(IM1).EQ.0) THEN                                             242100
      IF(HX(IP1).NE.0.AND.P(IP1).LT.P(IN)) GO TO 30                     242200
      END IF                                                            242300
      IF(HX(JM1).EQ.0) THEN                                             242400
      IF(HX(JP1).NE.0.AND.P(JP1).LT.P(IN)) GO TO 30                     242500
      END IF                                                            242600
      IF(HX(IP1).EQ.0) THEN                                             242700
      IF(HX(IM1).NE.0.AND.P(IM1).LT.P(IN)) GO TO 30                     242800
      END IF                                                            242900
      IF(HX(JP1).EQ.0) THEN                                             243000
      IF(HX(JM1).NE.0.AND.P(JM1).LT.P(IN)) GO TO 30                     243100
      END IF                                                            243200
      GO TO 50                                                          243300
   30 NTYP(IN)=3                                                        243400
   40 CONTINUE                                                          243500
      I=0                                                               243600
   50 IF(I.EQ.JLST) GO TO 60                                            243700
C                                                                       243800
C   RESET SEEPAGE FACE HEIGHT AND BOUNDARIES                            243900
C                                                                       244000
      JLAST(K)=I                                                        244100
      GO TO 80                                                          244200
   60 IF(JFST.EQ.JLST) GO TO 80                                         244300
      DO 70 I=1,JFST                                                    244400
      IN=JSPX(1,I,K)                                                    244500
      JJ=JSPX(2,I,K)                                                    244600
      NN=JSPX(3,I,K)                                                    244700
      IF(CS1.EQ.1) THEN                                                 244800
      Z1=DZZ(JJ)                                                        244900
      ELSE                                                              245000
      Z1=DZZ(JJ)*CS1+RX(NN)*CS2                                         245100
      END IF                                                            245200
      NTYP(IN)=1                                                        245300
      P(IN)=-Z1                                                         245400
   70 CONTINUE                                                          245500
      JLAST(K)=JFST                                                     245600
   80 CONTINUE                                                          245700
   90 CONTINUE                                                          245800
      END                                                               245900
      SUBROUTINE VSEVAP                                                 246000
C******                                                                 246100
CVSEVAP                                                                 246200
C******                                                                 246300
C                                                                       246400
C  PURPOSE: TO COMPUTE SURFACE EVAPORATION RATES                        246500
C                                                                       246600
C                                                                       246700
C-----------------------------------------------------------------      246800
C                                                                       246900
C   SPECIFICATIONS FOR ARRAYS AND SCALARS                               247000
C                                                                       247100
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               247200
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2              247300
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                             247400
      COMMON/KCON/HX(1600),NTYP(1600)                                   247500
      COMMON/HCON/HCND(1600),HKLL(1600),HKTT(1600)                      247600
```

```
      COMMON/PRESS/P(1600),PXXX(1600),CS1,CS2                          247700
      COMMON/DISCH/Q(1600),QQ(1600),ETOUT,ETOUT1                       247800
      COMMON/PTET/DPTH(1600),RT(1600),RDC(6,25),ETCYC,                 247900
     &PEVAL(25),PTVAL(25),PET,PEV,HROOT,HA,SRES,RTDPTH,                248000
     &RTBOT,RTTOP,NPV                                                  248100
      LOGICAL RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                       248200
      COMMON/LOG1/RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                   248300
C                                                                      248400
C-----------------------------------------------------------------     248500
C                                                                      248600
      ETOUT1=0                                                         248700
      IF(SRES.EQ.0) RETURN                                             248800
      DO 10 J=2,NLYY                                                   248900
      DO 10 N=2,NXRR                                                   249000
      IN=NLY*(N-1)+J                                                   249100
      IF(NTYP(IN).EQ.5) THEN                                           249200
C                                                                      249300
C    COMPUTE TEMPORARY EVAP RATE, CHECK AGAINST MAX AND                249400
C    CORRECT IF NECESSARY                                              249500
C                                                                      249600
      AREA=DXR(N)                                                      249700
      IF(RAD)AREA=PI2*RX(N)*DXR(N)                                     249800
      PETT=PEV*AREA                                                    249900
      IF(CS1.EQ.1.) THEN                                               250000
      Z1=DZZ(J)                                                        250100
      ELSE                                                             250200
      Z1=DZZ(J)*CS1+(RX(N))*CS2                                        250300
      END IF                                                           250400
      PTMP=P(IN)+Z1                                                    250500
      HKX=HCND(IN)*HX(IN)                                              250600
      EV=HKX*SRES*(HA-PTMP)*AREA                                       250700
      IF(EV.GT.0.) EV=0.                                               250800
      IF(EV.GT.PETT) THEN                                              250900
      Q(IN)=EV                                                         251000
      ELSE                                                             251100
      Q(IN)=PETT                                                       251200
      END IF                                                           251300
      ETOUT1=ETOUT1+Q(IN)                                              251400
      END IF                                                           251500
   10 CONTINUE                                                         251600
      RETURN                                                           251700
      END                                                              251800
      SUBROUTINE VSPLNT                                                251900
C******                                                                252000
CVSPLNT                                                                252100
C******                                                                252200
C                                                                      252300
C   THIS SUBROUTINE COMPUTES ACTUAL ET AS A FUNCTION OF A ROOT         252400
C        ACTIVITY FUNCTION, HYDRAULIC CONDUCTIVITY OF THE SOIL,        252500
C        AND THE DIFFERENCE IN PRESSURE HEAD BETWEEN THE ROOTS AND     252600
C        THE SOIL                                                      252700
C                                                                      252800
C ---------------------------------------------------------------      252900
C                                                                      253000
C   SPECIFICATIONS FOR ARRAYS AND SCALARS                              253100
C                                                                      253200
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                              253300
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2             253400
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                            253500
      COMMON/KCON/HX(1600),NTYP(1600)                                  253600
      COMMON/PRESS/P(1600),PXXX(1600),CS1,CS2                          253700
      COMMON/DISCH/Q(1600),QQ(1600),ETOUT,ETOUT1                       253800
      COMMON/HCON/HCND(1600),HKLL(1600),HKTT(1600)                     253900
      COMMON/PTET/DPTH(1600),RT(1600),RDC(6,25),ETCYC,                 254000
     &PEVAL(25),PTVAL(25),PET,PEV,HROOT,HA,SRES,RTDPTH,                254100
     &RTBOT,RTTOP,NPV                                                  254200
      COMMON/TCON/STIM,DSMAX,KTIM,NIT,NIT1,KP                          254300
```

```
      LOGICAL RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                        254400
      COMMON/LOG1/RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                    254500
C                                                                       254600
C   SUM TRANSPIRATION FOR EACH COLUMN                                   254700
C                                                                       254800
      ETOUT=0                                                           254900
      IF(PET.GE. 0)RETURN                                               255000
      DO 50 I=2,NXRR                                                    255100
      ETR=0                                                             255200
      AREA=DXR(I)                                                       255300
      IF (RAD) AREA=PI2*RX(I)*DXR(I)                                    255400
      PETT=AREA*PET                                                     255500
      DO 10 J=2,NLYY                                                    255600
C                                                                       255700
C   COMPUTE TRANSPIRATION FOR EACH NODE IN COLUMN                       255800
C                                                                       255900
      IN=NLY*(I-1)+J                                                    256000
      IF(NTYP(IN).EQ.0.AND.HX(IN).GT.0) THEN                            256100
      VOL=AREA*DELZ(J)                                                  256200
      IF(DPTH(IN).GT.RTDPTH) GO TO 20                                   256300
C                                                                       256400
C   TRANSPIRATION IS ZERO IF NTYP IS NOT 0, NODE IS DEEPER              256500
C   THAN RTDPTH, OR PRESSURE IS LESS THAN HROOT                         256600
C                                                                       256700
      IF(CS1.EQ.1.) THEN                                                256800
      Z1=DZZ(J)                                                         256900
      ELSE                                                              257000
      Z1=DZZ(J)*CS1+(RX(I))*CS2                                         257100
      END IF                                                            257200
      PTMP=P(IN)+Z1                                                     257300
      IF(PTMP.LE.HROOT) THEN                                            257400
      Q(IN)=0                                                           257500
      ELSE                                                              257600
      HXX=HCND(IN)*HX(IN)*RT(IN)*VOL                                    257700
C                                                                       257800
C   Q IS TRANSPIRATION FOR EACH NODE.  ETR IS TOTAL FOR COLUMN          257900
C                                                                       258000
      Q(IN)=(HROOT-PTMP)*HXX                                            258100
      ETR=ETR+Q(IN)                                                     258200
      END IF                                                            258300
      END IF                                                            258400
   10 CONTINUE                                                          258500
   20 IF(ETR.LT.PETT) THEN                                              258600
C                                                                       258700
C   IF TOTAL TRANSPIRATION FOR COLUMN IS GREATER                        258800
C   THAN POTENTIAL THEN ADJUST TRANSPIRATION VALUES                     258900
C                                                                       259000
      R1=PETT/ETR                                                       259100
      ETR=PETT                                                          259200
      DO 30 K=2,J                                                       259300
      IN=NLY*(I-1)+K                                                    259400
      IF(HX(IN).GT.0.AND.NTYP(IN).EQ.0) THEN                            259500
      IF(DPTH(IN).GT.RTDPTH) GO TO 40                                   259600
      Q(IN)=Q(IN)*R1                                                    259700
      END IF                                                            259800
   30 CONTINUE                                                          259900
   40 CONTINUE                                                          260000
      END IF                                                            260100
      ETOUT=ETOUT+ETR                                                   260200
   50 CONTINUE                                                          260300
      RETURN                                                            260400
      END                                                               260500
      SUBROUTINE VSPET                                                  260600
C*****                                                                  260700
CVSPET                                                                  260800
C*****                                                                  260900
C                                                                       261000
```

```
C    PURPOSE: TO COMPUTE VALUES OF PEV,SRES,HA,PET,RTDPTH,RTBOT,RTTOP,    261100
C             AND HROOT FOR EVAPORATION AND TRANSPIRATION CALCULATIONS.   261200
C             VALUES ARE DETERMINED BY LINEAR INTERPOLATION IN TIME       261300
C             BETWEEN EVAPOTRANSPIRATION PERIODS.                         261400
C                                                                         261500
C-------------------------------------------------------------------      261600
C                                                                         261700
C    SPECIFICATIONS FOR ARRAYS AND SCALARS                                261800
C                                                                         261900
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                 262000
      COMMON/PTET/DPTH(1600),RT(1600),RDC(6,25),ETCYC,                    262100
     &PEVAL(25),PTVAL(25),PET,PEV,HROOT,HA,SRES,RTDPTH,                   262200
     &RTBOT,RTTOP,NPV                                                     262300
      COMMON/TCON/STIM,DSMAX,KTIM,NIT,NIT1,KP                             262400
      LOGICAL RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                          262500
      COMMON/LOG1/RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                      262600
C                                                                         262700
C-------------------------------------------------------------------      262800
C                                                                         262900
      IF (NPV.EQ.1) THEN                                                  263000
C                                                                         263100
C   IF ONLY 1 PERIOD THEN ALL VALUES ARE CONSTANT                         263200
C                                                                         263300
      IF(BCIT) THEN                                                       263400
      PEV=-PEVAL(1)                                                       263500
      SRES=RDC(1,1)                                                       263600
      HA=RDC(2,1)                                                         263700
      END IF                                                              263800
      IF(ETSIM) THEN                                                      263900
      PET=-PTVAL(1)                                                       264000
      RTDPTH=RDC(3,1)                                                     264100
      RTBOT=RDC(4,1)                                                      264200
      RTTOP=RDC(5,1)                                                      264300
      HROOT=RDC(6,1)                                                      264400
      END IF                                                              264500
      ELSE                                                                264600
C                                                                         264700
C   DETERMINE WHICH PERIOD TO USE                                         264800
C                                                                         264900
      ETCYC1=NPV*ETCYC                                                    265000
      SITY=MOD(STIM,ETCYC1)                                               265100
      I=(SITY/ETCYC)+2                                                    265200
      IF(I.EQ.1) THEN                                                     265300
      K=NPV                                                               265400
      ELSE                                                                265500
      K=I-1                                                               265600
      END IF                                                              265700
C                                                                         265800
C   LINEARLY INTERPOLATE                                                  265900
C                                                                         266000
      FRPER=(MOD(SITY,ETCYC))/ETCYC                                       266100
      IF (BCIT) THEN                                                      266200
      PEV=-PEVAL(K)-(PEVAL(I)-PEVAL(K))*FRPER                             266300
      SRES=RDC(1,K)+(RDC(1,I)-RDC(1,K))*FRPER                             266400
      HA=RDC(2,K)+(RDC(2,I)-RDC(2,K))*FRPER                               266500
      END IF                                                              266600
      IF (ETSIM) THEN                                                     266700
      PET=-PTVAL(K)-(PTVAL(I)-PTVAL(K))*FRPER                             266800
      RTDPTH=RDC(3,K)+(RDC(3,I)-RDC(3,K))*FRPER                           266900
      RTBOT=RDC(4,K)+(RDC(4,I)-RDC(4,K))*FRPER                            267000
      RTTOP=RDC(5,K)+(RDC(5,I)-RDC(5,K))*FRPER                            267100
      HROOT=RDC(6,K)+(RDC(6,I)-RDC(6,K))*FRPER                            267200
      END IF                                                              267300
      END IF                                                              267400
      RETURN                                                              267500
      END                                                                 267600
      DOUBLE PRECISION FUNCTION VSRDF(Z1,Z2)                              267700
```

```
C*****                                                                  267800
CVSRDF                                                                  267900
C*****                                                                  268000
C                                                                       268100
C     PURPOSE: TO DETERMINE THE ROOT ACTIVITY AT EACH NODE WITHIN       268200
C              THE ROOT ZONE FOR EACH TIME STEP                         268300
C                                                                       268400
C                                                                       268500
C-----------------------------------------------------------------      268600
C                                                                       268700
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               268800
      COMMON/PTET/DPTH(1600),RT(1600),RDC(6,25),ETCYC,                  268900
     &PEVAL(25),PTVAL(25),PET,PEV,HROOT,HA,SRES,RTDPTH,                 269000
     &RTBOT,RTTOP,NPV                                                   269100
C                                                                       269200
C-------------------------------------------------------------          269300
C                                                                       269400
C                                                                       269500
C   LINEARLY INTERPOLATE USING DEPTH OF NODE AND MAXIMUM ROOT DEPTH     269600
C                                                                       269700
      IF(RTDPTH.GT.Z1.AND.RTDPTH.GT.0)THEN                              269800
      IF(RTDPTH.GE.Z1+Z2)THEN                                           269900
      ZZ=Z1+0.5*Z2                                                      270000
      ZZ1=1.                                                            270100
      ELSE                                                              270200
      ZZ=(Z1+RTDPTH)*0.5                                                270300
      ZZ1=(RTDPTH-Z1)/Z2                                                270400
      END IF                                                            270500
      VSRDF=ZZ1*(ZZ*RTBOT+(RTDPTH-ZZ)*RTTOP)/RTDPTH                     270600
      ELSE                                                              270700
      VSRDF=0.0                                                         270800
      END IF                                                            270900
      RETURN                                                            271000
      END                                                               271100
      DOUBLE PRECISION FUNCTION VSDTHU(P,I,HK)                          271200
C******                                                                 271300
CVSDTHU                                                                 271400
C******                                                                 271500
C                                                                       271600
C    FIRST DERIVATIVE OF MOISTURE CONTENT AS A FUNCTION OF PRESSURE HEAD 271700
C                                                                       271800
C   VAN GENUCHTEN FUNCTION                                              271900
C                                                                       272000
C          HK(I,1)=SATURATED HYDRAULIC CONDUCTIVITY                     272100
C          HK(I,2)=SPECIFIC STORAGE                                     272200
C          HK(I,3)=POROSITY                                             272300
C          HK(I,4)=ALPHA PRIME                                          272400
C          HK(I,5)=RESIDUAL MOISTURE CONTENT                            272500
C          HK(I,6)=BETA PRIME                                           272600
C                                                                       272700
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               272800
      DIMENSION HK(10,100)                                              272900
      VSDTHU=0.D0                                                       273000
      IF(P.GE.0.0)RETURN                                                273100
      SE=HK(I,3)-HK(I,5)                                                273200
      EN=HK(I,6)                                                        273300
      EM=2.-1./EN                                                       273400
      ALPH=HK(I,4)                                                      273500
      A=P/ALPH                                                          273600
      VSDTHU=-(EN-1)*SE*A**(EN-1)/(ALPH*(1+A**EN)**EM)                  273700
      RETURN                                                            273800
      END                                                               273900
      DOUBLE PRECISION FUNCTION VSTHNV(V,I,HK)                          274000
C******                                                                 274100
CVSTHNV                                                                 274200
C******                                                                 274300
C                                                                       274400
```

```
C     INITIAL UNSATURATED PRESSURE HEAD AS A FUNCTION OF VOLUMETRIC        274500
C     MOISTURE CONTENT                                                    274600
C                                                                         274700
C     VAN GENUCHTEN FUNCTION                                              274800
C                                                                         274900
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                 275000
      DIMENSION HK(10,100)                                                275100
      VSTHNV=0.0                                                          275200
      IF(V.GE.HK(I,3)) RETURN                                             275300
      IF(V.GT.HK(I,5)) GO TO 10                                           275400
      WRITE(6,4000) V,I                                                   275500
      STOP                                                                275600
   10 SE=(V-HK(I,5))/(HK(I,3)-HK(I,5))                                    275700
      EN=HK(I,6)                                                          275800
      EM=1.-1./EN                                                         275900
      ALPH=HK(I,4)                                                        276000
      VSTHNV=ALPH*(1/SE**(1/EM)-1)**(1-EM)                                276100
      RETURN                                                              276200
 4000 FORMAT(/,28HINITIAL MOISTURE CONTENT OF ,F7.3,49HIS LESS THAN RES   276300
     &IDUAL MOISTURE CONTENT FOR CLASS ,I4,/,                             276400
     &14HPROGRAM HALTED)                                                  276500
      END                                                                 276600
      DOUBLE PRECISION FUNCTION VSTHU(P,I,HK)                             276700
C*****                                                                    276800
CVSTHU                                                                    276900
C*****                                                                    277000
C                                                                         277100
C     MOISTURE CONTENT AS A FUNCTION OF PRESSURE HEAD                     277200
C                                                                         277300
C  VAN GENUCHTEN FUNCTION                                                 277400
C                                                                         277500
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                 277600
      DIMENSION HK(10,100)                                                277700
      VSTHU=HK(I,3)                                                       277800
      IF(P .GE. 0.0)RETURN                                                277900
      EN=HK(I,6)                                                          278000
      EM=-(1.-1./EN)                                                      278100
      A=HK(I,3)-HK(I,5)                                                   278200
      ALPH=HK(I,4)                                                        278300
      VSTHU=HK(I,5)+A*(1+(P/ALPH)**EN)**EM                                278400
      RETURN                                                              278500
      END                                                                 278600
      DOUBLE PRECISION FUNCTION VSHKU(P,I,HK)                             278700
C*****                                                                    278800
CVSHKU                                                                    278900
C*****                                                                    279000
C                                                                         279100
C     RELATIVE HYDRAULIC CONDUCTIVITY WITH RESPECT TO PRESSURE HEAD       279200
C                                                                         279300
C   VAN GENUCHTEN FUNCTION                                                279400
C                                                                         279500
C                                                                         279600
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                 279700
      DIMENSION HK(10,100)                                                279800
      VSHKU=1.00                                                          279900
      IF(P.GE.0.0)RETURN                                                  280000
      EN=HK(I,6)                                                          280100
      EM=1.-1./EN                                                         280200
      A=P/HK(I,4)                                                         280300
      TOP=A**EN                                                           280400
      DEN=(1+TOP)**(EM/2.)                                                280500
      TOP=1-TOP/A*(1+TOP)**(-EM)                                          280600
      VSHKU=TOP*TOP/DEN                                                   280700
      RETURN                                                              280800
      END                                                                 280900
C                                                                         281000
C                                                                         281100
```

```
C                                                                       281200
C   NOTE -- AS LISTED HERE THE PROGRAM USES THE FUNCTIONAL RELATIONS    281300
C           OF THE VAN GENUCHTEN FORM.                                  281400
C           FUNCTIONS FOR THE THREE ALTERNATIVE RELATIONS ARE LISTED    281500
C           BELOW.  TO USE ONE OF THESE: FIRST PLACE A 'C' (FOR COMMENT) 281600
C           IN THE FIRST COLUMN OF EVERY LINE IN THE VAN GENUCHTEN      281700
C           ROUTINES.  NEXT REMOVE THE COMMENT DESIGNATIONS FOR THE     281800
C           DESIRED SET OF ROUTINES -- 'C&' FOR BROOKS-COREY            281900
C                                      'C$' FOR HAVERKAMP               282000
C                                      'C+' FOR TABULAR DATA            282100
C                                                                       282200
C&    DOUBLE PRECISION FUNCTION VSDTHU(P,I,HK)                          282300
C******                                                                 282400
CVSDTHU                                                                 282500
C******                                                                 282600
C                                                                       282700
C    FIRST DERIVATIVE OF MOISTURE CONTENT AS A FUNCTION OF PRESSURE HEAD 282800
C                                                                       282900
C    BROOKS AND COREY, CSU HYDROLOGY PAPER NO. 17 PP.3-4                283000
C                                                                       283100
C          HK(I,1)=SATURATED HYDRAULIC CONDUCTIVITY                     283200
C          HK(I,2)=SPECIFIC STORAGE                                     283300
C          HK(I,3)=POROSITY                                             283400
C          HK(I,4)=BUBBLING PRESSURE                                    283500
C          HK(I,5)=RESIDUAL MOISTURE CONTENT                            283600
C          HK(I,6)=LAMBDA                                               283700
C                                                                       283800
C&    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               283900
C&    DIMENSION HK(10,100)                                              284000
C&    VSDTHU=0.DO                                                       284100
C&    IF(P.GE.HK(I,4))RETURN                                            284200
C&    VSDTHU=-((HK(I,3)-HK(I,5))*HK(I,6)*(HK(I,4)/P)**HK(I,6))/P        284300
C&    RETURN                                                            284400
C&    END                                                               284500
C&    DOUBLE PRECISION FUNCTION VSTHNV(V,I,HK)                          284600
C******                                                                 284700
CVSTHNV                                                                 284800
C******                                                                 284900
C                                                                       285000
C    INITIAL UNSATURATED PRESSURE HEAD AS A FUNCTION OF VOLUMETRIC      285100
C    MOISTURE CONTENT                                                   285200
C                                                                       285300
C    BROOKS AND COREY, CSU HYDROLOGY PAPER NO. 17 , PP.3-4              285400
C                                                                       285500
C&    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               285600
C&    DIMENSION HK(10,100)                                              285700
C&    VSTHNV=HK(I,4)                                                    285800
C&    IF(V.GE.HK(I,3)) RETURN                                           285900
C&    IF(V.GT.HK(I,5)) GO TO 1                                          286000
C&    WRITE(6,100) V,I                                                  286100
C&100 FORMAT(/,28HINITIAL MOISTURE CONTENT OF ,F7.3,49HIS LESS THAN RES 286200
C&   1IDUAL MOISTURE CONTENT FOR CLASS ,I4,/,                           286300
C&   214HPROGRAM HALTED)                                               286400
C&    STOP                                                              286500
C&1   SE=(V-HK(I,5))/(HK(I,3)-HK(I,5))                                  286600
C&    VSTHNV=HK(I,4)/(SE**(1.00/HK(I,6)))                               286700
C&    RETURN                                                            286800
C&    END                                                               286900
C&    DOUBLE PRECISION FUNCTION VSTHU(P,I,HK)                           287000
C*****                                                                  287100
CVSTHU                                                                  287200
C*****                                                                  287300
C                                                                       287400
C    MOISTURE CONTENT AS A FUNCTION OF PRESSURE HEAD BELOW BUBBLING     287500
C    PRESSURE: = POROSITY ELSEWHERE                                     287600
C                                                                       287700
C    BROOKS AND COREY, CSU HYDROLOGY PAPER NO.17, PP.3-4                287800
```

```
C                                                                       287900
C&    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               288000
C&    DIMENSION HK(10,100)                                              288100
C&    VSTHU=HK(I,3)                                                     288200
C&    IF(P.GE.HK(I,4))RETURN                                            288300
C&    VSTHU=HK(I,5)+(HK(I,3)-HK(I,5))*(HK(I,4)/P)**HK(I,6)              288400
C&    RETURN                                                            288500
C&    END                                                               288600
C&    DOUBLE PRECISION FUNCTION VSHKU(P,I,HK)                           288700
C*****                                                                  288800
CVSHKU                                                                  288900
C*****                                                                  289000
C                                                                       289100
C    RELATIVE HYDRAULIC CONDUCTIVITY WITH RESPECT TO PRESSURE HEAD      289200
C                                                                       289300
C    BROOKS AND COREY, CSU HYDROLOGY PAPER NO. 3                        289400
C                                                                       289500
C                                                                       289600
C&    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               289700
C&    DIMENSION HK(10,100)                                              289800
C&    VSHKU=1.00                                                        289900
C&    IF(P.GE.HK(I,4))RETURN                                            290000
C&    VSHKU=(HK(I,4)/P)**(2.+3.*HK(I,6))                                290100
C&    IF(VSHKU.LT.1.D-38)VSHKU=0.00                                     290200
C&    RETURN                                                            290300
C&    END                                                               290400
C$    DOUBLE PRECISION FUNCTION VSDTHU(P,I,HK)                          290500
C******                                                                 290600
CVSDTHU                                                                 290700
C******                                                                 290800
C                                                                       290900
C    FIRST DERIVATIVE OF MOISTURE CONTENT AS A FUNCTION OF PRESSURE HEAD 291000
C                                                                       291100
C   HAVERKAMP FUNCTION                                                  291200
C                                                                       291300
C          HK(I,1)=SATURATED HYDRAULIC CONDUCTIVITY                     291400
C          HK(I,2)=SPECIFIC STORAGE                                     291500
C          HK(I,3)=POROSITY                                             291600
C          HK(I,4)=A PRIME                                              291700
C          HK(I,5)=RESIDUAL MOISTURE CONTENT                            291800
C          HK(I,6)=B PRIME                                              291900
C          HK(I,7)=ALPHA                                                292000
C          HK(I,8)=BETA                                                 292100
C                                                                       292200
C$    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               292300
C$    DIMENSION HK(10,100)                                              292400
C$    VSDTHU=0.D0                                                       292500
C$    IF(P.GE.0.0)RETURN                                                292600
C$    SE=HK(I,3)-HK(I,5)                                                292700
C$    ALPH=HK(I,7)                                                      292800
C$    EM=HK(I,8)                                                        292900
C$    TOP=P/ALPH                                                        293000
C$    DEN=1+TOP**EM                                                     293100
C$    DEN=DEN*DEN                                                       293200
C$    VSDTHU=SE*EM*TOP**(EM-1)/(ALPH*DEN)                               293300
C$    RETURN                                                            293400
C$    END                                                               293500
C$    DOUBLE PRECISION FUNCTION VSTHNV(V,I,HK)                          293600
C******                                                                 293700
CVSTHNV                                                                 293800
C******                                                                 293900
C                                                                       294000
C    INITIAL UNSATURATED PRESSURE HEAD AS A FUNCTION OF VOLUMETRIC      294100
C    MOISTURE CONTENT                                                   294200
C                                                                       294300
C    HAVERKAMP FUNCTION                                                 294400
C                                                                       294500
```

```
C$    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                  294600
C$    DIMENSION HK(10,100)                                                 294700
C$    VSTHNV=0.0                                                           294800
C$    IF(V.GE.HK(I,3)) RETURN                                              294900
C$    IF(V.GT.HK(I,5)) GO TO 1                                             295000
C$    WRITE(6,100) V,I                                                     295100
C$100 FORMAT(/,28HINITIAL MOISTURE CONTENT OF ,F7.3,49HIS LESS THAN RES    295200
C$   1IDUAL MOISTURE CONTENT FOR CLASS ,I4,/,                              295300
C$   214HPROGRAM HALTED)                                                   295400
C$    STOP                                                                 295500
C$1   SE=(V-HK(I,5))/(HK(I,3)-HK(I,5))                                     295600
C$    VSTHNV=HK(I,7)*(1.0/SE-1.0)**(1.0/HK(I,8))                           295700
C$    RETURN                                                               295800
C$    END                                                                  295900
C$    DOUBLE PRECISION FUNCTION VSTHU(P,I,HK)                              296000
C******                                                                    296100
CVSTHU                                                                     296200
C******                                                                    296300
C                                                                          296400
C    MOISTURE CONTENT AS A FUNCTION OF PRESSURE HEAD                       296500
C                                                                          296600
C  HAVERKAMP FUNCTION                                                      296700
C                                                                          296800
C$    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                  296900
C$    DIMENSION HK(10,100)                                                 297000
C$    VSTHU=HK(I,3)                                                        297100
C$    IF(P .GE. 0.0)RETURN                                                 297200
C$    VSTHU=HK(I,5)+(HK(I,3)-HK(I,5))/((P/HK(I,7))**HK(I,8)+1.)            297300
C$    RETURN                                                               297400
C$    END                                                                  297500
C$    DOUBLE PRECISION FUNCTION VSHKU(P,I,HK)                              297600
C*****                                                                     297700
CVSHKU                                                                     297800
C*****                                                                     297900
C                                                                          298000
C    RELATIVE HYDRAULIC CONDUCTIVITY WITH RESPECT TO PRESSURE HEAD         298100
C                                                                          298200
C  HAVERKAMP FUNCTION                                                      298300
C                                                                          298400
C                                                                          298500
C$    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                  298600
C$    DIMENSION HK(10,100)                                                 298700
C$    VSHKU=1.00                                                           298800
C$    IF(P.GE.0.0)RETURN                                                   298900
C$    VSHKU=1.0/((P/HK(I,4))**HK(I,6)+1)                                   299000
C$    RETURN                                                               299100
C$    END                                                                  299200
C ***************************************************************************299300
C ***************************************************************************299400
C                                                                          299500
C+    SUBROUTINE INTERP (P,I,HK)                                           299600
C******                                                                    299700
CINTERP                                                                    299800
C******                                                                    299900
C                                                                          300000
C   THIS SUBROUTINE PERFORMS LINEAR INTERPOLATION OF PRESSURE              300100
C   HEADS FOR RELATIVE HYDRAULIC CONDUCTIVITY (VSHKU), VOLUMETRIC          300200
C   MOISTURE CONTENT (VSTHU), AND MOISTURE CAPACITY (VSDTHU).              300300
C                                                                          300400
C                                                                          300500
C   TO USE THIS METHOD FOR EVALUATING THE NONLINEAR FUNCTIONS,             300600
C   THE USER MUST ENTER A TABLE OF PRESSURE HEADS                          300700
C   AND VALUES OF RELATIVE                                                 300800
C   CONDUCTIVITIES,AND MOISTURE CONTENTS                                   300900
C   WHICH CORRESPOND TO EACH PRESSURE HEAD INTO ARRAY HK ON                301000
C   B-7 CARDS FOR EACH TEXTURAL CLASS.  SET NPROP (CARD B-5) EQUAL         301100
C   TO 3*(NUMBER OF PRESSURE HEADS IN TABLE + 1).                          301200
```

```
C   BEGINNING WITH HK(ITEX,4), ENTER ALL PRESSURE HEADS IN DESCENDING     301300
C   ORDER STARTING WITH THE HIGHEST VALUE,                                301400
C   NEXT ENTER THE NUMBER 99,                                             301500
C   NEXT ENTER THE RELATIVE HYDRAULIC                                     301600
C   CONDUCTIVITY FOR EACH PRESSURE HEAD,                                  301700
C   NEXT ENTER THE NUMBER 99,                                             301800
C   NEXT ENTER THE VOLUMETRIC MOISTURE CONTENT FOR EACH PRESSURE          301900
C   HEAD, FINALLY ENTER THE NUMBER 99.                                    302000
C                                                                         302100
C+    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                 302200
C+    DIMENSION HK(10,100)                                                302300
C+    COMMON I1,I2,I3,I4,I5,I6,DELP                                       302400
C+    IF (I2.GT.0) GO TO 1                                                302500
C+    I2=4                                                                302600
C+    DO 2 J=I2,100                                                       302700
C+    IF (HK(I,J).LT.99) GO TO 2                                          302800
C+    I3=J-I2+1                                                           302900
C+    I1=I3+I3                                                            303000
C+    GO TO 1                                                             303100
C+ 2  CONTINUE                                                            303200
C+ 1  IF(HK(I,I2).LE.P) THEN                                              303300
C+    DELP=0                                                              303400
C+    I5=I2                                                               303500
C+    I6=I2                                                               303600
C+    ELSE                                                                303700
C+    I4=I2+I3-2                                                          303800
C+    IF(HK(I,I4).GE.P)THEN                                               303900
C+    I5=I4-1                                                             304000
C+    I6=I4                                                               304100
C+    DELP=0                                                              304200
C+    ELSE                                                                304300
C+    I4=I4-1                                                             304400
C+    DO 3 J=I2+1,I4                                                      304500
C+    IF(HK(I,J).GT.P) GO TO 3                                            304600
C+    I5=J-1                                                              304700
C+    I6=J                                                                304800
C+    DELP=(P-HK(I,I6))/(HK(I,I5)-HK(I,I6))                               304900
C+    RETURN                                                              305000
C+ 3  CONTINUE                                                            305100
C+    END IF                                                              305200
C+    END IF                                                              305300
C+    RETURN                                                              305400
C+    END                                                                 305500
C+    DOUBLE PRECISION FUNCTION VSHKU (P,I,HK)                            305600
C*****                                                                    305700
CVSHKU                                                                    305800
C*****                                                                    305900
C                                                                         306000
C   RELATIVE HYDRAULIC CONDUCTIVITY AS A FUNCTION OF PRESSURE HEAD        306100
C   DETERMINED BY LINEAR INTERPOLATION OF KR VS HP TABLE WHICH IS         306200
C   INPUT BY USER.                                                        306300
C                                                                         306400
C+    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                 306500
C+    DIMENSION HK(10,100)                                                306600
C+    COMMON I1,I2,I3,I4,I5,I6,DELP                                       306700
C+    CALL INTERP (P,I,HK)                                                306800
C+    IF(I5.EQ.I6)THEN                                                    306900
C+    VSHKU=HK(I,I3+I5)                                                   307000
C+    RETURN                                                              307100
C+    ELSE                                                                307200
C+    VSHKU=HK(I,I3+I6)+(HK(I,I3+I5)-HK(I,I3+I6))*DELP                    307300
C+    RETURN                                                              307400
C+    END IF                                                              307500
C+    END                                                                 307600
C+    DOUBLE PRECISION FUNCTION VSDTHU(P,I,HK)                            307700
C******                                                                   307800
CVSDTHU                                                                   307900
```

```
C******                                                                 308000
C                                                                       308100
C   MOISTURE CAPACITY AS A FUNCTION OF PRESSURE HEAD AS                 308200
C   DETERMINED FROM TABLE OF THETA VS HP WHICH IS INPUT                 308300
C   BY USER.                                                            308400
C                                                                       308500
C+    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               308600
C+    DIMENSION HK(10,100)                                              308700
C+    COMMON I1,I2,I3,I4,I5,I6,DELP                                     308800
C+    IF (I5.EQ.I6) THEN                                                308900
C+    VSDTHU=0.                                                         309000
C+    RETURN                                                            309100
C+    ELSE                                                              309200
C+    VSDTHU=(HK(I,I1+I5)-HK(I,I1+I6))/(HK(I,I5)-HK(I,I6))              309300
C+    RETURN                                                            309400
C+    END IF                                                            309500
C+    END                                                               309600
C+    DOUBLE PRECISION FUNCTION VSTHU (P,I,HK)                          309700
C*****                                                                  309800
CVSTHU                                                                  309900
C*****                                                                  310000
C                                                                       310100
C   VOLUMETRIC MOISTURE CONTENT AS A FUNCTION OF PRESSURE HEAD          310200
C   AS DETERMINED BY LINEAR INTERPOLATION OF THETA VS HP TABLE          310300
C   WHICH IS INPUT BY USER.                                             310400
C                                                                       310500
C+    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               310600
C+    DIMENSION HK(10,100)                                              310700
C+    COMMON I1,I2,I3,I4,I5,I6,DELP                                     310800
C+    IF (DELP.EQ.0) THEN                                               310900
C+    VSTHU=HK(I,I1+I6)                                                 311000
C+    ELSE                                                              311100
C+    VSTHU=HK(I,I1+I6)+(HK(I,I1+I5)-HK(I,I1+I6))*DELP                  311200
C+    END IF                                                            311300
C+    RETURN                                                            311400
C+    END                                                               311500
C+    DOUBLE PRECISION FUNCTION VSTHNV(P,I,HK)                          311600
C******                                                                 311700
CVSTHNV                                                                 311800
C******                                                                 311900
C                                                                       312000
C      NOTE -- THIS FUNCTION IS NOT OPERATIVE WHEN USING INTERPOLATION  312100
C              ROUTINES.  INITIAL CONDITIONS MUST BE INPUT IN TERMS OF  312200
C              PRESSURE HEADS NOT MOISTURE CONTENTS.                    312300
C                                                                       312400
C+    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               312500
C+    DIMENSION HK(10,100)                                              312600
C+    WRITE(6,100)                                                      312700
C+    STOP                                                              312800
C+100 FORMAT(5X,'INPUT OF MOISTURE CONTENT FOR INITIAL CONDITIONS IS ',  312900
C+   1'NOT ALLOWED WHEN USING TABULAR DATA '/                           313000
C+   25X,'FOR MOISTURE RETENTION AND CONDUCTIVITY CURVES',/             313100
C+   35X,'SIMULATION TERMINATED')                                       313200
C+    END                                                               313300
      SUBROUTINE VTVELO                                                 313400
C******                                                                 313500
CVTVELO                                                                 313600
C******                                                                 313700
C                                                                       313800
C    ROUTINE TO CALCULATE VELOCITIES AT BOUNDARIES OF ADJACENT CELLS    313900
C    VX IS VELOCITY IN X-DIRECTION BETWEEN CURRENT NODE AND NODE TO     314000
C    THE LEFT.                                                          314100
C    VZ IS VELOCITY IN Z-DIRECTION BETWEEN CURRENT NODE AND NODE        314200
C    ABOVE.                                                             314300
C                                                                       314400
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               314500
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2              314600
```

```
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                              314700
      COMMON/KCON/HX(1600),NTYP(1600)                                    314800
      COMMON/RPROP/HK(10,100),HT(10,20),ANIZ(10)                         314900
      COMMON/MPROP/THETA(1600),THLST(1600)                               315000
      COMMON/PRESS/P(1600),PXX(1600),CS1,CS2                             315100
      COMMON/HCON/HCND(1600),HKLL(1600),HKTT(1600)                       315200
      COMMON/JTXX/JTEX(1600)                                             315300
      COMMON/WGT/WUS,WDS                                                 315400
      COMMON/TRXX/DX1(1600),DX2(1600),DZ1(1600),DZ2(1600),VX(1600),      315500
     &VZ(1600),CC(1600),COLD(1600),CS(1600),QT(1600),NCTYP(1600),        315600
     &RET(1600)                                                          315700
      LOGICAL RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                         315800
      COMMON/LOG1/RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                     315900
      DO 10 I=2,NXRR                                                     316000
      N1=NLY*(I-1)                                                       316100
      DO 10 J=2,NLYY                                                     316200
      N=N1+J                                                             316300
      VX(N)=0                                                            316400
      VZ(N)=0                                                            316500
      IF(HX(N).NE.0) THEN                                                316600
      JM1=N-1                                                            316700
      IM1=N-NLY                                                          316800
      IF(HX(JM1).NE.0) THEN                                              316900
C                                                                        317000
C   CALCULATE VERTICAL VELOCITY                                          317100
C                                                                        317200
      AREA=DXR(I)                                                        317300
      IF (RAD) AREA=PI2*RX(I)*DXR(I)                                     317400
      GRAD=P(JM1)-P(N)                                                   317500
      THETA1=0.5*(THETA(N)+THETA(JM1))*AREA                              317600
      IF(WUS.EQ.0.) THEN                                                 317700
      VZ(N)=HKTT(N)*DSQRT(HCND(N)*HCND(JM1))*GRAD/THETA1                 317800
      ELSE                                                               317900
      IF(P(JM1).GT.P(N))THEN                                             318000
      ALA=WUS                                                            318100
      BTA=WDS                                                            318200
      ELSE                                                               318300
      ALA=WDS                                                            318400
      BTA=WUS                                                            318500
      END IF                                                             318600
      VZ(N)=HKTT(N)*(ALA*HCND(JM1)+BTA*HCND(N))*GRAD/THETA1              318700
      END IF                                                             318800
      END IF                                                             318900
      IF(HX(IM1).NE.0) THEN                                              319000
C                                                                        319100
C   CALCULATE HORIZONTAL VELOCITY                                        319200
C                                                                        319300
      GRAD=P(IM1)-P(N)                                                   319400
      AREA=DELZ(J)                                                       319500
      IF (RAD) AREA=PI2*AREA*(RX(I)-0.5*DXR(I))                          319600
      THETA1=0.5*(THETA(N)+THETA(IM1))*AREA                              319700
      IF(WUS.EQ.0) THEN                                                  319800
      VX(N)=HKLL(N)*DSQRT(HCND(N)*HCND(IM1))*GRAD/THETA1                 319900
      ELSE                                                               320000
      IF(P(IM1).GT.P(N)) THEN                                            320100
      ALA=WUS                                                            320200
      BTA=WDS                                                            320300
      ELSE                                                               320400
      ALA=WDS                                                            320500
      BTA=WUS                                                            320600
      END IF                                                             320700
      VX(N)=HKLL(N)*(ALA*HCND(IM1)+BTA*HCND(N))*GRAD/THETA1              320800
      END IF                                                             320900
      END IF                                                             321000
      END IF                                                             321100
   10 CONTINUE                                                           321200
      RETURN                                                             321300
```

```
      END                                                               321400
      SUBROUTINE VTDCOEF                                                321500
C*******                                                                321600
CVTDCOEF                                                                321700
C*******                                                                321800
C                                                                       321900
C    ROUTINE TO CALCULATE DISPERSION COEFFICIENTS AS FUNCTIONS          322000
C    OF DISPERSIVITIES AND VELOCITIES.  DIAGNOL TERMS ARE               322100
C    CONTAINED IN ARRAYS DX1 AND DZ1.  CROSS PRODUCT TERMS              322200
C    ARE IN DX2 AND DZ2                                                 322300
C                                                                       322400
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               322500
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2              322600
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                             322700
      COMMON/KCON/HX(1600),NTYP(1600)                                   322800
      COMMON/RPROP/HK(10,100),HT(10,20),ANIZ(10)                        322900
      COMMON/MPROP/THETA(1600),THLST(1600)                              323000
      COMMON/JTXX/JTEX(1600)                                            323100
      COMMON/TRXX/DX1(1600),DX2(1600),DZ1(1600),DZ2(1600),VX(1600),     323200
     &VZ(1600),CC(1600),COLD(1600),CS(1600),QT(1600),NCTYP(1600),       323300
     &RET(1600)                                                         323400
      LOGICAL RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                        323500
      COMMON/LOG1/RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                    323600
      DO 10 I=2,NXRR                                                    323700
      N1=NLY*(I-1)                                                      323800
      DO 10 J=2,NLYY                                                    323900
      N=N1+J                                                            324000
      DX1(N)=0                                                          324100
      DX2(N)=0                                                          324200
      DZ1(N)=0                                                          324300
      DZ2(N)=0                                                          324400
      PEX=0.                                                            324500
      PEZ=0.                                                            324600
      IMX=0                                                             324700
      JMX=0                                                             324800
      IMZ=0                                                             324900
      JMZ=0                                                             325000
      IF(HX(N).NE.0) THEN                                               325100
      N2=JTEX(N)                                                        325200
      AL=HT(N2,1)                                                       325300
      AT=HT(N2,2)                                                       325400
      DM=HT(N2,3)                                                       325500
      V1=VX(N)                                                          325600
      V2=VZ(N)                                                          325700
      JM1=N-1                                                           325800
      IM1=N-NLY                                                         325900
      JP1=N+1                                                           326000
      IP1=N+NLY                                                         326100
      IP2=IP1-1                                                         326200
      IM2=IM1+1                                                         326300
      IF(HX(JM1).NE.0.) THEN                                            326400
      V3=0.25*(V1+VX(IP1)+VX(IP2)+VX(JM1))                              326500
      V32=V3*V3                                                         326600
      V22=V2*V2                                                         326700
      VV2=V32+V22                                                       326800
C                                                                       326900
C   CALCULATE DZ1 AND DZ2                                               327000
C                                                                       327100
      N2=JTEX(JM1)                                                      327200
      AL1=DSQRT(AL*HT(N2,1))                                            327300
      AT1=DSQRT(AT*HT(N2,2))                                            327400
      DM1=DSQRT(DM*HT(N2,3))                                            327500
      AREA=DXR(I)                                                       327600
      IF(RAD) AREA=PI2*AREA*RX(I)                                       327700
      T1=0.5*(THETA(JM1)+THETA(N))                                      327800
      DD1=(DZZ(J)-DZZ(J-1))/AREA                                        327900
      T2=T1/DD1                                                         328000
```

```
      IF(VV2.EQ.0.) THEN                                                328100
      DZ1(N)=DM1                                                        328200
      ELSE                                                              328300
      VAVE=DSQRT(VV2)                                                   328400
      DL=AL1*VAVE                                                       328500
      DT=AT1*VAVE                                                       328600
      DZ1(N)=(DL*V22+DT*V32)/VV2+DM1                                    328700
      DD1=(RX(I+1)-RX(I-1))/AREA                                        328800
      DZ2(N)=T1*(DL-DT)*V2*V3/(DD1*VV2)                                 328900
      END IF                                                            329000
C                                                                       329100
C  CALCULATE VERTICAL CELL PECLET NUMBER                                329200
C                                                                       329300
      PE=DABS(VZ(N))*(DZZ(J)-DZZ(J-1))/DZ1(N)                           329400
      DZ1(N)=T2*DZ1(N)                                                  329500
      IF(PE.GT.PEZ) THEN                                                329600
      PEZ=PE                                                            329700
      IMZ=I                                                             329800
      JMZ=J                                                             329900
      END IF                                                            330000
      END IF                                                            330100
      IF(HX(IM1).NE.0.) THEN                                            330200
      V3=0.25*(V2+VZ(JP1)+VZ(IM1)+VZ(IM2))                              330300
      V32=V3*V3                                                         330400
      V12=V1*V1                                                         330500
      VV2=V12+V32                                                       330600
C                                                                       330700
C   CALCULATE DX1 AND DX2                                               330800
C                                                                       330900
      N2=JTEX(IM1)                                                      331000
      AL1=DSQRT(AL*HT(N2,1))                                            331100
      AT1=DSQRT(AT*HT(N2,2))                                            331200
      DM1=DSQRT(DM*HT(N2,3))                                            331300
      AREA=DELZ(J)                                                      331400
      IF(RAD) AREA=PI2*AREA*(RX(I)-0.5*DXR(I))                          331500
      DD1=(RX(I)-RX(I-1))/AREA                                          331600
      T1=0.5*(THETA(IM1)+THETA(N))                                      331700
      T2=T1/DD1                                                         331800
      IF(VV2.EQ.0.) THEN                                                331900
      DX1(N)=DM1                                                        332000
      ELSE                                                              332100
      VAVE=DSQRT(VV2)                                                   332200
      DL=AL1*VAVE                                                       332300
      DT=AT1*VAVE                                                       332400
      DX1(N)=(DL*V12+DT*V32)/VV2+DM1                                    332500
      DD1=(DZZ(J+1)-DZZ(J-1))/AREA                                      332600
      DX2(N)=T1*(DL-DT)*V1*V3/(VV2*DD1)                                 332700
      END IF                                                            332800
C                                                                       332900
C  CALCULATE HORIZONTAL CELL PECLET NUMBER                              333000
C                                                                       333100
      PE=DABS(VX(N))*(RX(I)-RX(I-1))/DX1(N)                             333200
      DX1(N)=DX1(N)*T2                                                  333300
      IF(PE.GT.PEX) THEN                                                333400
      PEX=PE                                                            333500
      IMX=I                                                             333600
      JMX=J                                                             333700
      END IF                                                            333800
      END IF                                                            333900
      END IF                                                            334000
   10 CONTINUE                                                          334100
C                                                                       334200
C  WRITE MAXIMUM CELL PECLET NUMBERS                                    334300
C                                                                       334400
      WRITE(6,4000) PEX,JMX,IMX,PEZ,JMZ,IMZ                             334500
      RETURN                                                            334600
 4000 FORMAT(4X,'  MAXIMUM CELL PECLET NUMBER  --  HORIZONTAL  ',E14.5,  334700
```

```
     &'     ROW ',I4,'  COLUMN ',I4,/,38X,'VERTICAL    ',E14.5,          334800
     &'     ROW ',I4,'  COLUMN ',I4)                                     334900
      END                                                               335000
      SUBROUTINE VTSETUP                                                335100
C*******                                                                335200
CVTSETUP                                                                335300
C*******                                                                335400
C                                                                       335500
C    ROUTINE TO ASSEMBLE MATRIX EQUATIONS FOR ADVECTION-DISPERSION      335600
C    EQUATIONS AND TO CALL MATRIX SOLVER.                               335700
C                                                                       335800
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               335900
      COMMON/PRESS/P(1600),PXXX(1600),CS1,CS2                           336000
      COMMON/RSPAC/DELZ(600),DZZ(600),DXR(600),RX(600),PI2              336100
      COMMON/ISPAC/NLY,NLYY,NXR,NXRR,NNODES                             336200
      COMMON/KCON/HX(1600),NTYP(1600)                                   336300
      COMMON/RPROP/HK(10,100),HT(10,20),ANIZ(10)                        336400
      COMMON/MPROP/THETA(1600),THLST(1600)                              336500
      COMMON/DISCH/Q(1600),QQ(1600),ETOUT,ETOUT1                        336600
      COMMON/EQUAT/A(1600),B(1600),C(1600),D(1600),E(1600),RHS(1600),   336700
     &XI(1600)                                                          336800
      COMMON/JTXX/JTEX(1600)                                            336900
      COMMON/JCON/JSTOP,JFLAG                                           337000
      COMMON/SCON/DHMX(201),DELT,HMAX,TMAX,EPS,NUMT,ITMAX,MINIT,ITEST   337100
      COMMON/TCON/STIM,DSMAX,KTIM,NIT,NIT1,KP                           337200
      COMMON/TRXX/DX1(1600),DX2(1600),DZ1(1600),DZ2(1600),VX(1600),     337300
     &VZ(1600),CC(1600),COLD(1600),CS(1600),QT(1600),NCTYP(1600),       337400
     &RET(1600)                                                         337500
      COMMON/TRXY1/A0(1600),B0(1600),C0(1600),D0(1600),E0(1600)         337600
      LOGICAL TRANS,TRANS1,SORP,SSTATE                                  337700
      COMMON/TRXY/MB9(72),NMB9,EPS1,TRANS,TRANS1,SORP,SSTATE            337800
      LOGICAL RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                        337900
      COMMON/LOG1/RAD,BCIT,ETSIM,SEEP,ITSTOP,CIS,CIT                    338000
      SAVE JFLAG1                                                       338100
      IF(KTIM.EQ.1) THEN                                                338200
      JFLAG1=1                                                          338300
      DO 10N=1,NNODES                                                   338400
      A0(N)=0                                                           338500
      B0(N)=0                                                           338600
      C0(N)=0                                                           338700
      D0(N)=0                                                           338800
      E0(N)=0                                                           338900
   10 CONTINUE                                                          339000
      END IF                                                            339100
C                                                                       339200
C   INITIALIZE VARIABLES                                                339300
C                                                                       339400
      DO 20 I=2,NXRR                                                    339500
      N1=NLY*(I-1)                                                      339600
      DO 20 J=2,NLYY                                                    339700
      N=N1+J                                                            339800
      A(N)=0                                                            339900
      B(N)=0                                                            340000
      C(N)=0                                                            340100
      D(N)=0                                                            340200
      E(N)=0                                                            340300
      RHS(N)=0                                                          340400
      COLD(N)=CC(N)                                                     340500
      QT(N)=0                                                           340600
      IF(NTYP(N).EQ.1) QT(N)=VSFLX1(N)                                  340700
      IF(HX(N).NE.0) THEN                                               340800
      N2=JTEX(N)                                                        340900
      RET(N)=VTRET(CC(N),N2,HT)                                         341000
      IM1=N-NLY                                                         341100
      JM1=N-1                                                           341200
      JP1=N+1                                                           341300
      IP1=N+NLY                                                         341400
```

```
      IP2=IP1-1                                                          341500
      IM2=IM1+1                                                          341600
      IM3=IM1-1                                                          341700
      IP3=IP1+1                                                          341800
      IF(RAD) THEN                                                       341900
      AREAX=PI2*DELZ(J)*(RX(I)-0.5*DXR(I))                               342000
      AREAX1=PI2*DELZ(J)*(RX(I)+0.5*DXR(I))                              342100
      AREAZ=PI2*DXR(I)*RX(I)                                             342200
      ELSE                                                               342300
      AREAX=DELZ(J)                                                      342400
      AREAX1=AREAX                                                       342500
      AREAZ=DXR(I)                                                       342600
      END IF                                                             342700
      VOL=AREAZ*DELZ(J)                                                  342800
      AREAX=AREAX*0.5*(THETA(IM1)+THETA(N))                              342900
      AREAX1=AREAX1*0.5*(THETA(IP1)+THETA(N))                            343000
      AREAZ1=AREAZ*0.5*(THETA(JP1)+THETA(N))                             343100
      AREAZ=AREAZ*0.5*(THETA(JM1)+THETA(N))                              343200
C                                                                        343300
C   CALCULATE LHS OF MATRIX EQUATION                                     343400
C                                                                        343500
      SS=THETA(N)*P(N)*HK(N2,2)/HK(N2,3)                                 343600
      E(N)=-DX1(N)-DZ1(N)-DX1(IP1)-DZ1(JP1)                              343700
     &-VOL*(HT(N2,4)*(THETA(N)+SS+RET(N)))                               343800
      SS=THETA(N)+SS+SS-THLST(N)*(1+PXXX(N)*HK(N2,2)/HK(N2,3))           343900
      IF(HX(IM1).NE.0) THEN                                              344000
      A(N)=DX1(N)+0.5*(+DZ2(N)-DZ2(JP1))                                 344100
      IF(.NOT.CIS) THEN                                                  344200
      IF(VX(N).GT.0) THEN                                                344300
      A(N)=A(N)+AREAX*VX(N)                                              344400
      ELSE                                                               344500
      E(N)=E(N)+AREAX*VX(N)                                              344600
      END IF                                                             344700
      ELSE                                                               344800
      VV=AREAX*0.5*VX(N)                                                 344900
      A(N)=A(N)+VV                                                       345000
      E(N)=E(N)+VV                                                       345100
      END IF                                                             345200
      END IF                                                             345300
      IF(HX(JM1).NE.0) THEN                                              345400
      B(N)=DZ1(N)+0.5*(+DX2(N)-DX2(IP1))                                 345500
      IF(.NOT.CIS) THEN                                                  345600
      IF(VZ(N).GT.0) THEN                                                345700
      B(N)=B(N)+AREAZ*VZ(N)                                              345800
      ELSE                                                               345900
      E(N)=E(N)+AREAZ*VZ(N)                                              346000
      END IF                                                             346100
      ELSE                                                               346200
      VV=0.5*AREAZ*VZ(N)                                                 346300
      B(N)=B(N)+VV                                                       346400
      E(N)=E(N)+VV                                                       346500
      END IF                                                             346600
      END IF                                                             346700
      IF(HX(IP1).NE.0) THEN                                              346800
      C(N)=DX1(IP1)+0.5*(-DZ2(N)+DZ2(JP1))                               346900
      IF(.NOT.CIS) THEN                                                  347000
      IF(VX(IP1).LT.0) THEN                                              347100
      C(N)=C(N)-AREAX1*VX(IP1)                                           347200
      ELSE                                                               347300
      E(N)=E(N)-AREAX1*VX(IP1)                                           347400
      END IF                                                             347500
      ELSE                                                               347600
      VV=0.5*AREAX1*VX(IP1)                                              347700
      C(N)=C(N)-VV                                                       347800
      E(N)=E(N)-VV                                                       347900
      END IF                                                             348000
      END IF                                                             348100
```

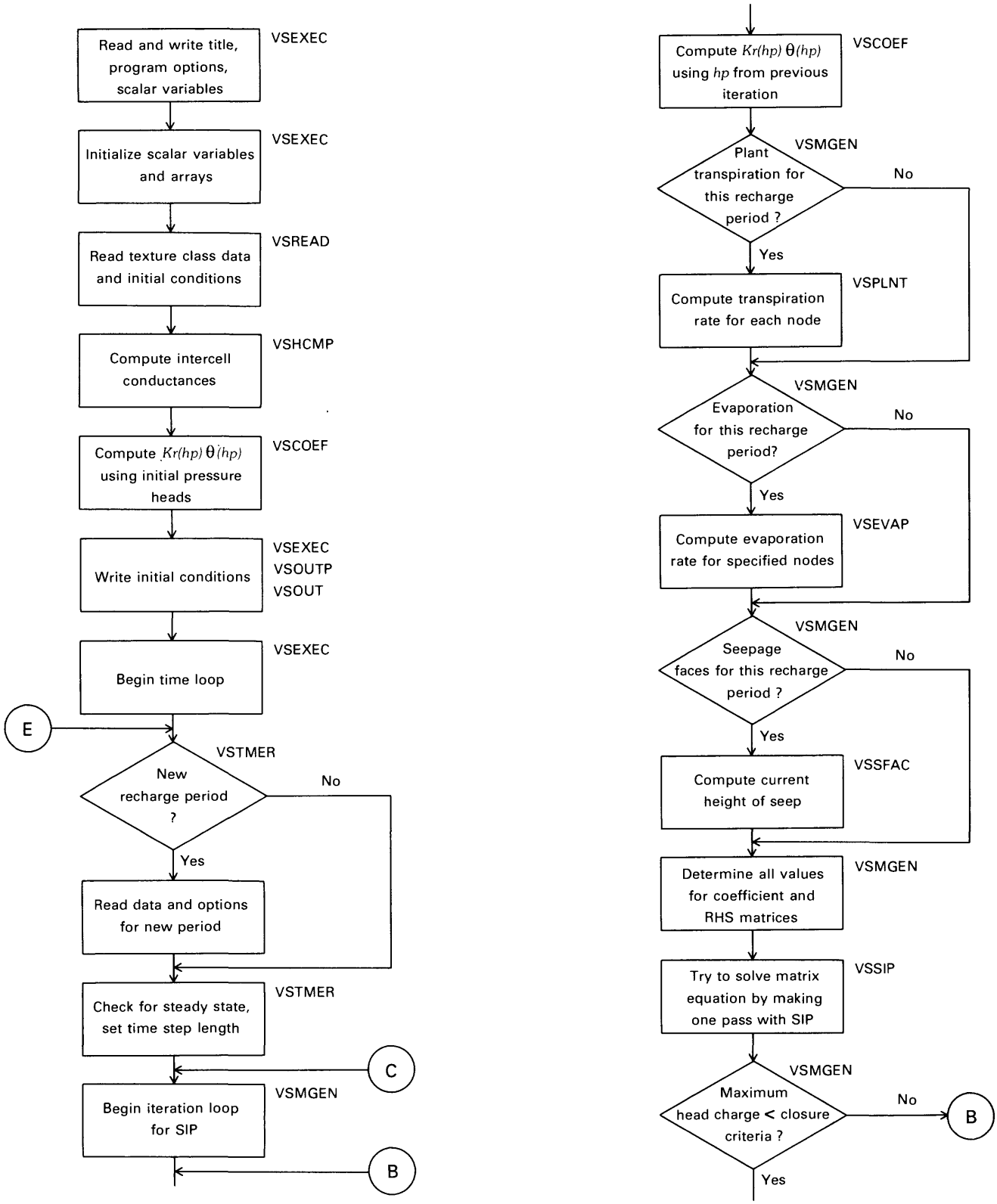
```
      IF(HX(JP1).NE.0) THEN                                              348200
      D(N)=DZ1(JP1)+0.5*(-DX2(N)+DX2(IP1))                               348300
      IF(.NOT.CIS) THEN                                                  348400
      IF(VZ(JP1).LT.0) THEN                                              348500
      D(N)=D(N)-AREAZ1*VZ(JP1)                                           348600
      ELSE                                                               348700
      E(N)=E(N)-AREAZ1*VZ(JP1)                                           348800
      END IF                                                             348900
      ELSE                                                               349000
      VV=0.5*AREAZ1*VZ(JP1)                                              349100
      D(N)=D(N)-VV                                                       349200
      E(N)=E(N)-VV                                                       349300
      END IF                                                             349400
      END IF                                                             349500
      IF(Q(N).LT.0..AND.NTYP(N).NE.5) E(N)=E(N)+Q(N)                     349600
      IF(QQ(N).LT.0.) E(N)=E(N)+QQ(N)                                    349700
      IF(QT(N).GT.0) E(N)=E(N)-QT(N)                                     349800
C                                                                        349900
C  CENTERED-IN-TIME DIFFERENCING CAN BE USED ONLY AFTER THE              350000
C  FIRST TIME STEP IN ANY RECHARGE PERIOD.                               350100
C                                                                        350200
      IF(CIT.AND.JFLAG1.NE.1) THEN                                       350300
      A(N)=0.5*A(N)                                                      350400
      B(N)=0.5*B(N)                                                      350500
      C(N)=0.5*C(N)                                                      350600
      D(N)=0.5*D(N)                                                      350700
      E(N)=0.5*E(N)                                                      350800
      END IF                                                             350900
      E(N)=E(N)-VOL*(THETA(N)+SS+RET(N))/DELT                            351000
      END IF                                                             351100
   20 CONTINUE                                                           351200
C                                                                        351300
C  BEGIN LOOP TO CALCULATE RHS AND CALL MATRIX SOLVER                    351400
C                                                                        351500
      DO 50 IT=1,ITMAX                                                   351600
      DO 30 I=2,NXRR                                                     351700
      N1=NLY*(I-1)                                                       351800
      DO 30 J=2,NLYY                                                     351900
      N=N1+J                                                             352000
      IM1=N-NLY                                                          352100
      JM1=N-1                                                            352200
      JP1=N+1                                                            352300
      IP1=N+NLY                                                          352400
      IP2=IP1-1                                                          352500
      IM2=IM1+1                                                          352600
      IM3=IM1-1                                                          352700
      IP3=IP1+1                                                          352800
      IF(RAD) THEN                                                       352900
      VOL=PI2*DELZ(J)*DXR(I)*RX(I)                                       353000
      ELSE                                                               353100
      VOL=DELZ(J)*DXR(I)                                                 353200
      END IF                                                             353300
      N2=JTEX(N)                                                         353400
      IF(SORP) THEN                                                      353500
      IF(IT.GT.1) THEN                                                   353600
C                                                                        353700
C   FOR NONLINEAR SORPTION RECALCULATE RET,E                             353800
C                                                                        353900
      RET1=RET(N)                                                        354000
      RET(N)=VTRET(CC(N),N2,HT)                                          354100
      IF(CIT.AND.JFLAG1.NE.1) THEN                                       354200
      T1=0.5                                                             354300
      ELSE                                                               354400
      T1=1.                                                              354500
      END IF                                                             354600
      E(N)=E(N)+VOL*(RET1-RET(N))*(1./DELT+HT(N2,4)*T1)                  354700
      END IF                                                             354800
```

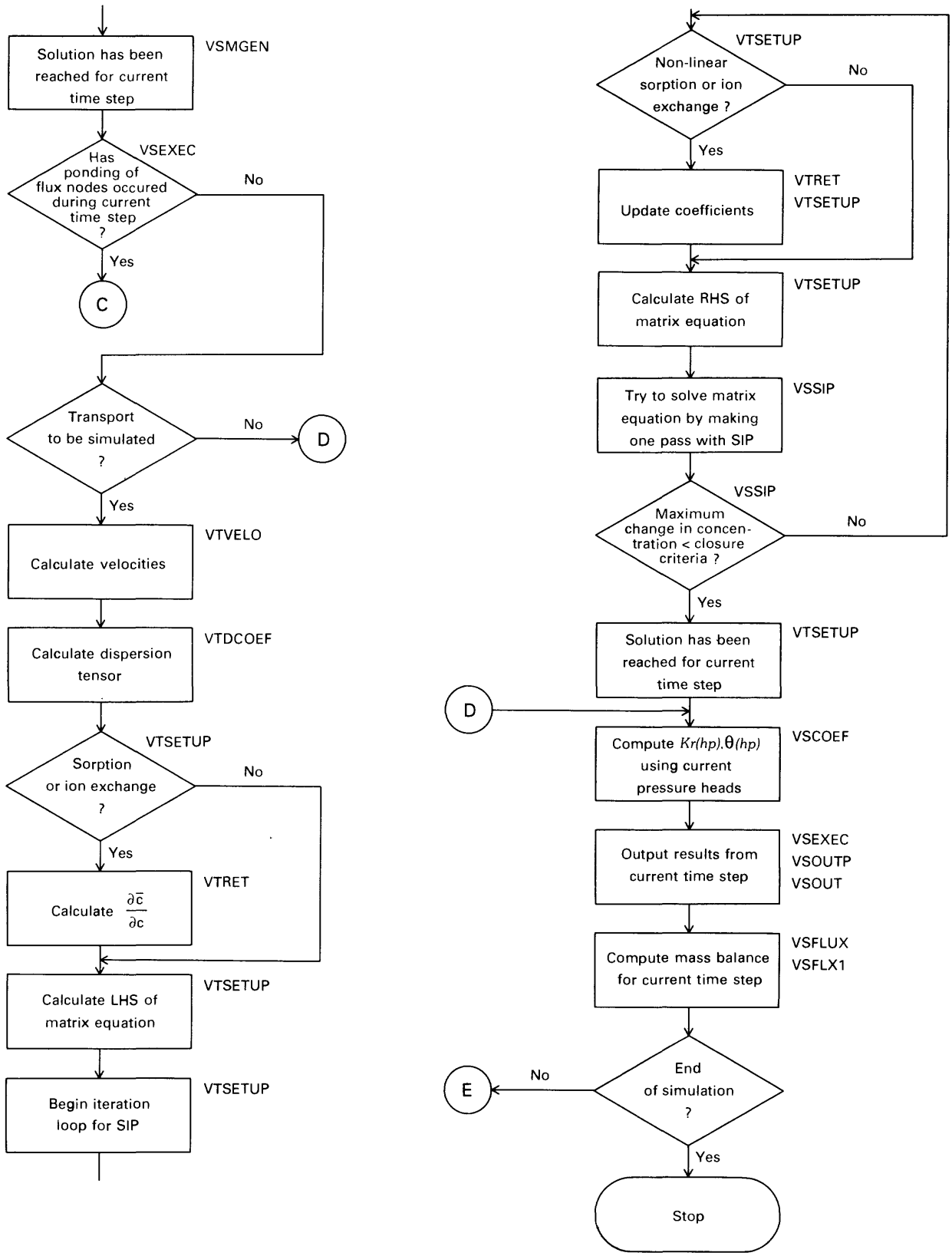
```
      END IF                                                            354900
C                                                                       355000
C   CALCULATE RHS OF MATRIX EQUATION                                    355100
C                                                                       355200
      RHS(N)=-VOL*(THETA(N)*(1+P(N)*HK(N2,2)/HK(N2,3))+RET(N))*COLD(N)/ 355300
     &DELT+0.5*(DX2(N)*(CC(IM2)-CC(IM3))+                               355400
     &DX2(IP1)*(CC(IP2)-CC(IP3))+DZ2(N)*(CC(IP2)-CC(IM3))               355500
     &+DZ2(JP1)*(CC(IM2)-CC(IP3)))-A(N)*CC(IM1)-B(N)*CC(JM1)            355600
     &-C(N)*CC(IP1)-D(N)*CC(JP1)-E(N)*CC(N)                             355700
      IF (CIT.AND.JFLAG1.NE.1) RHS(N)=RHS(N)-AO(N)*COLD(IM1)-BO(N)      355800
     **COLD(JM1)-CO(N)*COLD(IP1)-DO(N)*COLD(JP1)-EO(N)*COLD(N)          355900
      IF(QQ(N).GT.0.) RHS(N)=RHS(N)-QQ(N)*CS(N)                         356000
      IF(QT(N).LT.0..AND.NCTYP(N).EQ.0) RHS(N)=RHS(N)+QT(N)*CS(N)       356100
      IF(QT(N).LE.0.AND.NCTYP(N).EQ.2) RHS(N)=RHS(N)-CS(N)              356200
   30 CONTINUE                                                          356300
      NIT1=NIT1+1                                                       356400
C                                                                       356500
C   CALL MATRIX SOLVER                                                  356600
C                                                                       356700
      CALL SLVSIP                                                       356800
      IF(ITEST.EQ.0) THEN                                               356900
      IF (CIT) THEN                                                     357000
      DO 40 I=2,NXRR                                                    357100
      N1=NLY*(I-1)                                                      357200
      DO 40 J=2,NLYY                                                    357300
      N=N1+J                                                            357400
      IF(HX(N).EQ.0) GO TO 40                                           357500
      AO(N)=A(N)                                                        357600
      BO(N)=B(N)                                                        357700
      CO(N)=C(N)                                                        357800
      DO(N)=D(N)                                                        357900
      IF(RAD) THEN                                                      358000
      AREAZ=PI2*DXR(I)*RX(I)                                            358100
      ELSE                                                              358200
      AREAZ=DXR(I)                                                      358300
      END IF                                                            358400
      VOL=AREAZ*DELZ(J)                                                 358500
      N2=JTEX(N)                                                        358600
      SS=HK(N2,2)/HK(N2,3)                                              358700
      SS=THETA(N)*(1+(SS+SS)*P(N))-THLST(N)*(1+SS*PXXX(N))              358800
      EO(N)=E(N)+VOL*(THETA(N)+SS+RET(N))/DELT                          358900
   40 CONTINUE                                                          359000
      END IF                                                            359100
      JFLAG1=JFLAG                                                      359200
      RETURN                                                            359300
      END IF                                                            359400
   50 CONTINUE                                                          359500
      JFLAG1=JFLAG                                                      359600
      WRITE(6,4000)                                                     359700
      IF (.NOT.ITSTOP) RETURN                                           359800
      JSTOP=1                                                           359900
      JFLAG=1                                                           360000
      RETURN                                                            360100
 4000 FORMAT(' MAXIMUM NUMBER OF ITERATIONS EXCCEDED FOR TRANSPORT'     360200
     &' EQUATION')                                                      360300
      END                                                               360400
      DOUBLE PRECISION FUNCTION VTRET(P,I,HT)                           360500
C*****                                                                  360600
CVTRET                                                                  360700
C*****                                                                  360800
C                                                                       360900
C  SLOPE OF SORPTION ISOTHERM  --  LANGMUIR                             361000
C                                                                       361100
      IMPLICIT DOUBLE PRECISION (A-H,P-Z)                               361200
      DIMENSION HT(10,20)                                               361300
      VTRET=HT(I,5)*HT(I,6)*HT(I,7)/(1+HT(I,6)*P)**2                    361400
      RETURN                                                            361500
```

```
      END                                                                361600
C                                                                        361700
C                                                                        361800
C                                                                        361900
C  NOTE -- AS LISTED HERE THE PROGRAM USES THE VTRET FUNCTION            362000
C          ROUTINE FOR THE LANGMUIR ISOTHERM.                            362100
C          FUNCTIONS FOR THE 5 ALTERNATIVE VERSIONS OF VTRET ARE         362200
C          LISTED BELOW.  TO USE ONE OF THESE PLACE A 'C' IN             362300
C          COLUMN 1 OF ALL LINES IN THE LANGMUIR VERSION AND             362400
C          REMOVE THE COMMENT DESIGNATIONS FOR THE DESIRED               362500
C          VERSION OF VTRET -- 'CF' FREUNDLICH ISOTHERM                  362600
C                              'CM' MONO-MONOVALENT ION EXCHANGE         362700
C                              'CD' DIVALENT-DIVALENT ION EXCHANGE       362800
C                              'CE' MONO-DIVALENT ION EXCHANGE           362900
C                              'CG' DI-MONOVALENT ION EXCHANGE           363000
C                                                                        363100
C                                                                        363200
C                                                                        363300
CF    DOUBLE PRECISION FUNCTION VTRET(P,I,HT)                            363400
C*****                                                                   363500
CVTRET                                                                   363600
C*****                                                                   363700
C                                                                        363800
C  SLOPE OF SORPTION ISOTHERM  --  FREUNDLICH                            363900
C                                                                        364000
CF    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                364100
CF    DIMENSION HT(10,20)                                                364200
CF    IF(HT(I,6).EQ.0.) THEN                                             364300
CF    VTRET=0.D0                                                         364400
CF    RETURN                                                             364500
CF    END IF                                                             364600
CF    IF(HT(I,7).EQ.1.) THEN                                             364700
CF    VTRET=HT(I,5)*HT(I,6)                                              364800
CF    ELSE                                                               364900
CF    IF(P.EQ.0.) THEN                                                   365000
CF    VTRET=0.0                                                          365100
CF    ELSE                                                               365200
CF    VTRET=HT(I,5)*HT(I,6)*HT(I,7)*P**(HT(I,7)-1)                       365300
CF    END IF                                                             365400
CF    END IF                                                             365500
CF    RETURN                                                             365600
CF    END                                                                365700
CM    DOUBLE PRECISION FUNCTION VTRET(P,I,HT)                            365800
C*****                                                                   365900
CVTRET                                                                   366000
C*****                                                                   366100
C                                                                        366200
C  SLOPE OF SORPTION CURVE FOR                                           366300
C  MONOVALENT-MONOVALENT ION EXCHANGE                                    366400
C                                                                        366500
CM    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                366600
CM    DIMENSION HT(10,20)                                                366700
CM    VTRET=HT(I,5)*HT(I,6)*HT(I,7)*HT(I,8)/(P*(HT(I,6)-1)               366800
CM   1+HT(I,8))**2                                                       366900
CM    RETURN                                                             367000
CM    END                                                                367100
CD    DOUBLE PRECISION FUNCTION VTRET(P,I,HT)                            367200
C*****                                                                   367300
CVTRET                                                                   367400
C*****                                                                   367500
C                                                                        367600
C  SLOPE OF SORPTION CURVE FOR                                           367700
C  DIVALENT-DIVALENT ION EXCHANGE                                        367800
C                                                                        367900
CD    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                368000
CD    DIMENSION HT(10,20)                                                368100
CD    VTRET=HT(I,5)*HT(I,6)*HT(I,7)*HT(I,8)/((P+P)*                      368200
```

```
CD   1(HT(I,6)-1)+HT(I,8))**2                                                  368300
CD    RETURN                                                                  368400
CD    END                                                                     368500
CE    DOUBLE PRECISION FUNCTION VTRET(P,I,HT)                                 368600
C*****                                                                        368700
CVTRET                                                                        368800
C*****                                                                        368900
C                                                                             369000
C  SLOPE OF SORPTION CURVE FOR                                                369100
C  MONOVALENT-DIVALENT ION EXCHANGE                                           369200
C                                                                             369300
CE    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                     369400
CE    DIMENSION HT(10,20)                                                     369500
CE    IF(P.LE.0) THEN                                                         369600
CE    VTRET=0.                                                                369700
CE    ELSE                                                                    369800
CE    P1=P*P                                                                  369900
CE    P2=P1*HT(I,6)                                                           370000
CE    P3=HT(I,8)-P                                                            370100
CE    P3=P3+P3                                                                370200
CE    P4=HT(I,6)*(P+P)                                                        370300
CE    CB=(-P2+DSQRT(P2*P2+(P3+P3)*P2*HT(I,7)))/P3                             370400
CE    IF(CB.LT.HT(I,7))THEN                                                   370500
CE    VTRET=HT(I,5)*(CB*CB+P4*(HT(I,7)-CB))/(P3*CB+P2)                        370600
CE    ELSE                                                                    370700
CE    VTRET=HT(I,5)*(HT(I,7)*HT(I,7))/(P3*HT(I,7)+P2)                         370800
CE    END IF                                                                  370900
CE    END IF                                                                  371000
CE    RETURN                                                                  371100
CE    END                                                                     371200
CG    DOUBLE PRECISION FUNCTION VTRET(P,I,HT)                                 371300
C*****                                                                        371400
CVTRET                                                                        371500
C*****                                                                        371600
C                                                                             371700
C  SLOPE OF SORPTION CURVE FOR                                                371800
C  DIVALENT-MONOVALENT ION EXCHANGE                                           371900
C                                                                             372000
CG    IMPLICIT DOUBLE PRECISION (A-H,P-Z)                                     372100
CG    DIMENSION HT(10,20)                                                     372200
CG    IF(P.LE.0.) THEN                                                        372300
CG    VTRET=0.                                                                372400
CG    ELSE                                                                    372500
CG    IF((P+P).GE.HT(I,8)) THEN                                               372600
CG    VTRET=0.00                                                              372700
CG    ELSE                                                                    372800
CG    P1=P*HT(I,6)                                                            372900
CG    P2=P1+P1+P1+P1                                                          373000
CG    P4=HT(I,8)-P-P                                                          373100
CG    P5=P4*P4                                                                373200
CG    P6=HT(I,7)**2                                                           373300
CG    P3=-P2*HT(I,7)-P5                                                       373400
CG    P7=P3*P3-4*P2*P1*P6                                                     373500
CG    IF (P7.GT.0) THEN                                                       373600
CG    CB=(-P3-DSQRT(P7))/(P2+P2)                                              373700
CG    ELSE                                                                    373800
CG    CB=0.                                                                   373900
CG    END IF                                                                  374000
CG    VTRET=HT(I,5)*(-CB*CB*4*HT(I,6)+4*CB*(HT(I,6)*HT(I,7)-P4)-HT(I,6)       374100
CG   1*P6)/(P2*(CB+CB-HT(I,7))-P5)                                            374200
CG    END IF                                                                  374300
CG    END IF                                                                  374400
CG    RETURN                                                                  374500
CG    END                                                                     374600
```

PROGRAM FLOW CHART




Solution has been reached for current time step
VSMGEN
Has ponding of flux nodes occured during current time step ?
VSEXEC
No
Yes
C
Transport to be simulated ?
No
D
Yes
Calculate velocities
VTVELO
Calculate dispersion tensor
VTDCOEF
Sorption or ion exchange ?
VTSETUP
No
Yes
Calculate $\frac{\partial \bar{c}}{\partial c}$
VTRET
Calculate LHS of matrix equation
VTSETUP
Begin iteration loop for SIP
VTSETUP
Non-linear sorption or ion exchange ?
VTSETUP
No
Yes
Update coefficients
VTRET
VTSETUP
Calculate RHS of matrix equation
VTSETUP
Try to solve matrix equation by making one pass with SIP
VSSIP
Maximum change in concentration < closure criteria ?
VSSIP
No
Yes
Solution has been reached for current time step
VTSETUP
D
Compute $Kr(hp).\theta(hp)$ using current pressure heads
VSCOEF
Output results from current time step
VSEXEC
VSOUTP
VSOUT
Compute mass balance for current time step
VSFLUX
VSFLX1
End of simulation ?
No
E
Yes
Stop