FEMFLOW3D: A Finite-Element Program for the Simulation of Three-Dimensional Aquifers. Version 1.0

By T. J. Durbin and L.D. Bond

U.S. GEOLOGICAL SURVEY

Open-File Report 97-810

Sacramento, California
1998

U.S. DEPARTMENT OF THE INTERIOR
BRUCE BABBITT, Secretary

U.S. GEOLOGICAL SURVEY
Thomas J. Casadevall, Acting Director

For additional information write to:

District Chief
U.S. Geological Survey
Placer Hall
6000 J Street
Sacramento, CA 95819-6129

Copies of this report can be purchased from:

U.S. Geological Survey
Information Services
Box 25286
Federal Center
Denver, CO 80225

CONTENTS

FIGURES

TABLES

FEMFLOW3D: A FINITE-ELEMENT PROGRAM FOR THE SIMULATION OF THREE-DIMENSIONAL AQUIFERS. Version 1.0

By T.J. Durbin *and* L.D. Bond

Abstract

This document describes a computer program that simulates three-dimensional ground-water systems using the finite-element method. The program was developed to simulate regional ground-water systems, but it can be applied to small-scale problems as well. This program can be used to simulate both confined and water-table aquifers.

The program simulates a linearized three-dimensional free-surface ground-water system with a fixed grid. *FEMFLOW3D* is applicable to the simulation of various free-surface ground-water systems for which the change in aquifer thickness is small relative to the overall aquifer thickness.

The finite-element method provides flexibility in the design of a geometric grid that represents the physical dimensions of an aquifer system. For example, features that can be well represented with a finite-element grid include irregular, random geographic and geologic features; irregular boundaries; and increased detail within localized areas of particular interest within the study area.

The structure of the computer program consists of a main program, which serves as a simple driver, and a set of subroutines in which the calculations are performed. The background, mathematical basis, structure, and inputs for each of the subroutines are described in the document where applicable. Each subroutine generally handles (1) a part of the mathematical calculations related to the finite-element method, (2) a specific feature of the hydrologic system, or (3) special features related to the management input or output data.

Hydrologic features that can be represented with the program include stream-aquifer interactions, phreatophytic evapotranspiration, highly permeable fault zones, land subsidence, and land-aquifer interactions associated with land-use activities. The program can also represent the primary features associated with complex irrigation systems, such as irrigated agriculture, and can calculate the ground-water recharge that results from these activities. Three boundary conditions, including specified-head boundaries, specified-flux boundaries, and variable-flux boundaries, can be represented with the program. The program also provides a method for identifying aquifer and river-bed parameters that can be used in the calibration of models.

This document also includes model validation, source code, and example input and output files. Model validation was performed using four test problems. For each test problem, the results of a model simulation with *FEMFLOW3D* were compared with either an analytic solution or the results of an independent numerical approach. The source code, written in the ANSI x3.9-1978 FORTRAN standard, and the complete input and output of an example problem are listed in the appendixes.

1.0 INTRODUCTION

This document describes the computer program *FEMFLOW3D*, which is a finite-element program for the simulation of three-dimensional aquifers. The document is divided into three principal sections: Section 3.0, which describes the mathematical basis of the program; Section 4.0, which describes simulations that have been made with *FEMFLOW3D* for problems that have a known solution and describes the comparison of the simulations with the known solution; and Section 5.0, which describes the input formats for the program *FEMFLOW3D*. The mathematical descriptions in Section 3.0 are more detailed than some readers may require, but these descriptions are necessary for the complete documentation of *FEMFLOW3D*. Most readers can skip Section 3.0 and can proceed directly to Section 5.0, which tells how to prepare the input files for *FEMFLOW3D*.

The program *FEMFLOW3D* simulates the three-dimensional flow of ground water. The program was developed to simulate regional ground-water systems but can also be applied to small-scale problems. The program can be used to simulate confined or water-table aquifers for problems at regional or smaller scales.

FEMFLOW3D uses an iterative procedure to simulate the hydraulic response of a three-dimensional water-table aquifer. However, the simulation of a water-table aquifer has been done by others using other approaches. The most commonly applied approach represents a ground-water system as a quasi three-dimensional system of two-dimensional aquifers connected hydraulically by leakage through interaquifer aquitards. To solve this problem, Bredehoeft and Pinder (1970) used the finite-difference method and Durbin (1978) used the finite-element method. Finite-difference algorithms for simulated quasi three-dimensional ground-water flow are documented by Trescott (1975) and Trescott and Larson (1977). In each of these methods, the free surface is represented by applying the Dupuit assumptions to the upper layer of a model. For the upper layer, transmissivity is a function of calculated head, and the storage coefficient is set to a value representing specific yield at the free surface.

In some fully three-dimensional representations of a ground-water system, the effect of the free surface can be approximated by keeping the geometry of the flow domain fixed while assigning a storage-coefficient value representing specific yield to the upper parts of the model grid. Using this approach, the large storage effect at the free surface is represented, but the nonlinear effects of the changed saturated thickness are not. Another approach uses fixed geometry to represent the free-surface problem as a saturated/unsaturated flow simulation. Freeze (1971) used the fixed geometry approach within a finite-difference approximation, and Frind and Verge (1978) used it within a finite-element approximation.

The effects of geometry changes in the flow domain can be represented by using a deforming model grid. Leake (1977) used this approach by eliminating or adding blocks to a finite-difference grid as the calculated heads within the aquifer changed with time. Leake demonstrated his algorithm for the two-dimensional x-z case, but the approach can be extended to the three-dimensional case. By using this approach, the effects of storage changes at the free surface and the deformation of the flow domain are included. However, a more rigorous approach was taken by Neuman and Witherspoon (1971). Starting with the governing equation for ground-water flow and with a linearized partial differential equation describing the free-surface boundary condition, Neuman and Witherspoon developed a finite-element algorithm for the two-dimensional x-z case that uses a deforming grid.

The finite-element program *FEMFLOW3D* is based on the application of the work of Neuman and Witherspoon (1971) to the three-dimensional case with a fixed grid. Thus, the model simulates a linearized three-dimensional free-surface ground-water system. Neuman and Witherspoon (1971) uses this system in their analytical solution of the problem of pumping from a partially penetrating well within a three-dimensional free-surface aquifer. The use of a fixed grid within *FEMFLOW3D* disregards the effects of a change in aquifer thickness on ground-water flow. However, *FEMFLOW3D* does represent the effects of a free-surface boundary condition. For most free-surface ground-water systems, the free-surface effect is a more important phenomenon than is the saturated-thickness effect. Accordingly, *FEMFLOW3D* is applicable to the simulation of various free-surface ground-water systems, and it is particularly applicable to ground-water systems for which the change in aquifer thickness is small relative to the overall aquifer thickness.

2.0 GENERAL FEATURES OF *FEMFLOW3D*

The finite-element program *FEMFLOW3D* has several general features that aid the three-dimensional simulation of either regional or small-scale ground-water systems. These features include

1. Grid specification. Specification of the finite-element grid is done in terms of triangular prisms instead of tetrahedrons, although the internal model calculations are done using tetrahedral elements. Additionally, triangular prisms with zero-height edges can be used to represent particular hydrogeologic units that pinch out or can be used to make geographic changes in the vertical discretization of the grid.

2. Specified-head boundaries. Specified-head boundary conditions can be imposed with boundary heads that are either constant with time or vary with time. In the latter case, the boundary heads are specified in terms of tables representing the hydrograph of the boundary heads. Additionally, drainage nodes can be specified that allow discharge from the modeled flow domain, but that do not allow flow into the domain.

3. Specified-flux boundaries and internal source-sink terms. The specified-flux boundary conditions and internal sources-sink terms are defined in terms of a group of various data sets that can be combined in different configurations for each time step.

4. Variable-flux boundaries. The variable-flux boundary condition can be imposed to simulate time-variant boundary fluxes in response to changing boundary heads. This is a boundary condition that allows the finite volume of the modeled flow domain to be extended to infinity by attaching the analytical solution for a semi-infinite linear aquifer to the boundary of the flow domain.

5. River-aquifer simulation. To simulate river-aquifer interactions, riverflow is routed through a main channel and its tributaries. The exchange of water between the local river-channel reach and the ground-water system depends on the wetted width of the reach, flow depth, river-bed elevation, river-bed thickness, and river-bed hydraulic conductivity. Additionally, when water

seeps from the river reach to the ground-water system, the simulation allows a break in the hydraulic connection between the river and the ground-water system. The break occurs where the depth to the water table below the elevation of the river bed is greater than the thickness of the river bed. For this condition, the rate of ground-water recharge from the river is independent of the depth to the water table. This condition occurs when the hydraulic gradient from the river to the water table reaches unity in the vertical direction.

6. Phreatophyte simulation. The evapotranspiration of ground water from a shallow water table owing to phreatophytes can be simulated by specifying the maximum evapotranspiration rate and the extinction depth.

7. Fault-zone simulation. The effect of highly transmissive fault zones can be simulated without a representation in the finite-element grid, but poorly transmissive fault zones can be simulated only by an explicit representation in the finite-element grid. The representation of highly transmissive fault zones is done by specifying linkages between particular node pairs that allow the movement of water along the links, in addition to the movement of water through the aquifer continuum. These linkages can also be used as a convenient method for specifying the hydraulic condition in a well that pumps from multiple layers in the finite-element grid. The linkages distribute the discharge from individual layers to maintain nearly identical hydraulic heads in each layer, which represents the condition in a well casing where hydraulic heads typically are nearly uniform throughout the water column within the casing.

8. Land-subsidence simulation. The subsidence of the land surface as the result of aquifer compaction is simulated by specifying the elastic and inelastic specific storage of the fine-grained beds and by specifying the initial preconsolidation heads within the ground-water system. Accordingly, the elastic and inelastic release of stored water and the elastic uptake of water are simulated; correspondingly, the cumulative change in storage within an aquifer column is used to determine the cumulative change in the elevation of the land surface.

9. Water-use specification. Ground-water pumpage and recharge for an irrigated-agriculture system can conveniently be simulated. Input files consist of a data-management system for the storage and the retrieval of well-site data, ground-water pumpage data, irrigated-area data, crop-inventory data, and surface-water delivery data. The data is processed by an algorithm that calculates the consumption of water by crops and the recharge of ground-water that results from precipitation and the application of irrigation water. Data can be organized on the basis of geographic or political boundaries and are not restricted by the layout of the model grid.

10. Parameter estimation. The aquifer parameters for the model can be estimated using a weighted least-squared procedure. This procedure uses information on the expected value and variance of the expected value for hydraulic conductivity, specific storage, and specific yield, along with information on the measured ground-water levels and the variance of the measurements, to estimate the maximum likelihood values of hydraulic conductivity, specific storage, and specific yield.

3.0 DESCRIPTION OF *FEMFLOW3D*

The finite-element program *FEMFLOW3D* consists of the main program and a set of subroutines (fig. 1). The main program is a simple driver; the various subroutines perform the model calculations. The primary calculations are done in the subroutine *MODEL*. The FORTRAN source code for *FEMFLOW3D* is on the diskette in the pocket at the back of the report. The files that contain the source code have the extension .FOR. The source code for the program *FEMFLOW3D* is written in the ANSI x3.9-1978 FORTRAN standard. Two compiler specifications must be included when compiling the FORTRAN source code: (1) The compiler must accept "ENTRY" statements that have a different argument list than the argument list for the main subroutine call, and (2) the internal variables within the subroutines must be saved.

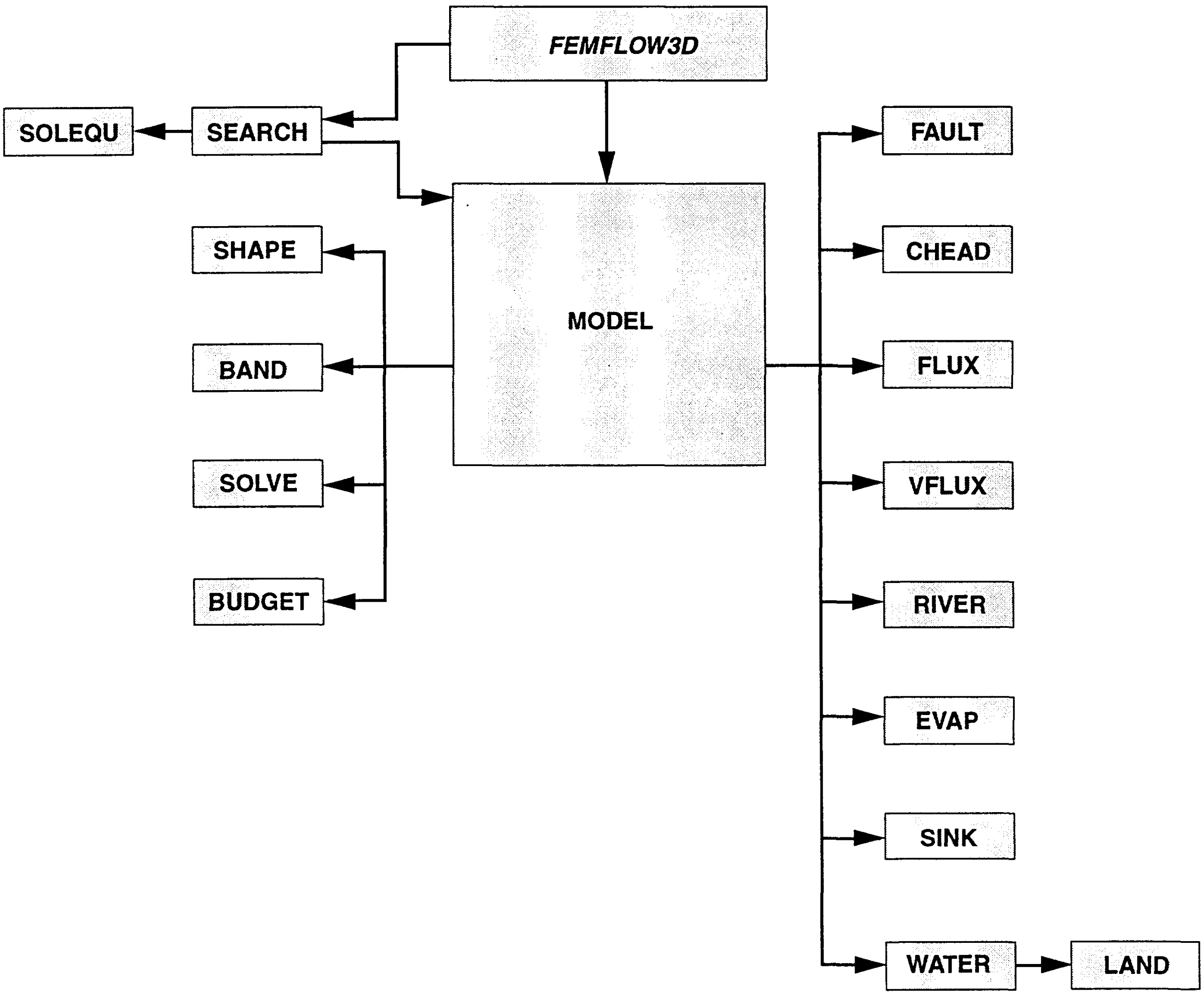


Figure 1. Structure of *FEMFLOW3D*.

3.1 Program *FEMFLOW3D*

3.1.1 Background

The finite-element program *FEMFLOW3D* derives its name from its main program called *FEMFLOW3D.* This main program is the driver program that makes calls only to the subroutines *MODEL* and *SEARCH.* Unless specified as the driver program, references to *FEMFLOW3D* refer to the finite-element program, including all of its subroutines.

3.1.2 Mathematical Basis

No calculations are done in the driver program, *FEMFLOW3D.*

3.1.3 Structure of Subroutine

The driver program, *FEMFLOW3D,* executes two alternative sets of subroutine calls. A call is determined by parameter estimation. Subroutine *MODEL* is structured into three basic blocks, *MODEL1, MODEL2,* and *MODEL3.* If parameter identification is not done, the driver program calls *MODEL1* and *MODEL3* in sequence, and then stops. If parameter identification is done, the driver program calls *MODEL1* and *SEARCH* in sequence and then calls *MODEL2* from subroutine *SEARCH.*

3.2 SUBROUTINE *MODEL*

3.2.1 Background

Subroutine *MODEL* is the central module of *FEMFLOW3D.* This subroutine controls data input for *FEMFLOW3D* and executes the principal calculations relating to the implementation of the finite-element method. Data are input below the primary entry point *MODEL1* either directly as read statements or indirectly as calls to the primary entry points of other subroutines. The finite-element calculations are executed below the secondary entry point *MODEL2* or *MODEL3.* These calculations include calculating volume and surface integrations, assembling elemental and global matrices, and solving a system of linear equations that yields the calculated hydraulic head for each time step. The progression of the calculations through time is controlled below the entry points *MODEL2* or *MODEL3.*

3.2.2 Mathematical Basis

3.2.2.1 Governing Equation

The calculations in subroutine *MODEL* are based on the three-dimensional equation of ground-water flow in the form

$$\frac{\partial}{\partial x}\left(K_{xx}\frac{\partial h}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_{yy}\frac{\partial h}{\partial y}\right) + \frac{\partial}{\partial z}\left(K_{zz}\frac{\partial h}{\partial z}\right) - S_s\frac{\partial h}{\partial t} + W = 0, \qquad (3.2\text{-}1)$$

where

h is hydraulic head [L],
K_{xx} is the hydraulic conductivity in the x direction [L/t],
K_{yy} is the hydraulic conductivity in the y direction [L/t],
K_{zz} is the hydraulic conductivity in the z direction [L/t],
S_s is the specific storage [1/L],
W is the source injection rate per unit volume [L^3/t per L^3 or 1/t],
x is the coordinate in the x direction [L],
y is the coordinate in the y direction [L],
z is the coordinate in the z direction [L], which has the same datum as the hydraulic head, and
t is time [t].

The form of the governing equation assumes that the coordinate system is aligned with the principal components of the hydraulic conductivity tensor.

The governing equation is subject to the water-table boundary condition in the form

$$K_{xx} \frac{\partial h}{\partial x} n_x + K_{yy} \frac{\partial h}{\partial y} n_y + K_{zz} \frac{\partial h}{\partial z} n_z = - S_y \frac{\partial h_f}{\partial t} n_z \tag{3.2-2}$$

and

$$h_f(x,y,t) = h(x,y,z,t)|_{\Gamma_F} \tag{3.2-3}$$

where

n_x is the x component of the outward normal vector on the water-table surface Γ_F [dimensionless],
n_y is the y component [dimensionless],
n_z is the z component [dimensionless],
S_y is the specific yield [dimensionless],
h_f is the elevation of the water table [L],
Γ_F is the water-table surface [L^2],
t is time [t],

and the specific-flux boundary condition in the form

$$K_{xx} \frac{\partial h}{\partial x} n_x + K_{yy} \frac{\partial h}{\partial y} n_y + K_{zz} \frac{\partial h}{\partial z} n_z = q \tag{3.2-4}$$

where q is the inward discharge normal to the boundary surface [L/t].

3.2.2.2 Finite-Element Approximation

In *FEMFLOW3D*, Equation 3.2-1 is solved using the Galerkin finite-element method described by Durbin and Berenbrock (1985). In this method, the exact continuous solution to Equation 3.2-1 is replaced by an approximate piecewise continuous solution. This piecewise continuous solution is defined by coefficient values specified at nodes in the model grid. Solution values between the nodes are calculated using piecewise continuous interpolation or basis functions that depend on the coefficients and are defined over the elements in the grid. *FEMFLOW3D* uses tetrahedral elements and linear interpolation functions.

Application of the Galerkin finite-element method to the spatial domain results in a system of ordinary differential equations in time. To solve this system of equations for each time step, the time derivative is approximated using a first-order, implicit, finite-difference scheme. Corresponding to the implicit approximation of the time derivative, the coefficients of the system of equations generated by the finite-element method are evaluated for each time step at the end of the new time step.

The coefficients of the system of equations are dependent, in part, on calculated hydraulic heads, causing the system of ordinary differential equations to be nonlinear. Specifically, nonlinearity results from (1) the use of the drainage node form of the specified-head boundary condition, (2) the representation of river-aquifer interactions, (3) the representation of evapotranspiration from ground water, and (4) the simulation of land subsidence. Because the nonlinearities are not severe, a solution to the nonlinear system of ordinary differential equations can be obtained by simple iteration, which is sometimes referred to as Picard iteration. At each iteration, a system of linear equations is generated using the hydraulic heads from the last iteration to evaluate the coefficients of the system of equations.

<u>Application of Galerkin method.</u>--To apply the finite-element method, the linear operator *L( )* from Equation 3.2-1 is defined as

$$L(h') \equiv \frac{\partial}{\partial x}\left(K_{xx}\frac{\partial h'}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_{yy}\frac{\partial h'}{\partial y}\right) + \frac{\partial}{\partial z}\left(K_{zz}\frac{\partial h'}{\partial z}\right) - S_s\frac{\partial h'}{\partial t} + W = 0. \quad (3.2\text{-}5)$$

To solve $L(h) = O$, interpolating function is used in the form (Pinder and Gray, 1977)

$$h'(x,y,z,t) = \sum_{i=1}^{n} H_i(t)\phi_i(x,y,z), \tag{3.2-6}$$

where

- h' is a series approximation to h [L],
- H_i are the undetermined coefficients [L],
- ϕ_i are linearly independent interpolating functions defined over the flow domain Ω [dimensionless],
- n is the number of nodal points [dimensionless], and
- Ω is the three-dimensional flow domain [L^3].

The series approximation to Equation 3.2-5 will provide an exact representation as n approaches infinity. For a finite series, however, the approximation will not satisfy exactly Equation 3.2-5 and a residual R will result. The residual is defined by

$$R(x,y,z,t) = L\left[\sum_{i=1}^{n} H_i(t)\phi_i(x,y,z)\right]. \tag{3.2-7}$$

If the trial solution was exact, the residual would vanish. For the trial solution, however, the average residual within the domain is forced to zero, in an average sense, through the selection of the undetermined coefficients H_i.

The undetermined coefficients are calculated by setting the weighted integrals of the residual to zero. In the Galerkin method (Pinder and Gray, 1977), the interpolating functions are used as weighting functions, that is, the inner product of the residual with each linearly interpolating function is set to zero, or

$$\int_{\Omega} L\left[\sum_{j=1}^{n} H_j(t)\phi_j(x,y,z)\right]\phi_i(x,y,z)d\Omega = 0 \tag{3.2-8}$$

$$for\ i = 1,2,\ldots,n.$$

Using Equation 3.2-8 n equations can be solved for the n values of H_j.

System of ordinary differential equations.--To solve the system of ordinary differential equations, Equation 3.2-8 can be simplified. First, Equation 3.2-8 is expanded to obtain the system of n equations

$$\begin{aligned}\int_{\Omega} \Bigg[& \frac{\partial}{\partial x}\left(K_{xx}\frac{\partial}{\partial x}\sum_{j=1}^{n} H_j\phi_j\right) \\ & + \frac{\partial}{\partial y}\left(K_{yy}\frac{\partial}{\partial y}\sum_{j=1}^{n} H_j\phi_j\right) \\ & + \frac{\partial}{\partial z}\left(K_{zz}\frac{\partial}{\partial z}\sum_{j=1}^{n} H_j\phi_j\right) \\ & - S_s\frac{\partial h'}{\partial t} + W \Bigg] \phi_i d\Omega = 0 \\ & for\ i = 1,2,\ldots,n. \end{aligned} \tag{3.2-9}$$

Second, the second-order terms in Equation 3.2-9 are eliminated by applying Green's theorem (Pinder and Gray, 1977). By assuming that hydraulic conductivity is constant for each element and by recalling that H_j is a function of time only, the application of Green's theorem yields

$$\begin{aligned}& \int_{\Omega}\sum_{j=1}^{n}\left(K_{xx}\frac{\partial\phi_i}{\partial x}\frac{\partial\phi_j}{\partial x} + K_{yy}\frac{\partial\phi_i}{\partial y}\frac{\partial\phi_j}{\partial y} + K_{zz}\frac{\partial\phi_i}{\partial z}\frac{\partial\phi_j}{\partial z}\right)H_j\, d\Omega \\ & + \int_{\Omega} S_s\phi_i\frac{\partial h'}{\partial t}d\Omega - \int_{\Omega} W\phi_i d\Omega \\ & - \int_{\Gamma_F}\left(K_{xx}\frac{\partial h'}{\partial x}n_x + K_{yy}\frac{\partial h'}{\partial y}n_y + K_{zz}\frac{\partial h'}{\partial z}n_z\right)\phi_i d\Gamma \\ & - \int_{\Gamma_R}\left(K_{xx}\frac{\partial h'}{\partial x}n_x + K_{yy}\frac{\partial h'}{\partial y}n_y + K_{zz}\frac{\partial h'}{\partial z}n_z\right)\phi_i d\Gamma = 0 \\ & for\ i = 1,2,\ldots,n, \end{aligned} \tag{3.2-10}$$

where

Γ is the overall boundary surface of the flow domain Ω [L^2],
Γ_F is the free-surface part of the boundary surface [L^2], and
Γ_R is the remaining part of the boundary surface [L^2].

The integral over the surface Γ_F represents the free-surface conditions on that boundary, and the integral over the surface Γ_R represents a specific-flux condition on that boundary.

Third, by substituting Equations 3.2-2 and 3.2.3 into the last term of Equation 3.2-10 and by replacing the partial derivatives of h with the general definition of the total derivative, the result is

$$\sum_{j=1}^{n} \int_{\Omega} \left(K_{xx} \frac{\partial \phi_i}{\partial x} \frac{\partial \phi_j}{\partial x} + K_{yy} \frac{\partial \phi_i}{\partial y} \frac{\partial \phi_j}{\partial y} + K_{zz} \frac{\partial \phi_i}{\partial z} \frac{\partial \phi_j}{\partial z} \right) H_j d\Omega + \sum_{j=1}^{n} \int_{\Omega} S_s \phi_i \phi_j \frac{dH_j}{dt} d\Omega - \int_{\Omega} W \phi_i d\Omega - \int_{\Gamma_R} q \phi_i d\Gamma + \sum_{j=1}^{n} \int_{\Gamma_F} S_y \phi_i \phi_j \frac{dH_j}{dt} n_z d\Gamma = 0 \quad \text{for } i = 1,2,\ldots,n, \tag{3.2-11}$$

where

q is the inward discharge normal to the surface Γ [L/t],
Γ is the overall boundary surface of the flow domain Ω [L^2], and
Γ_F is the free-surface part of the boundary surface [L^2].

In obtaining Equation 3.2-11 from 3.2-10, the boundary integral represented by the last term of Equation 3.2-10 is partitioned in the model domain for water-table surface.

The n equations of Equation 3.2-11 can be written in matrix form as

$$[A]\{H\} + [B]\left\{\frac{dH}{dt}\right\} - \{F\} = O. \tag{3.2-12}$$

The typical elements of the matrices $[A]$ and $[B]$ and the vector $\{F\}$ are

$$A_{ij} = \int_{\Omega}\left(K_{xx}\frac{\partial\phi_i}{\partial x}\frac{\partial\phi_j}{\partial x} + K_{yy}\frac{\partial\phi_i}{\partial y}\frac{\partial\phi_j}{\partial y} + K_{zz}\frac{\partial\phi_i}{\partial z}\frac{\partial\phi_j}{\partial z}\right)d\Omega, \tag{3.2-13}$$

$$B_{ij} = \int_{\Omega} S_s\phi_i\phi_j d\Omega + \int_{\Gamma_F} S_y\phi_i\phi_j d\Gamma, \tag{3.2-14}$$

and

$$F_i = \int_{\Omega} W\phi_i d\Omega + \int_{\Gamma_R} q\phi_i d\Gamma, \tag{3.2-15}$$

which are referred to as the conductance matrix, the storage matrix, and the force vector, respectively.

3.2.2.3 Interpolating Function

To generate the set of algebraic equations represented by Equation 3.2-12, integrations of the interpolating functions must be carried out in the form

$$\int \frac{\partial \phi_i}{\partial x} \frac{\partial \phi_j}{\partial x} d\Omega,$$

$$\int \frac{\partial \phi_i}{\partial y} \frac{\partial \phi_j}{\partial y} d\Omega,$$

$$\int \frac{\partial \phi_i}{\partial z} \frac{\partial \phi_j}{\partial z} d\Omega,$$

$$\int \phi_i \phi_j d\Omega,$$

$$\int \phi_i d\Omega,$$

$$\int \phi_i \phi_j d\Gamma,$$

and

$$\int \phi_i d\Gamma.$$

To facilitate these integrations, the interpolating functions are defined separately for each element, but when combined produce the global-interpolating functions within the flow domain. The elemental interpolating functions used in this work are linear and are defined for tetrahedral elements within the interior of the flow domain and for triangular elements on the boundary surfaces of the flow domain.

<u>Volume integrations</u>.--Within a tetrahedral element, the trial solution (Equation 3.2-6) can be expressed as

$$h' = \sum_{i=1}^{4} H_i \phi_i^e \,, \tag{3.2-16}$$

where ϕ_i^e represent the elemental interpolating functions defined only within the element *e*. The interpolating functions for the node *i* are given by the relation (Zienkiewicz, 1977)

$$\phi_i^e = \frac{1}{6V}(a_i + b_i x + c_i y + d_i z), \tag{3.2-17}$$

where the coefficients of Equation 3.2-17 are given by the determinants

$$a_i = \begin{vmatrix} x_j & y_j & z_j \\ x_m & y_m & z_m \\ x_p & y_p & z_p \end{vmatrix}, \tag{3.2-18}$$

$$b_i = -\begin{vmatrix} 1 & y_j & z_j \\ 1 & y_m & z_m \\ 1 & y_p & z_p \end{vmatrix}, \tag{3.2-19}$$

$$c_i = -\begin{vmatrix} x_j & 1 & z_j \\ x_m & 1 & z_m \\ x_p & 1 & z_p \end{vmatrix}, \tag{3.2-20}$$

$$d_i = -\begin{vmatrix} x_j & y_j & 1 \\ x_m & y_m & 1 \\ x_p & y_p & 1 \end{vmatrix}, \tag{3.2-21}$$

$$6V = \begin{vmatrix} 1 & x_i & y_i & z_i \\ 1 & x_j & y_j & z_j \\ 1 & x_m & y_m & z_m \\ 1 & x_p & y_p & z_p \end{vmatrix}, \tag{3.2-22}$$

and where V is the volume of a tetrahedral element [L^3]. The indexes p, i, j, and m are the nodal numbers for a tetrahedral element. The ordering of nodal numbers must follow the right-hand rule, that is, the first three nodes (p, i, and j) are numbered in a counterclockwise manner when viewed from the last (m).

Integrations using derivatives of interpolating functions from Equations 3.2-17 through 3.2-18 are given by the relations (Zienkiewicz, 1977)

$$\int_e K_{xx} \frac{\partial \phi_i^e}{\partial x} \frac{\partial \phi_j^e}{\partial x} d\Omega = \frac{K_{xx}}{36V} b_i b_j , \tag{3.2-23}$$

$$\int_e K_{yy} \frac{\partial \phi_i^e}{\partial y} \frac{\partial \phi_j^e}{\partial y} d\Omega = \frac{K_{yy}}{36V} c_i c_j , \tag{3.2-24}$$

and

$$\int_e K_{zz} \frac{\partial \phi_i^e}{\partial z} \frac{\partial \phi_j^e}{\partial z} d\Omega = \frac{K_{zz}}{36V} d_i d_j . \tag{3.2-25}$$

Finally,

$$A_{ij}^e = \frac{1}{36V} (K_{xx} b_i b_j + K_{yy} c_i c_j + K_{zz} d_i d_j) , \tag{3.2-26}$$

where K_{xx}, K_{yy}, and K_{zz} are assumed constant for an element, and A_{ij}^e is the elemental contribution to matrix $[A^e]$ for i = 1,2,3,4 and j = 1,2,3,4 locally.

The matrix $[A]$ is obtained by summing the contribution from each node for each element matrix $[A^e]$. For example, if nodes i and j in the element nodal system correspond to nodes p and q in the global nodal system, the A_{ij}^e in the element stiffness matrix is added to A_{pq} in the global stiffness matrix. This procedure is repeated for each node in an element and for all elements in the domain Ω.

Integrations with only the interpolating function, and not their derivatives, are given by the relations (Zienkiewicz, 1977)

$$B_{ij}^e = \int_e S_s \phi_i^e \phi_j^e d\Omega = \frac{S_s V}{20} \quad for\ i \neq j \tag{3.2-27}$$

and

$$B_{ij}^e = \int_e S_s \phi_i^e \phi_j^e d\Omega = \frac{S_s V}{10} \quad for\ i = j, \tag{3.2-28}$$

where S_s is assumed constant for an element and B_{ij}^e is the elemental contribution to the matrix $[B^e]$ for i = 1,2,3,4 and j = 1,2,3,4 locally. The matrix $[B]$ is obtained by summing the contributions from each elemental matrix $[B^e]$, as described above for matrices $[A]$ and $[A^e]$.

Surface integrations.--On the free surface, it is necessary to perform integrations of the interpolating functions in the form

$$\int_e \phi_i^e \phi_j^e d\Gamma,$$

where the two-dimensional functions are now defined on the free surface. Where the flow domain is defined by an assemblage of tetrahedral elements, the free surface (and other boundary surfaces) can be defined by selected triangular faces of those tetrahedral elements that form the free surface. In other words, the free surface is defined by an assemblage of triangular elements representing the faces of those tetrahedral elements. Furthermore, if the free surface is assumed to be approximately horizontal, then the integrations can be carried out in the x-y plane according to the description below.

Within a triangular element, the approximate solution (Equation 3.2-6) can be expressed as

$$h' = \sum_{i=1}^{3} H_i \phi_i^e, \tag{3.2-29}$$

where ϕ_i^e represent the elemental interpolating functions defined only within the element e. The interpolating function for the node i is given by the relation (Zienkiewicz, 1977)

$$\phi_i^e = \frac{1}{2A} (a_i + b_i x + c_i y), \tag{3.2-30}$$

where

$$a_i = \begin{vmatrix} x_j & y_j \\ x_m & y_m \end{vmatrix}, \tag{3.2-31}$$

$$b_i = - \begin{vmatrix} 1 & y_j \\ 1 & y_m \end{vmatrix}, \tag{3.2-32}$$

$$c_i = - \begin{vmatrix} x_j & 1 \\ x_m & 1 \end{vmatrix}, \tag{3.2-33}$$

$$2A = \begin{vmatrix} 1 & x_i & y_i \\ 1 & x_j & y_j \\ 1 & x_m & y_m \end{vmatrix}, \tag{3.2-34}$$

and where A is the area of a triangular element [L^2]. The indexes i, j, and m are the nodal numbers for a triangular element. The ordering of nodal numbers must follow the right-hand rule, that is, the nodes are numbered in a counterclockwise direction when viewed from above the x-y plane.

Using Equations 3.2-30 through 3.2-34, integrations of the interpolating functions are given by the relations (Zienkiewicz, 1977)

$$B_{ij}^e = \int_e S_y \phi_i^e \phi_j^e \, d\Gamma = \frac{S_y}{12} A \quad for\ i \neq j \tag{3.2-35}$$

and

$$B_{ij}^e = \int_e S_y \phi_i^e \phi_j^e \, d\Gamma = \frac{S_y}{6} A \quad for\ i = j, \tag{3.2-36}$$

where S_y is assumed constant over an element and B_{ij}^e is the elemental contribution to matrix *[B]* for i = 1,2,3 and j = 1,2,3 locally. The matrix *[B]* is obtained by summing the contributions from each elemental matrix $[B^e]$, as described above for the matrices *[A]* and $[A^e]$.

3.2.2.4 Integration in Time

Although the matrices *[A]* and *[B]* and the vector *{F}* can now be evaluated, the system of ordinary differential equations must still be solved. To do this, the time derivative is approximated using the first-order, implicit, finite-difference scheme

$$[A]\{H_{t+\Delta t}\} + \frac{1}{\Delta t}[B]\{H_{t+\Delta t} - H_t\} - \{F\} = O, \tag{3.2-37}$$

which can be rearranged to obtain

$$\left([A] + \frac{1}{\Delta t}[B]\right)\{H_{t+\Delta t}\} = \frac{1}{\Delta t}[B]\{H_t\} + \{F\}, \tag{3.2-38}$$

where Δt is the time step [t]. By the implicit approximation of the time derivative, the matrices *[A]* and *[B]* and the vector *{F}* are evaluated at the new time step $t+\Delta t$.

3.2.2.5 Iterative Solution

Because the coefficients of the matrices *[A]* and *[B]* and the vector *{F}* are dependent, in part, on hydraulic head, Equation 3.2-38 represents a system of nonlinear algebraic equations. However, the nonlinearity introduced by head-dependent sources and sinks and by land subsidence is not severe. Therefore, the solution of Equation 3.2-38 can be obtained by a simple iterative procedure, in which, at the *k-th* iteration, Equation 3.2-38 takes the form

$$\left([A]^{(k-1)} + \frac{1}{\Delta t}[B]^{(k-1)}\right)\{H_{t+\Delta t}\}^{(k)} = \frac{1}{\Delta t}[B]^{(k-1)}\{H_t\} + \{F\}^{(k-1)}. \qquad (3.2\text{-}39)$$

At each iteration, the matrices *[A]* and *[B]* and the vector *{F}* are updated, and a solution is obtained the for new values of $\{H_{t+\Delta t}\}$ by the square-root or Cholesky method (Pinder and Gray, 1977). The process is repeated until the absolute difference between $\{H_{t+\Delta t}\}^{(k)}$ and $\{H_{t+\Delta t}\}^{(k-1)}$ at any particular node is less than some specified value. In most applications, convergence is obtained in two to five iterations.

3.2.3 Structure of Subroutine

3.2.3.1 General Structure

Subroutine *MODEL* is structured into two basic blocks. The first block, which is entered by a call to *MODEL1*, is for the input and display of data describing the ground-water problem to be solved. These data are entered by read statements within the *MODEL1* block and within the subroutine blocks *CHEAD1*, *FLUX1*, *RIVER1*, *EVAP1*, *VFLUX1*, *FAULT1*, *SINK1*, *WATER1*, and *SEARCH*, which are called in *MODEL1*. Within the first block, switches are set for the display of the input data and of the computational results.

Within the second block, which is entered by the call to *MODEL2*, the finite-element method is implemented (fig. 2). The finite-element calculations are organized into two computational loops. The outer loop is the time-step loop, and the inner loop is the iterative-solution loop, which relates to the solution method described in Section 3.2.2.5. For each time step, the iterative-solution loop is repeated until the solution converges. The closure criterion for convergence is that the calculated hydraulic heads change less than a specified value between two successive iterations. The actual closure criterion is

$$\max_i \left|H_i^{(k)} - H_i^{(k-1)}\right| \leq \epsilon, \qquad (3.2\text{-}40)$$

where

- H_i is the calculated hydraulic head at the node i [L],
- k is the iteration counter [dimensionless], and
- ϵ is the closure criterion [L].

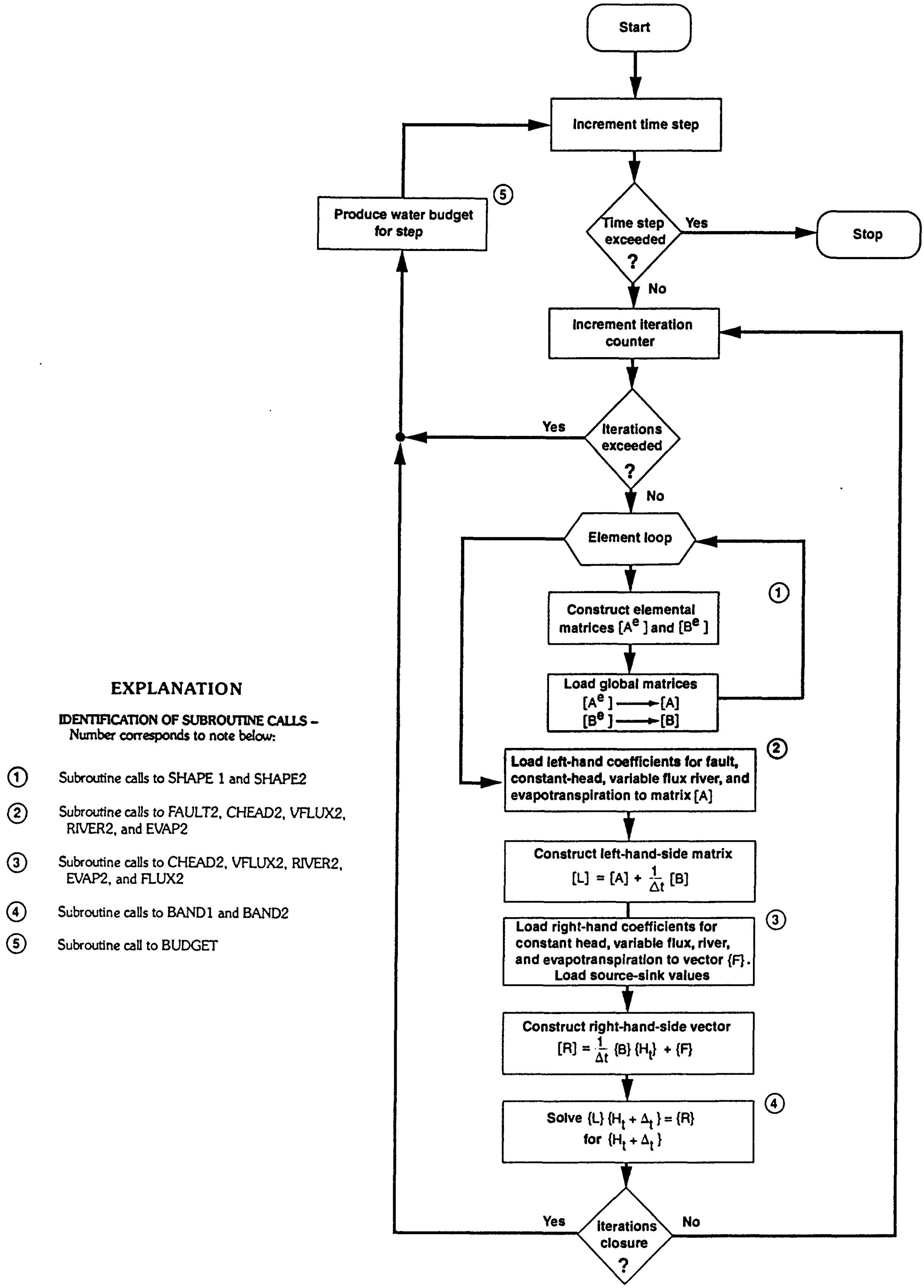


Figure 2. Structure of subroutine *MODEL*.

3.2.3.2 Compressed Storage

A compressed storage scheme is used in subroutine *MODEL* for the matrices *[A]* and *[B]*. Because these matrices are symmetric, the storage requirement can be reduced. It is necessary to store only the upper half or the lower half of a symmetric matrix. The half that is not stored can be reconstructed at any time from the relation

$$A_{ij} = A_{ji} , \tag{3.2-41}$$

which applies to any symmetric matrix. Besides being symmetric, the matrices *[A]* and *[B]* are banded. Because it is necessary to store only part of the matrices within the band of nonzero coefficients, minimizing the bands reduce the storage requirement for the matrices. The size of the bands is minimized when the nodes are numbered to minimize the numerical difference between the highest and lowest node within each element. Within a band, the element with the largest numerical difference between its highest and lowest nodes determines the size of the band. Within the finite-element grid, the band with largest size determines the storage requirement of the matrices for the model. Band size is usually minimized when the nodes are sequentially numbered in stacks on vertical planes across the shortest dimension of the grid.

Using matrix *[A]* as an example, figure 3 shows how the symmetric and the banded structure of matrices *[A]* and *[B]* can be used to reduce the storage requirements. As indicated on the figure, the diagonal of matrix *[A]* occurs in the compressed matrix *[A′]* as column 1. Accordingly, a column of matrix *[A]* occurs in the compressed matrix *[A′]* as a diagonal where the diagonal runs downward from the left.

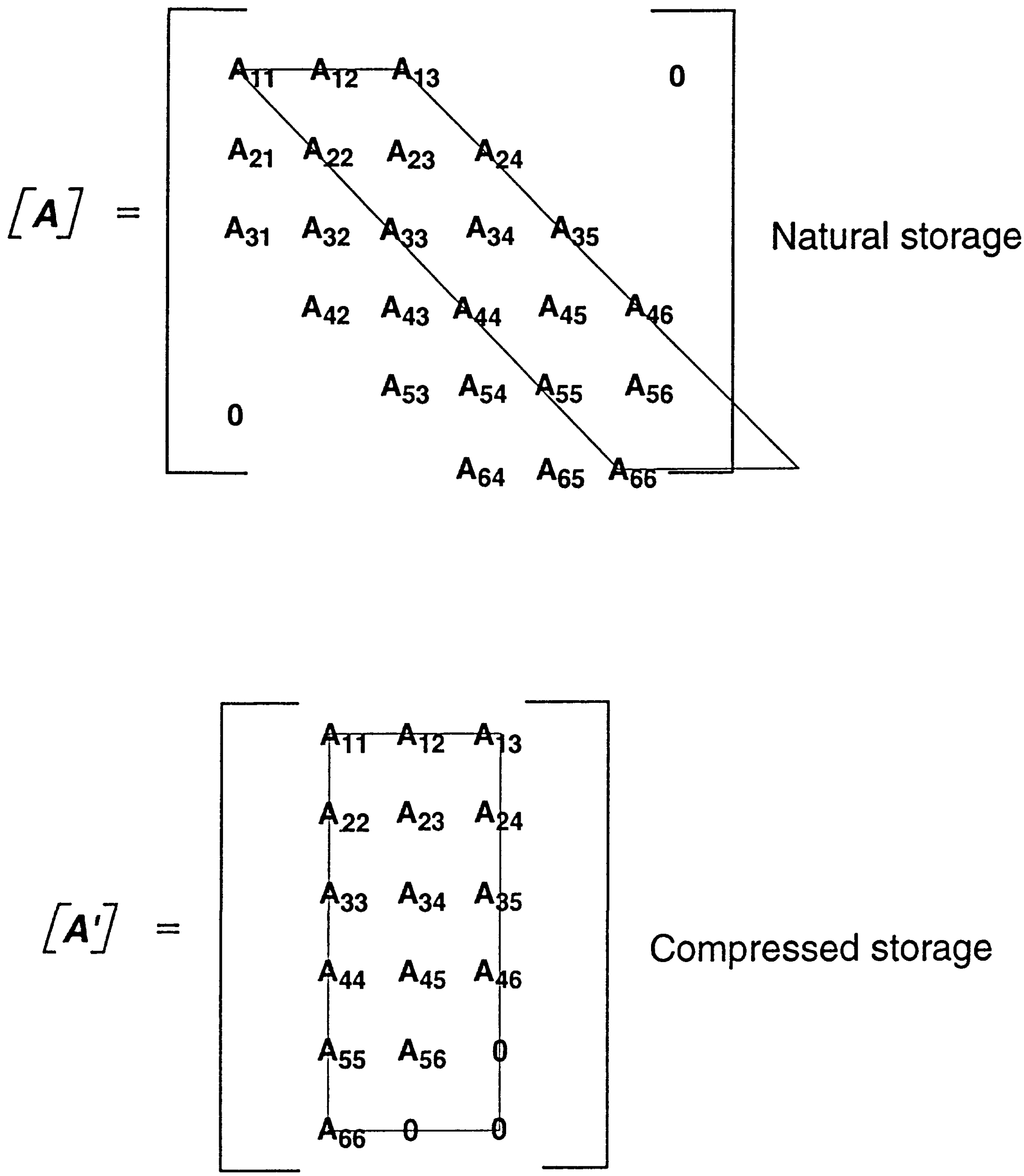


Figure 3. Compressed storage of matrix *[A]*.

3.2.3.3 Linear Problems

In some applications of *FEMFLOW3D*, the matrices *[A]* and *[B]*, do not change during simulation. This occurs when the coefficients of *[A]*, *[B]*, and *[F]* are independent of *{H}*, which means that Equation 3.2-38 is linear. This linear equation occurs when (1) the drainage-node form of the specified-head boundary condition is not used, (2) river-aquifer interactions are not simulated, (3) evapotranspiration from ground water is not simulated, and (4) land subsidence is not simulated. If each of these conditions is satisfied, the matrices *[A]* and *[B]* need to be constructed only once. Further, if the time step is constant, the upper triangularization of the matrix

$$\left([A] + \frac{1}{\Delta t}[B]\right),$$

which occurs in Equation 3.2-38, needs to be done only once in subroutine *BAND*. Accordingly, a switch can be set in subroutine *MODEL* to identify the problem as linear. In this case, the matrices *[A]* and *[B]* will be constructed only once, which will reduce the computational requirements of the application.

3.2.3.4 Finite-Element Grid

The finite-element grid defines the geometry of the ground-water system. The grid consists of a solid configuration of tetrahedrons. Because tetrahedrons are difficult solids with which to work, the actual grid is assembled from prismatic elements with a triangular cross section. The prismatic elements are oriented spatially with subhorizontal triangular faces and subvertical quadrilateral faces. Subroutine *MODEL* then automatically fits three tetrahedrons into each prismatic element (fig. 4). However, to allow flexibility in the construction of three-dimensional grids, the subroutine also accepts prismatic elements that contain edges of zero height (fig. 5). A prismatic element with one zero-height edge contains two tetrahedrons, and an element with two zero-height edges contains one. These special elements can be used to represent geologic features that taper to zero thickness or they can be used to include a vertically fine grid in zones of particular interest. Without these special elements, a vertical zone of fine gridding can be terminated only by carrying it to the edge of the grid.

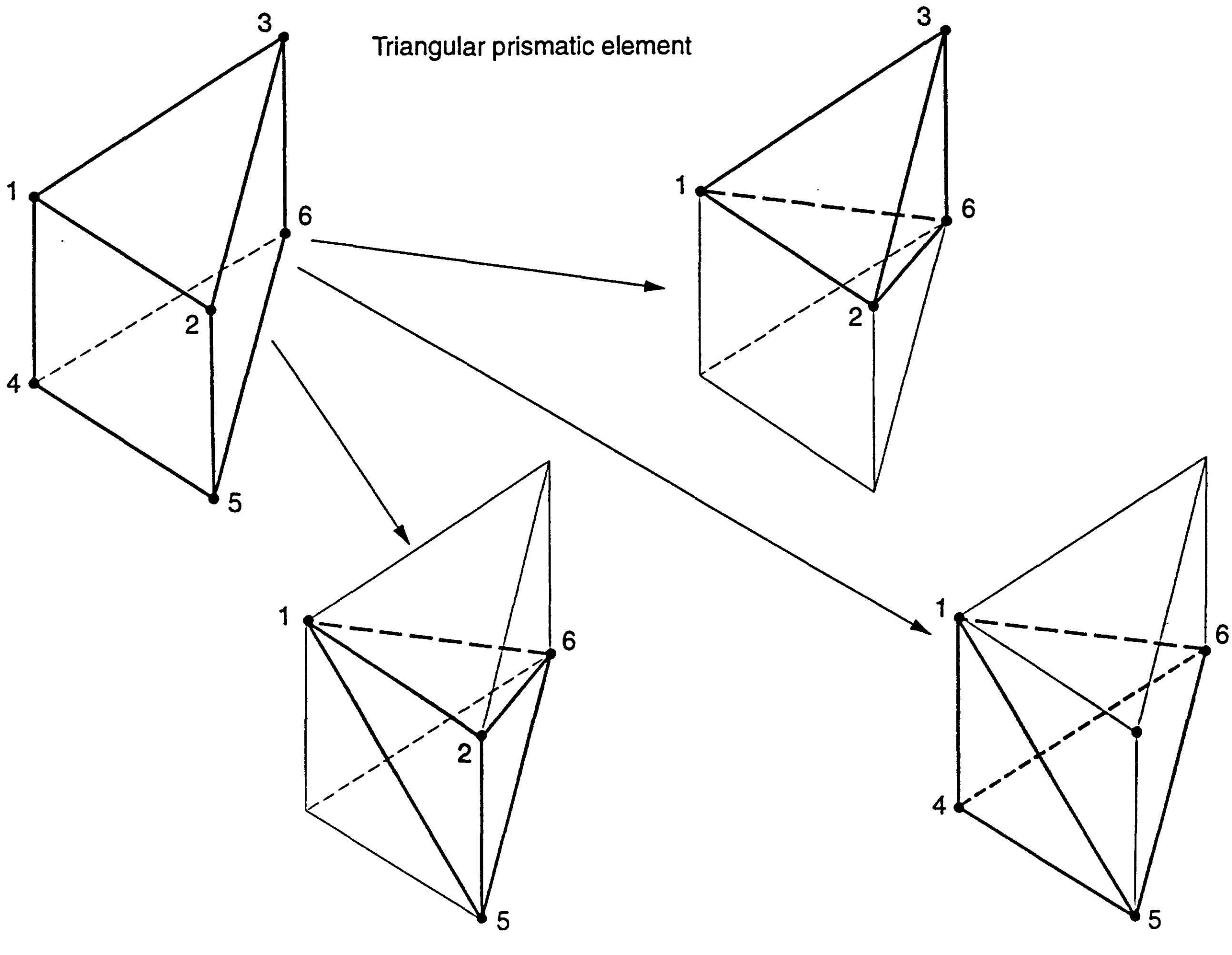


Figure 4. Disaggregation of triangular prismatic elements into tetrahedrons.

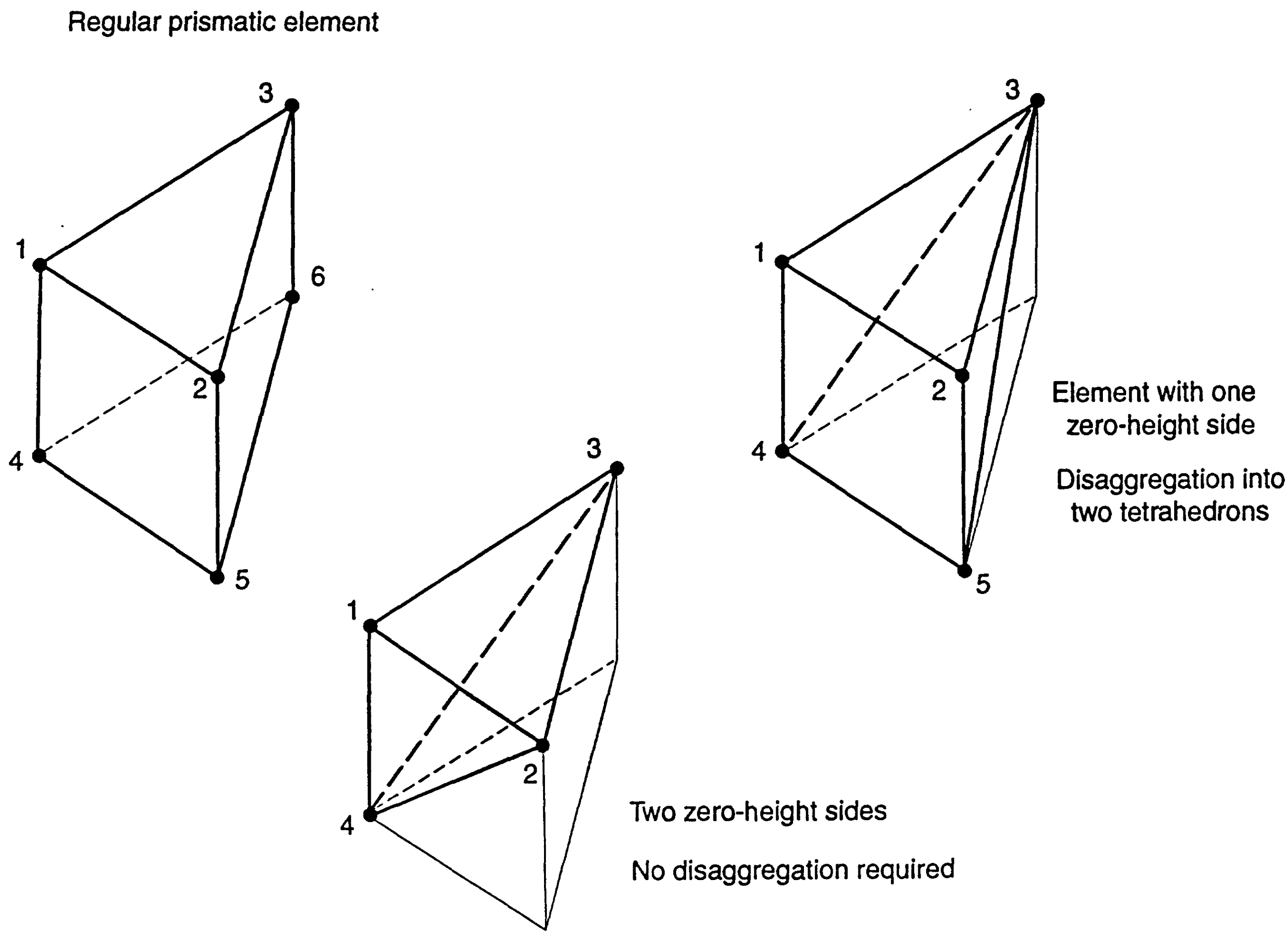


Figure 5. Triangular prisms with zero-height edges.

3.3 Subroutine *CHEAD*

3.3.1 Background

Subroutine *CHEAD* is used to represent specified-head boundary conditions, which can take three different forms. In the first form, unlimited discharge can occur into or from the ground-water system through the specified-head nodes. For example, this type of boundary condition is used with constantly flowing rivers that maintain a hydraulic connection to the water table. In the second form, unlimited discharge can occur from the ground-water system through specified-head nodes, but discharge into the ground-water system through the nodes is limited. This boundary condition is used when a water table periodically drops below a river or a lake. In the third form, unlimited discharge can occur from the ground-water system through specified-head nodes, but no discharge can occur into the ground-water system through the nodes. This boundary condition is used to represent drainage systems. Such a condition might exist where ground-water can discharge to a subsurface drain pipe. However, if water otherwise does not flow in the pipe, water cannot discharge from the pipe to the ground-water system.

The heads specified in the subroutine *CHEAD* either may be constant or may vary with time. A single head elevation for each specified-head node may be assigned for the entire simulation to represent invariant conditions, such as the elevation of a subsurface drain pipe. Alternatively, the head elevation for each specified-head node may vary in time in accordance with a specified hydrograph, which is input to the model as a table of hydraulic heads at specified times

3.3.2 Mathematical Basis

<u>Discharge equation</u>.--The equation for the discharge of water across the boundary surface owing to specified heads is

$$q_B(x,y,t) = \sum_{i=1}^{n} C_{Bi}(H_{Bi} - H_i)\, \delta(x-x_i)\, \delta(y-y_i)\,, \qquad (3.3\text{-}1)$$

where

- q_B is the functional representation of ground-water discharges owing to the specified-head boundary condition [L/t],
- C_{Bi} is a coefficient representing the leakance of the specified-head boundary condition for the node i [L^2/t],
- H_{Bi} is the specified head for the node i [L],
- H_i is the calculated hydraulic head for the node i,
- x_i is the x coordinate for the node i [L],
- y_i is the y coordinate for the node i [L],
- δ is the Dirac delta function [1/L], and
- n is the number of nodes [dimensionless],

and where the coordinate system locally is oriented within the plane of the boundary surface. For a drainage boundary condition in which the direction of flow across the boundary is outward, flow occurs only when H_i is greater than H_{Bi} in Equation 3.3-1.

The discharge q_B has a nonzero value at the specified-head nodes and a zero value elsewhere, where the discharge q_B is the discharge necessary to produce calculated hydraulic heads H_i at node i that are approximately equal to the boundary heads H_{Bi} at the node i. This relation between specified and calculated heads can be achieved if the coefficient C_{Bi} in Equation 3.3-1 is sufficiently large. The practical application of the specified-head boundary condition does not require the careful determination of values for the coefficient C_{Bi}. It is sufficient to choose a value that is large enough that H_i will be close to H_{Bi}. Nevertheless, the value should not be so large that the difference between H_i and B_{Bi} is lost in the precision of the calculations.

Substitution into force vector.--Equation 3.2-15 is an expression for F_i, which is the quantity representing the internal and boundary fluxes for the model domain. Correspondingly, Equation 3.3-1 is an expression for the boundary fluxes representing specified-head boundaries. Equation 3.2-15 and 3.3-1 can be combined into the form

$$F_i = F_{Bi}' + C_{Bi}(H_{Bi} - H_i) \tag{3.3-2}$$

where F_{Bi}' represents the internal and boundary fluxes associated with node i except for the boundary fluxes associated with the specified-head boundary conditions.

The right-hand side of Equation 3.3-2 includes the dependent variable {H} from Equation 3.2-38, which is the system of algebraic equations representing the heads {H} at the end of the current time step. With the substitution of Equation 3.3-2 for {F} in Equation 3.2-38, the dependent variable occurs on both the right-hand and left-hand side of Equation 3.2-38 in the form

$$\left([A]+\frac{1}{\Delta t}[B]\right)\{H_{t+\Delta t}\}=\frac{1}{\Delta t}[B]\{H_t\}+\{F_B^{'}\}+\{C_B H_B\}-diag\{C_B\}\{H_{t+\Delta t}\} \tag{3.3-3}$$

or

$$\left([A]+\frac{1}{\Delta t}[B]+diag\{C_B\}\right)\{H_{t+\Delta t}\}=\frac{1}{\Delta t}[B]\{H_t\}+\{F_B^{'}\}+\{C_B H_B\} \tag{3.3-4}$$

where

$$\{C_B H_B\}=diag\{C_B\}\{H_B\}. \tag{3.3-5}$$

Equation 3.3-4 is equivalent to adding the quantities C_{Bi} to the diagonal of the left-hand martix

$$\left([A]+\frac{1}{\Delta t}[B]\right)$$

in Equation 3.2-38 and to adding the quantities C_{Bi} H_{Bi} to the right hand vector

$$\frac{1}{\Delta t}[B]\{H_t\}+\{F\}.$$

3.3.3 Structure of Subroutine

Subroutine *CHEAD* is structured into three basic blocks. The first block, which is entered by a call to *CHEAD1*, is for the input and display of data for the identification of specified-head nodes in the finite-element grid and the particular form of the boundary condition. The second block, which is entered by a call to *CHEAD2*, creates the coefficients to be added to *diag[A]* and *{F}*, as described in Section 3.3.2. The third block, which is entered by a call to *CHEAD3*, calculates the actual ground-water discharge through each specified-head node.

In the third block of subroutine *CHEAD*, the discharge of ground water through individual nodes in the finite-element grid is calculated from the expression

$$\{Q_B\} = \{C_{BR}\} - \{C_{BL}\, H\}, \tag{3.3-6}$$

where an element of $\{Q_B\}$ is given by

$$Q_{Bi} = C_{BRi} - C_{BLi} H_i, \tag{3.3-7}$$

and where $\{Q_B\}$ is a vector of inward discharges [L^3/t]. The cumulative discharge through all specified-head nodes is given by

$$Q_B = \sum_{i=1}^{n} Q_{Bi}, \tag{3.3-8}$$

where

Q_B is the cumulative discharge [L^3/t], and
n is the number of nodes [dimensionless].

3.4 Subroutine *FLUX*

3.4.1 Background

Subroutine *FLUX* is used to specify values for the source-sink term W and the boundary-flux term q in Equation 3.2-15. In subroutine *FLUX*, the boundary-flux term is for the specified-flux boundary condition.

3.4.2 Mathematical Basis

Discharge equation.--The internal source-sink fluxes and the boundary fluxes are both represented by nodal values. Mathematically, the internal source-sink fluxes are given by the relation

$$W_P(x,y,z,t) = \sum_{i=1}^{n} Q_{Pi}(t)\ \delta(x-x_i)\ \delta\ (y-y_i)\ \delta\ (z-z_i), \qquad (3.4\text{-}1)$$

where

W_P is the functional representation of internal source-sink fluxes [1/t],
Q_P is the source-sink flux at the node i [L^3/t],
x_i is the x coordinate of the node i [L],
y_i is the y coordinate of the node i [L],
z_i is the z coordinate of the node i [L],
δ is the Dirac delta function [1/L], and
n is the number of nodes [dimensionless].

The discharge w_P has a nonzero value at nodes where source-sink fluxes occur and a zero value elsewhere.

Substitution into force vector.--Equation 3.2-15 is an expression for F_i, which is the quantity representing the internal and boundary fluxes for the model domain. Correspondingly, Equation 3.4-1 is an expression for the boundary fluxes representing specified fluxes. Equations 3.2-15 and 3.4-1 can be combined into the form:

$$F_i = F'_{Pi} + Q_{Pi}\ , \qquad (3.4\text{-}2)$$

where F'_{Pi} represents the internal and boundary fluxes associated with node i except for the boundary fluxes associated with the specified fluxes. With the substitution of Equation 3.4-2 for {F} in Equation 3.2-38, the resulting expression is

$$\left([A]+\frac{1}{\Delta t}[B]\right)\{H_{t+\Delta t}\}=\frac{1}{\Delta t}[B]\{H_t\}+\{F'_P\}+Q_P. \qquad (3.4\text{-}3)$$

3.4.3 Structure of Subroutine

Subroutine *FLUX* is structured into two basic blocks. The first block, which is entered by a call to *FLUX1*, is for the input and display of data on nodal fluxes for either internal source-sink fluxes or boundary fluxes. The second block, which is entered by a call to *FLUX2*, combines data sets on nodal fluxes to produce the nodal fluxes for a particular time step.

The nodal fluxes are input to subroutine *FLUX* as a series of nodal-flux data sets. Then, the nodal fluxes for a particular time step are assigned to that time step by constructing a linear combination of the data sets in the form

$$\{Q_P\} = a_i \sum_{i=1}^{n} \{Q'_P\}_i \, , \tag{3.4-4}$$

where

$\{Q_P\}$ is the vector of combined nodal fluxes for the time step [L^3/t],
a_i is a multiplicative factor for the data set i [dimensionless],
n is the number of data sets [dimensionless], and
$\{Q'_P\}_i$ is the vector of nodal fluxes for the data set i [L^3/t].

Each data set represents a set of fluxes for each node. The flux for a particular node is the sum of the values from each data set multiplied by a factor for the data set, where the factor can be zero.

The cumulative discharge through all specified-flux nodes is given by

$$Q_P = \sum_{i=1}^{n} Q_{Pi} \, , \tag{3.4-5}$$

where

Q_P is the cumulative inward discharge [L^3/t], and
n is the number of nodes [dimensionless].

3.5 Subroutine *RIVER*

3.5.1 Background

Subroutine *RIVER* is used to simulate river-aquifer interactions. The simulation is performed by routing input inflows through a river network (or networks) that consists of a main river channel and tributary channels, while accounting for riverflow depletions and accretions owing to the exchange of water between the river and the ground-water system. Each river channel consists of a series of reaches associated with nodes in the model. In subroutine *RIVER*, the exchange of water is dependent on the river stage, river width, river-bed thickness, river-bed hydraulic conductivity, and ground-water levels at each node in the model.

Section 5.2.11 provides a detailed example of a representation of river networks in a model.

3.5.2 Mathematical Basis

Discharge equation.--The exchange of water between the river and the ground-water system has the general form

$$q_R\,(x,y,t) = \sum_{i=1}^{n} C_{Ri}\,(H_{Ri} - H_i)\;\delta\,(x-x_i)\;\delta(y-y_i), \tag{3.5-1}$$

where

q_R is the functional representation of the exchange rate [L/t],
C_{Ri} is a coefficient representing the river-bed leakance associated with the node i [L^2/t],
H_{Ri} is the stage in the river reach for the node i [L],
H_i is the calculated hydraulic head for the node i [L],
x_i is the x coordinate of the node i [L],
y_i is the y coordinate of the node i [L],
δ is the Dirac delta function [dimensionless], and
n is the number of nodes [dimensionless].

The exchange rate q_R has a nonzero value at the river nodes and a zero value elsewhere.

The coefficient C_{Ri} in Equation 3.5-1 depends on several factors, as indicated in the expression

$$C_{Ri} = L_i W_i \frac{K_i'}{B_i'} , \qquad (3.5\text{-}2)$$

where
L_i is the reach length associated with the node i [L],
W_i is the river width at node i [L],
K_i' is the vertical hydraulic conductivity of the river bed at node i [L/t], and
B_i' is the thickness of the river bed at node i [L].

Accordingly, the coefficient C_{Ri} is the product of the wetted area of the river reach $(L_i W_i)$ and the river-bed leakance (K_i'/B_i').

River width.--The river width, in turn, is given by the relation (Leopold and others, 1964, p. 214-241)

$$W_i = (\alpha_w Q_{Si})^{m_w} , \qquad (3.5\text{-}3)$$

where
α_w is a coefficient for the river width $[(t/L^2)^{1/m_w}]$,
Q_{Si} is the river flow $[L^3/t]$, and
m_w is an exponent for the river width [dimensionless].

The values of α_w and m_w are the intercept and slope of the log-log width-flow relation for the river.

Alternatively, the river width can be specified as a table of widths and the corresponding flows. Then, the width for a particular flow can be obtained from the table by interpolation between the tabulated values.

River stage.--The river stage H_{Ri} in Equation 3.5-1 also depends on several factors, as indicated in the expression (Leopold and others, 1964, p. 214-241)

$$H_{Ri} = H_{Di} + (\alpha_D Q_{Si})^{m_D} , \qquad (3.5\text{-}4)$$

where
H_{Ri} is the stage in the river reach at the node i [L],
H_{Di} is the river-bed elevation [L], which is measured from the same datum as H_i,
α_D is a coefficient for river depth $[(t/L^2)^{1/m_D}]$,
Q_{si} is the river flow $[L^3/t]$, and
m_D is an exponent for the river depth [dimensionless].

The values of α_D and m_D are the intercept and slope of the log-log depth-flow relation for the river.

Alternatively, the river depth can be specified as a table of depths and the corresponding flows. Then, the depth for a particular discharge can be obtained from the table by interpolation between the tabulated values.

Substitution into force vector.--Equation 3.2-15 is an expression for F_i, which is the quantity representing the internal and boundary fluxes for the model domain. Correspondingly, Equation 3.5-1 is an expression for the boundary fluxes representing river nodes. Equations 3.2-15 and 3.5-1 can be combined into the form

$$F_i = F'_{Ri} + C_{Ri}\left(H_{Ri} - H_i\right), \tag{3.5-5}$$

where F'_{Ri} represents the internal and boundary fluxes associated with node i except for the boundary fluxes associated with the river nodes.

The right-hand side of Equation 3.5-5 includes the dependent variable $\{H\}$ from Equation 3.2-38, which is the system of algebraic equations representing the heads $\{H\}$ at the end of the current time step. With the substitiution of Equation 3.5-5 for $\{F\}$ in Equation 3.2-38, the dependent variable occurs on both the right-hand and left-hand side of Equation 3.2-38 in the form

$$\left([A]+\frac{1}{\Delta t}[B]\right)\{H_{t+\Delta t}\}=\frac{1}{\Delta t}[B]\{H_t\}+\{F'_R\}+\{C_R H_R\}-diag\{C_R\}\{H_{t+\Delta t}\} \tag{3.5-6}$$

or

$$\left([A]+\frac{1}{\Delta t}[B]\right)+diag\{C_R\})\{H_{t+\Delta t}\}=\frac{1}{\Delta t}[B]\{H_t\}+\{F'_R\}+\{C_R H_R\}, \tag{3.5-7}$$

where

$$\{C_R H_R\}=diag\{C_R\}\{H_R\}, \tag{3.5-8)}$$

Equation 3.5-7 is equivalent to adding the quantities C_{Ri} to the diagonal of the left-hand martrix

$$\left([A]+\frac{1}{\Delta t}[B]\right)$$

in Equation 3.2-38 and to adding the quantities $C_{Ri}\ H_{Ri}$ to the right hand vector

$$\frac{1}{\Delta t}[B]\{H_t\}+\{F\}.$$

Several different conditions determine the values of C_{Ri}. First, if water flows to the lower end of the reach associated with the node i in the finite-element grid, then C_{Ri} is given by Equation 3.5-2 where L_i is the entire length of the reach. Second, if water does not flow to the lower end of the reach, then C_{Ri} again is given by Equation 3.5-2, except that L_i is the wetted length of the reach. Third, if no flow occurs within the reach, then C_{Ri} equals zero. Finally, if the ground water at the node i is below the bottom of the river-bed thickness B', then C_{Ri} equals zero, but the coefficient F_i in Equation 3.2-15 is replaced by

$$F'_{Ri} + C_{Ri}[H_{Ri} - (H_{Di} - B_i')],$$

where

$H_{Di} - B_i'$ is equal to the elevation of the base of the river bed material [L].

For this condition, the lefthand side of Equation 3.2-38 is unchanged. The length L_i is always the wetted length of the reach.

For this latter condition, the hydraulic connection between the river and the ground-water system is broken. The recharge rate to the ground-water system from the river is independent of the hydraulic heads in the ground-water system. For the condition of hydraulic disconnection, the seepage through the river-bed material depends on the head differential across the river-bed material. The head at the upper surface of the river-bed material is the river-surface elevation H_R. The head at the bottom surface of the river-bed material is defined by the assumption of unit hydraulic gradient from the bottom surface to the ground-water table. With that assumption, the head equals the water-table elevation plus the distance from the water table to the bottom surface. However, the sum of these quantities equals the elevation of the bottom surface of the bed materials H_{Di}-B_i.

3.5.3 Structure of Subroutine

Subroutine *RIVER* is structured into three basic blocks. The first block, which is entered by a call to *RIVER1*, is for the input and display of data for the specification of the river network, the physical properties of the river reaches, and the river inflow inputs. The second block, which is entered by a call to *RIVER2*, creates the coefficients to be added to *diag[A]* and *{F}*, as described in Section 3.5.2. The third block, which is entered by a call to *RIVER3*, calculates the components of the water budget for the river network, including the exchanges of water between the river network and the ground-water system.

In the third block, the exchange of water between the river and ground-water system is calculated from the expression

$$\{Q_R\} = \{C_{RR}\} - \{C_{RL}H\}, \tag{3.5-9}$$

where an element of $\{Q_R\}$ is given by

$$Q_{Ri} = C_{RRi} - C_{RLi}H_i, \tag{3.5-10}$$

and where $\{Q_R\}$ is a vector of inward discharges [L^3/t]. The cumulative discharge through all nodes is given by

$$Q_R = \sum_{i=1}^{n} Q_{Ri}, \tag{3.5-11}$$

where

Q_R is the cumulative inward discharge [L^3/t], and

n is the number of nodes [dimensionless].

3.6 Subroutine *EVAP*

3.6.1 Background

Subroutine *EVAP* is used to simulate the discharge of ground-water from a shallow water table owing to evapotranspiration from vegetated areas or evaporation from bare-soil areas. Simulation is done by assuming that the discharge is linearly related to the depth below the land surface to the water table. The linear relation holds until a maximum depth (extinction depth) is reached. If the water table drops below the extinction depth, evapotranspiration (or evaporation) ceases. In subroutine *EVAP*, the evapotranspiration rate depends on the local depth to the water table, the extinction depth, the potential evapotranspiration rate, and the size of the discharge area.

3.6.2 Mathematical Basis

<u>Discharge equation</u>.--The discharge of ground water from a shallow water table has the general form

$$q_E(x,y,t) = \sum_{i=1}^{n} C_{Ei}\,(H_{Ei} - H_i)\,\delta(x - x_i)\,\delta(y - y_i)\,, \tag{3.6-1}$$

where
- q_E is the functional representation discharge of ground water [L/t],
- C_{Ei} is a coefficient representing the evapotranspiration at the node i [L^2/t],
- H_{Ei} is the extinction-depth elevation [L],
- H_i is the calculated hydraulic head for the node i [L],
- x_i is the x coordinate of the node i [L],
- y_i is the y coordinate of the node i [L],
- δ is the Dirac delta function [1/L], and
- n is the number of nodes [dimensionless].

The discharge q_E has a nonzero value for nodes where ground-water discharge occurs and a zero value elsewhere.

The coefficient C_{Ei} in Equation 3.6-1 depends on several factors, as indicated in the expression

$$C_{Ei} = \frac{A_i E_{max}}{H_{Li} - H_{Ei}} , \tag{3.6-2}$$

where

A_i is the discharge area associated with node i [L^2],
E_{max} is the potential evapotranspiration rate per unit area [L/t],
H_{Li} is the land-surface elevation at the node i [L], and
H_{Ei} is the extinction-depth elevation at the node i [L], and is given by

$$H_{Ei} = H_{Li} - d_o , \tag{3.6-3}$$

where d_o is the extinction depth [L].

Substitution into force vector.--Equation 3.2-15 is an expression for F_i, which is the quantity representing the internal and boundary fluxes for the model domain. Correspondingly, Equation 3.6-1 is an expression for the boundary fluxes representing evapotranspiration nodes. Equations 3.2-15 and 3.6-1 can be combined into the form

$$F_i = F'_{Ei} + C_{Ei} (H_{Ei} - H_i), \tag{3.6-4}$$

where F'_{Ei} represents the internal and boundary fluxes associated with node i except for the boundary fluxes associated with the evapotranspiration nodes.

The right-hand side of Equation 3.6-4 includes the dependent variable *{H}* from Equation 3.2-38, which is the system of algebraic equations representing the heads *{H}* at the end of the current time step. With the substitution of Equation 3.6-4 for *{F}* in Equation 3.2-38, the dependent variable occurs on both the right-hand and left-hand side of Equation 3.2-38 in the form

$$\left([A] + \frac{1}{\Delta t}[B]\right)\{H_{t+\Delta t}\} = \frac{1}{\Delta t}[B]\{H_t\} + \{F'_E\} + \{C_E H_E\} - diag\{C_E\}\{H_{t+\Delta t}\} , \qquad (3.6\text{-}5)$$

or

$$\left([A] + \frac{1}{\Delta t}[B] + diag\{C_E\}\right)\{H_{t+\Delta t}\} = \frac{1}{\Delta t}[B]\{H_t\} + \{F'_E\}+\{C_E H_E\} , \qquad (3.6\text{-}6)$$

where

$$\{C_E H_E\} = diag\{C_E\}\{H_E\} . \qquad (3.6\text{-}7)$$

Equation 3.6.6 is equivalent to adding the quantities C_{Ei} to the diagonal of the left-hand martix

$$\left([A] + \frac{1}{\Delta t}[B]\right)$$

in Equation 3.2.38 and to adding the quantities C_{Ei} H_{Ei} to the right hand vector

$$\frac{1}{\Delta t}[B]\{H_t\} + \{F\} .$$

Several different conditions can occur for values of C_{Ei}. First, if the calculated hydraulic head H_i is above the extinction depth and below the land surface, then C_{Ei} is given by Equation 3.6-2. Second, if the calculated hydraulic head H_i is below the extinction depth, then C_{Ei} equals zero, and no discharge occurs from the shallow water table. Third, if the calculated hydraulic head H_i is at or above the land surface, C_{Ei} is set to a large value to create a specified-head boundary condition with the land-surface elevation as the boundary head.

For this last condition, the discharge rate from a shallow water table is greater than the potential evapotranspiration rate and ground water seeps onto the land surface. The simulation assumes that the seepage onto the land surface is removed from further interaction with the ground-water system, which, for example, would occur if the seepage is collected into surface-water channels and then exits the geographic area of the ground-water system as surface-water outflow.

3.6.3 Structure of Subroutine

Subroutine *EVAP* is structured into three basic blocks. The first block, which is entered by a call to *EVAP1*, is for the input and display of data for the discharge of ground water by evapotranspiration, including data on the discharge area, extinction depth, and potential evapotranspiration rate. The second block, which is entered by a call to *EVAP2*, creates the coefficients to be added to *diag[A]* and *{F}*, as described in Section 3.6.2. The third block, which is entered by a call to *EVAP3*, calculates the ground-water discharge by evapotranspiration.

In the third block, the discharge of ground water by evapotranspiration is calculated from the relation

$$\{Q_E\} = \{C_{ER}\} - \{C_{EL}H\}, \qquad (3.6\text{-}8)$$

where an element of $\{Q_E\}$ is given by

$$Q_{Ei} = C_{ERi} - C_{ELi}H_i, \qquad (3.6\text{-}9)$$

and where $\{Q_E\}$ is a vector of inward discharges [L^3/t]. The cumulative discharge through all nodes is given by

$$Q_E = \sum_{i=1}^{n} Q_{Ei}, \qquad (3.6\text{-}10)$$

where

Q_E is the cumulative inward discharge [L^3/t], and

n is the number of nodes [dimensionless].

3.7 Subroutine *VFLUX*

3.7.1 Background

Subroutine *VFLUX* is used to specify variable-flux boundary conditions. These conditions are for the boundary fluxes that would result if the modeled flow domain was extended outward a large distance from the actual boundary of the modeled flow domain. This is done by attaching the analytical solution for a semi-infinite linear ground-water system to the boundary of the modeled flow domain.

Section 5.2.13 provides a detailed example of a representation of variable-flux boundaries in a model.

3.7.2 Mathematical Basis

Discharge equation.--The exchange of water at the boundary of the modeled flow domain has the general form

$$q_V(x,y,t) = \sum_{i=1}^{n} [C_{Vi}(H_{Vi} - H_i) + Q_{Vi}' \delta(X-X_i) \delta(Y-Y_i)], \tag{3.7-1}$$

where

q_V is the functional representation of the exchange rate [L/t],
C_{Vi} is a coefficient relating to the effect in the current time step owing to the head change for node i at the boundary in the current time step [L^2/t],
H_{Vi} is the steady-state hydraulic head for node i [L],
H_i is the calculated hydraulic head for node i for the current time step [L],
Q_{Vi}' is the discharge across the boundary at node i in the current time step owing to head changes at the boundary in past time steps [L^3/t],
x_i is the x coordinate of the node i [L],
y_i is the y coordinate of the node i [L],
δ is the Dirac delta function [1/L],
n is the number of nodes [dimensionless],
i is the node number [dimensionless],

and where the coordinate system is locally oriented within the plane of the boundary surface.

Equation 3.7-1 applies only if the simulation starts with a steady-state initial condition. This requirement eliminates the need for considering the time fluctuations of the boundary heads at time prior to t=0 in the simulation.

The coefficient C_{Vi} and the discharge Q_{Vi}' have nonzero values for nodes on the variable-flux boundaries and zero values elsewhere. Correspondingly, the exchange rate q_v has a nonzero value for nodes where both C_{Vi} and Q_{Vi}' have nonzero values.

The coefficient C_{Vi} for the effect in the current time step owing to the head change in the current time step is given by the relation

$$C_{Vi} = \frac{2}{\Delta t}\left[\frac{K_i B_i W_i}{\left(\frac{\pi K_i}{S_{si}}\right)^{1/2}}\right]\Delta t^{1/2}, \qquad (3.7\text{-}2)$$

where

K_i is the hydraulic conductivity of the extended ground-water system for node i [L/t],
B_i is the thickness of the extended ground-water system for node i [L],
W_i is the width of the extended ground-water system for node i [L],
Δt is the time-step length [t],
S_{si} is the specific storage for the extended ground-water system for node i [1/L], and
i is the node number.

Semi-infinite aquifer.--Equation 3.7-2 is derived from the analytical solution for the discharge at $x = 0$ for a linear ground-water system that extends from $x = 0$ to $x = \infty$ (Carslaw and Jaeger, 1959). That solution is

$$q_{oi} = -\frac{K_i B_i W_i}{\left(\frac{\pi K_i t}{S_{si}}\right)^{1/2}}\Delta h_i\,, \qquad (3.7\text{-}3)$$

where

q_{oi} is the instantaneous discharge rate [L^3/t], and
Δhi is a step change in the boundary head at $x = 0$ in the extended ground-water system for node i [L].

The discharge q_{oi} is positive for discharge from the extended ground-water system into the modeled flow domain.

Equation 3.7-3 can be integrated over a time step to get the average discharge rate within the time step. The integration of

$$Q_{oi} = -\frac{1}{\Delta t}\int_{t_1}^{t_2}\left[\frac{K_iB_iW_i}{\left(\frac{\pi K_i t}{S_{si}}\right)^{1/2}}\right]\Delta h_i dt \tag{3.7-4}$$

yields

$$Q_{oi} = -\frac{2}{\Delta t}\left[\frac{K_iB_iW_i}{\left(\frac{\pi K_i}{S_{si}}\right)^{1/2}}\right]\Delta h_i(t_2^{1/2} - t_1^{1/2}), \tag{3.7-5}$$

where

Q_{oi} is the average discharge during the period t_1 to t_2 [L^3/t],

t_1 is the time at the start of a step head change Δh_i [t],

t_2 is the time at the end of a step head change [t],

and where

$$\Delta t = t_2 - t_1. \tag{3.7-6}$$

Equation 3.7-2 is obtained from Equation 3.7-5 by noting that the average discharge contribution for Δh of the current time step *m* coincides with $t_2 = \Delta t = t_m - t_{m-1}$ and $t_1 = 0$.

<u>Discharge owing to current step</u>.--Equation 3.7-5 can be rewritten in the form for a particular node *i*

$$Q_{oij} = C_{Vi}\,(-\Delta h_{ij}), \tag{3.7-7}$$

where

j is the index for time step (dimensionless)

which corresponds to Equations 3.7-1 and 3.7-2, where

$$-\Delta h_{ij} = H_{Vij} - H_{ij}\,. \tag{3.7-8}$$

Discharge owing to past steps. The discharge Q_{Vi}' across the boundary for a particular node i in the current time step owing to head changes at the boundary for a particular node i in a past time step is derived from Equation 3.7-5 in the form

$$Q_V' = -\sum_{j=1}^{m-1} \frac{2}{\Delta t_m} \left[\frac{KBW}{\left(\frac{\pi K}{S_s}\right)^{1/2}} \right] \Delta h_j \left[(t_m - t_{j-1})^{1/2} - (t_m - t_j)^{1/2} - (t_m - t_j)^{1/2} + (t_m - t_{j+1})^{1/2} \right], \quad (3.7\text{-}9)$$

where m is the index for the current time step [dimensionless] and the index i has been dropped for clarity from Q_v', K, B, W, S_s and Δh.

The indexes j and m refer to the time at the end of the indicated time step.

Equation 3.7-9 is based on the application of Equation 3.7-5 using the principal of superposition. Equation 3.7-9 is the solution for the case, where

$$\Delta h_i = 0 \; for \; t < t_m - \Delta t_m$$
$$\Delta h_i = H_{vi} - H_i \; for \; t_m - \Delta t \le t \le t_m$$
$$\Delta h_i = 0 \; for \; t > t_m.$$

On the basis of this case, Δh_i is the change in head from the steady-state head.

Substitution into force vector.-- Equation 3.2-15 is an expression for F_i, which is the quantity representing the internal and boundary fluxes for the model domain. Correspondingly, Equation 3.7-1 is an expression for the boundary fluxes representing variable-flux nodes. Equations 3.2-15 and 3.7-1 can be combined into the form

$$F_i = F'_{vi} + C_{vi}(H_{vi} - H_i) + Q'_v\,, \quad (3.7\text{-}10)$$

where F'_{vi} represents the internal and boundary fluxes associated with node i except for the boundary fluxes associated with the variable-flux nodes.

The right-hand side of Equation 3.7-10 includes the dependent variable *{H}* from Equation 3.2-38, which is the system of algebraic equations representing the heads *{H}* at the end of the current time step. With the substitution of Equation 3.7-10 for *{F}* in Equation 3.2-38, the dependent variable occurs on both the right-hand and left-hand side of Equation 3.2-38 in the form

$$\left([A] + \frac{1}{\Delta t}[B]\right)\{H_{t+\Delta t}\} = \frac{1}{\Delta t}[B]\{H_t\} + \{F'_v\} + \{C_v H_v\} - diag\{C_v\}\{H_{t+\Delta t}\} \qquad (3.7\text{-}11)$$

$$\left([A] + \frac{1}{\Delta t}[B] + diag\{C_v\}\right)\{H_{t+\Delta t}\} = \frac{1}{\Delta t}[B]\{H_t\} + \{F'_v\} + \{C_v H_v\} + Q'_v, \qquad (3.7\text{-}12)$$

where

$$\{C_v H_v\} = diag\{C_v\}\{H_v\}. \qquad (3.7\text{-}13)$$

Equation 3.7-12 is equivalent to adding the quantities C_{vi} to the diagonal of the left-hand matrix

$$\left([A] + \frac{1}{\Delta t}[B]\right)$$

in Equation 3.2-38 and to adding the quantities $C_{vi}\, H_{vi}$ to the right-hand vector

$$\frac{1}{\Delta t}[B]\{H_t\} + \{F\}.$$

3.7.3 Structure of Subroutine

Subroutine *VFLUX* is structured into two basic blocks. The first block, which is entered by a call to *VFLUX1*, is for the input and display of data for the variable-flux boundary condition, including data on the hydraulic conductivity, specific storage, aquifer thickness, and width of the extended ground-water system. The second block, which is entered by a call to *VFLUX2*, creates the coefficients to be added to *diag[A]* and *{F}*, as described in Section 3.7.2, and calculates the discharge of ground water through the individual nodes in the finite-element grid.

The discharge of ground water through the individual nodes is calculated from the expression

$$\{Q_v\} = \{C_{vR}\} - \{C_{VL}H\} + \{Q'_V\}, \tag{3.7-14}$$

where an element of $\{Q_V\}$ is given by

$$Q_{Vi} = C_{VRi} - C_{VLi}H_i + Q'_{Vi}, \tag{3.7-15}$$

where $\{Q_V\}$ is a vector of inward discharges [L^3/t]. The cumulative discharge through all variable-flux nodes is given by

$$Q_v = \sum_{i=1}^{n} Q_{vi}, \tag{3.7-16}$$

where

Q_V is the cumulative discharge [L^3/t], and
n is the number of nodes [dimensionless].

3.8 Subroutine *FAULT*

3.8.1 Background

Subroutine *FAULT* is used to simulate the effects of highly permeable features within the ground-water system, but without an explicit representation of the feature in the finite-element grid. An example of these features include fault zones, where ground water can easily move parallel to the fault plane and well casings perforated over a long depth interval, where ground water can easily move from one aquifer to another through the well casing. These features tend to short circuit the ground-water system by providing high-permeability paths for the movement of ground water through a range of lower permeability media that make up the bulk of the ground-water system.

Subroutine *FAULT* cannot simulate the effects of low-permeability fault zones. However, such hydrogeologic features can be simulated by the assigning low hydraulic-conductivity values to the elements that define the plane of the fault zone. The barrier effect of faults can also be simulated using this approach.

3.8.2 Mathematical Basis

Discharge equations.--The simulation of the effects of highly permeable features within a ground-water system is done in subroutine *FAULT* by allowing the direct exchange of water along a link from one node in the finite-element grid to another node. In this simulation, ground water will flow from one node to another node, where the outflow from the first node is equal to the inflow to the second node. The exchange of water between the node pairs, which can be described by Darcy's (1856) Law, is given by the relations

$$Q_{Fi} = C_{Fij} \, (H_j - H_i) \qquad (3.8\text{-}1)$$

$$Q_{Fj} = C_{Fji} \, (H_j - H_i), \qquad (3.8\text{-}2)$$

and

$$C_{fji} = - C_{fij} \, ,$$

where

Q_{Fi} is the outflow for node i [L^3/t],
C_{Fij} is a coefficient relating to the conductivity of the link between nodes i and j [L^2/t],
H_j is the hydraulic head at node j [L],
H_i is the hydraulic head at node i [L], and
Q_{Fj} is the inflow for node j [L^3/t].

The coefficient C_{Fij}, which relates to the conductivity of the link, is given by the expression

$$C_{Fij} = \frac{T_{ij} B_{ij}}{\Delta L_{ij}}, \tag{3.8-3}$$

where

T_{ij} is the transmissivity of the feature, parallel to the link of the node pair [L^2/t],
B_{ij} is the height or width of the part of the feature represented by the node pair [L], and
ΔL_{ij} is the distance between the node pairs that define the length of the feature [L].

Two examples representing faults and well casings, as well as the configuration of node pairs and their related parameters, are described in greater detail in Section 5.2.14.

If the highly permeably feature is a fault zone, a single layer of nodes defines the geometry of the fault plane. The coefficients ΔL_{ij} and B_{ij} describe the spatial relations between the nodes that define the fault plane. The transmissivity is the product of a unit thickness, normal to the plane of the fault, and the hydraulic conductivity of the fault zone, parallel to the fault node link. To represent both horizontal and vertical ground-water flow within a fault plane, two sets of fault links must be specified: one set links nodes in the horizontal direction, and the other set links nodes in the vertical direction.

If the highly permeable feature is a well casing, a single line of nodes defines the length of the casing perforations. The coefficient ΔL_{ij} is the distance between nodes in the line of nodes that represent the perforations, and the product of transmissivity times the width (diameter of the well) is the conductivity of the well casing to vertical flow.

Substitution into force vector.--Equation 3.2-15 is an expression for F_i, which is the quantity representing the internal and boundary fluxes for the model domain. Correspondingly, Equation 3.8-1 is an expression for the boundary fluxes representing fault nodes. Equations 3.2-15 and 3.8-1 can be combined into the form

$$F_i = F'_{Fi} + C_{Fij}(H_{Fi} - H_i) \qquad (3.8\text{-}4)$$

and

$$F_j = F'_{Fj} + C_{Fji}(H_{Fj} - H_j), \qquad (3.8\text{-}5)$$

where F'_{Fi} and F'_{Fj} represent the internal and boundary fluxes associated with nodes i and j except for the boundary fluxes associated with the fault nodes.

The right-hand side of Equations 3.8-4 and 3.8-5 include the dependent variable {H} from Equation 3.2-38, which is the system of algebraic equations representing the heads {H} at the end of the current time step. With the substitution of Equations 3.8-4 and 3.8-5 for {F} in Equation 3.2-38, the dependent variable occurs on both the right-hand and left-hand side of Equation 3.2-38 in the form

$$\left([A] + \frac{1}{\Delta t}[B]\right)\{H_{t+\Delta t}\} = \frac{1}{\Delta t}[B]\{H_t\} + \{F'_F\} + \{C_F H_F\} - diag\{C_F\}\{H_{t+\Delta t}\} \qquad (3.8\text{-}6)$$

$$\left([A] + \frac{1}{\Delta t}[B] + diag\{C_F\}\right)\{H_{t+\Delta t}\} = \frac{1}{\Delta t}[B]\{H_t\} + \{F'_F\} + \{C_F H_F\} + Q'_v\,, \qquad (3.8\text{-}7)$$

where

$$\{C_F H_F\} = diag\{C_F\}\{H_F\}. \qquad (3.8\text{-}8)$$

Equation 3.8-7 is equivalent to adding C_{Fi} or C_{Fj} to the diagonal of the left-hand matrix

$$\left([A] + \frac{1}{\Delta t}[B]\right)$$

in Equation 3.2-38 and to adding the quantities $C_{Fi}\,H_{Fi}$ or $C_{Fj}\,H_{Fj}$ to the right-hand vector

$$\frac{1}{At}[B]\{H_t\} + \{F\}.$$

3.8.3 Structure of Subroutine

Subroutine *FAULT* is structured into three basic blocks. The first block, which is entered by a call to *FAULT1*, is for the input and display of data on the physical characteristics of the links between node pairs. The second block, which is entered by a call to *FAULT2*, creates the coefficients to be added to *[A]*, as described in Section 3.8.2. The third block, which is entered by a call to *FAULT3*, calculates the ground-water discharges through the links between node pairs.

Discharge through fault-node pairs is calculated using Equation 3.8-1 or 3.8-2. Cumulative discharge through fault-node pairs is given by

$$Q_F = \sum_{i=1}^{n} Q_{Fi} \ , \tag{3.8-9}$$

where

Q_F is the cumulative discharge [L^3/t] and
n is the number of nodes [dimensionless].

However, the cumulative discharge is zero (or very close to zero) because each fault node pair has equal discharge in the opposite direction.

3.9 Subroutine *SINK*

3.9.1 Background

Subroutine *SINK* is used to simulate land subsidence that results from ground-water pumping. This subroutine is based on the interbed compaction model of Leake and Prudic (1988). Pumping causes storage depletion within the ground-water system, which is expressed in changes in the land-surface elevation. Those changes are temporary if the ground-water system is responding within the range of the elastic compressibility of the solid matrix. However, the changes are permanent if the ground-water system is responding within the range of inelastic compressibility of the solid matrix. An inelastic response occurs when the effective stress within the solid matrix of fine-grained hydrogeologic units exceeds the preconsolidation stress.

3.9.2 Mathematical Basis

Relation of subsidence to specific storage.--Changes in the elevation of the land surface (downward or upward) occur in response to changes in the effective stress. Effective stress is given by the relation (Terzaghi, 1925)

$$P' = P - U\,, \tag{3.9-1}$$

where
P' is the effective stress $[M/Lt^2]$,
P is the overburden or geostatic pressure $[M/Lt^2]$, and
U is the pore pressure $[M/Lt^2]$.

If lowering the hydraulic head in the ground-water system does not change the geostatic pressure, which neglects the change in the buoyancy, the change in effective stress is then related only to the change in pore pressure, or

$$\Delta P' = -\gamma_w\,\Delta h, \tag{3.9-2}$$

where
$\Delta P'$ is the change in effective stress $[M/Lt^2]$,
γ_w is the unit weight of water $[M/L^2t^2]$, and
Δh is the change in hydraulic head [L].

Elastic compaction or expansion of the solid matrix is given by the relation (Riley, 1969; Helm, 1975)

$$\Delta B = \Delta h \, S'_{Ske} \, B_o \, , \qquad (3.9\text{-}3)$$

where

- ΔB is the elastic change in thickness of the ground-water system [L], which is also the change in the land-surface elevation,
- S'_{Ske} is the elastic specific storage of the ground-water system [1/L], and
- B_o is the initial thickness of the ground-water system [L].

The approximate inelastic compaction of the solid matrix, which occurs primarily within fine-grained interbeds of the ground-water system, is given by the relation (Helm, 1975)

$$\Delta B^* = \Delta h \, S'_{Skv} \, B_o \, , \qquad (3.9\text{-}4)$$

where

- ΔB^* is the inelastic change in thickness of interbeds [L], which is also the permanent change in the land-surface elevation, and
- S'_{Skv} is the inelastic specific storage of the interbeds [1/L].

Equation 3.9-4 applies only when hydraulic head within the interbeds has gone below the previous lowest head or preconsolidation head. Furthermore, because significant time is required for a pressure change to propagate into an interbed, only some thickness intervals of an interbed may experience inelastic compaction.

The time required for a pressure change to propagate into an interbed is given by the relation (Riley, 1969)

$$\tau = \frac{S'_s \, B_o^2}{4K'} , \qquad (3.9\text{-}5)$$

where

- τ is the effective time constant for the interbed [t],
- S'_s is the effective bulk specific storage for the interbed [1/L],
- B_o is the initial thickness of the interbed [L], and
- K' is the vertical hydraulic conductivity of the interbed [L/t].

The time constant τ is the time required for 93 percent of the pressure change to occur at the center of the interbed. If the specific storage S'_{Skv} is used for the specific storage S'_s in Equation 3.9-5, then the resulting time constant is the upper limit.

<u>Discharge equation</u>.--On the basis of Equations 3.9-3 and 3.9-4, the rate of flow per unit volume of water flowing into storage in compressible interbeds is given by (Leake and Prudic, 1988)

$$q_S^{t+\Delta t} = \frac{S_{Sk}^{t+\Delta t}}{\Delta t}\left(H^{t+\Delta t} - H_P^t\right) + \frac{S_{Ske}}{\Delta t}\left(H_P^t - H^t\right), \qquad (3.9\text{-}6)$$

where

$$S_{Sk} = \begin{cases} S'_{Ske} \; for \; H^t > H_P^{t-\Delta t} \\ S'_{Skv} \; for \; H^t \le H_P^{t-\Delta t} \end{cases} \qquad (3.9\text{-}7)$$

and

- q_s is the rate of storage change per unit volume [L^3/t per L^3 or ^{1}/t],
- H is the calculated hydraulic head [L],
- t is time [t],
- H_P is the preconsolidation head [L], and
- Δt is the time step in the simulation [t].

Modification of storage matrix and force vector.--The implementation of Equation 3.9-6 in the finite-element method requires the modification of Equation 3.2-38 to yield

$$\left([A] + \frac{(1-p)}{\Delta t}[B] + \frac{p}{\Delta t}[C]\right)\{H_{t+\Delta t}\} = \frac{(1-p)}{\Delta t}[B]\{H_t\} + \frac{p}{\Delta t}[C]\{H_{pt}\} + \frac{p}{\Delta t}[D]\{H_t - H_{pt}\} - \{F\}, \tag{3.9-8}$$

where p is the proportion of the local ground-water system that is occupied by interbeds [dimensionless]. When p equals zero, Equation 3.9-8 reduces to Equation 3.2-38.

Typical elements of *[C]* and *[D]* are

$$C_{ij} = \int_{\Omega} S'_{Sk}\phi_i\phi_j d\Omega, \tag{3.9-9}$$

where S'_{Sk} is defined by Equation 3.9-7 and

$$D_{ij} = \int_{\Omega} S'_{Ske}\phi_i\phi_j d\Omega. \tag{3.9-10}$$

3.9.3 Structure of Subroutine

Subroutine *SINK* is structured into three basic blocks. The first block, which is entered by a call to *SINK1*, is for the input and display of data for the simulation of land subsidence. The input include data on the proportion of the local ground-water system occupied by interbeds, the elastic specific storage of the interbeds, and the inelastic specific storage. The second block, which is entered by a call to *SINK2*, is for constructing the elemental matrices for elastic and inelastic storage change within interbeds, where the elemental matrices correspond to the matrices *[C]* and *[D]*, as indicated in Equations 3.9-9 and 3.9-10. The third block is for the calculation of change in the land-surface elevation.

In the third block, the change in land-surface elevation is calculated from the relation

$$\Delta V_s = p\int_B [S'_{Sk}(H_{t+\Delta t}-H_p) + S'_{Ske}(H_P-H_t)]dz \,, \tag{3.9-11}$$

where
ΔV_s is the volume of storage change per unit area over the time interval Δt [L], and
B is the thickness of the ground-water system [L].

However,

$$\Delta B_s = \Delta V_s \,, \tag{3.9-12}$$

where ΔB_s is the change in the land-surface elevation [L].

Elastic and inelastic storage changes within interbeds are components of the water budget for the ground-water system. The rate of elastic storage change within interbeds is given by the relation

$$Q'_{Ske} = \int_\Omega p\, S'_{Ske} \frac{(H_p - H_t)}{\Delta t} d\Omega \,, \tag{3.9-13}$$

if H_p is between H_t and $H_{t+\Delta t}$, where Q'_{Ske} is the rate of elastic storage change [L^3/t] over the time interval. Otherwise,

$$Q'_{Ske} = \int_\Omega p\, S'_{Ske} \frac{(H_{t+\Delta t} - H_t)}{\Delta t} \Omega \,, \tag{3.9-14}$$

if H_p is less than $H_{t+\Delta t}$. The rate of inelastic storage change is given by

$$Q'_{Skv} = \int_\Omega p\, S'_{Skv} \frac{(H_{t+\Delta t} - H_p)}{\Delta t} d\Omega \,, \tag{3.9-15}$$

if H_p is between H_t and $H_{t+\Delta t}$, where Q'_{Ske} is the rate of inelastic storage change [L^3/t]. Otherwise,

$$Q'_{Skv} = 0 \,. \tag{3.9-16}$$

3.10 Subroutine *WATER*

3.10.1 Background

Subroutine *WATER* is used to simulate ground-water recharge and pumpage that result from the operation of an irrigated agricultural system. First, the subroutine distributes irrigation water from ground-water pumpage and surface-water deliveries to individual land parcels, which are referred to as user areas. Second, through a call to subroutine *LAND*, evapotranspiration and ground-water recharge are determined for each user area. Third, the ground-water recharge and ground-water pumpage are distributed to nodes within the finite-element grid.

An irrigated agricultural system, represented by data inputs to subroutine *WATER*, is shown in figure 6. The system consists of separate user areas that are irrigated either with surface water or ground water or both surface water and ground water. The system includes wells that directly deliver water for irrigation or that discharge into a canal system. In a typical application, the user areas represent aggregations of farms at a scale that is appropriate to the problem being simulated. However, user areas can also represent irrigated or nonirrigated parts of an urban area within the model boundary.

Data inputs to subroutine *WATER* specify the total acreage of crops and the soil characteristics within a user area and the precipitation on a user area. On the basis of these inputs, the actual evapotranspiration and the deep percolation of irrigation water are calculated within subroutine *LAND*. A representation of irrigation to a user area is shown in figure 7.

Recharge from a user area is calculated in subroutine *LAND*. Data inputs to subroutine *WATER* specify the nodes within the finite-element grid that represent the recharge from the user area. A representation of recharge to a user area is shown in figure 8. A user area that is arbitrarily distributed geographically in the finite-element grid will be connected with several nodes, depending on the geographic extent of the user area and the layout of the finite-element grid. The nodes and the proportion of user-area recharge for each node are specified by data inputs to subroutine *WATER*. Section 5.3.5 provides a description of the algorithm by which recharge is distributed to nodes in the finite-element grid.

Data inputs to subroutine *WATER* specify pumpage from individual wells. A representation of pumpage from a well is shown in figure 9. Additional data inputs specify the nodes within the finite-element grid that represent the location of a well. Pumpage for a well is distributed to six or more nodes, depending on the depth interval of the well screen. The nodes and the proportion of well pumpage for each node are specified by data inputs to subroutine *WATER*. Section 5.3.1 provides a description of the algorithm by which pumpage is distributed to nodes in the finite-element grid.

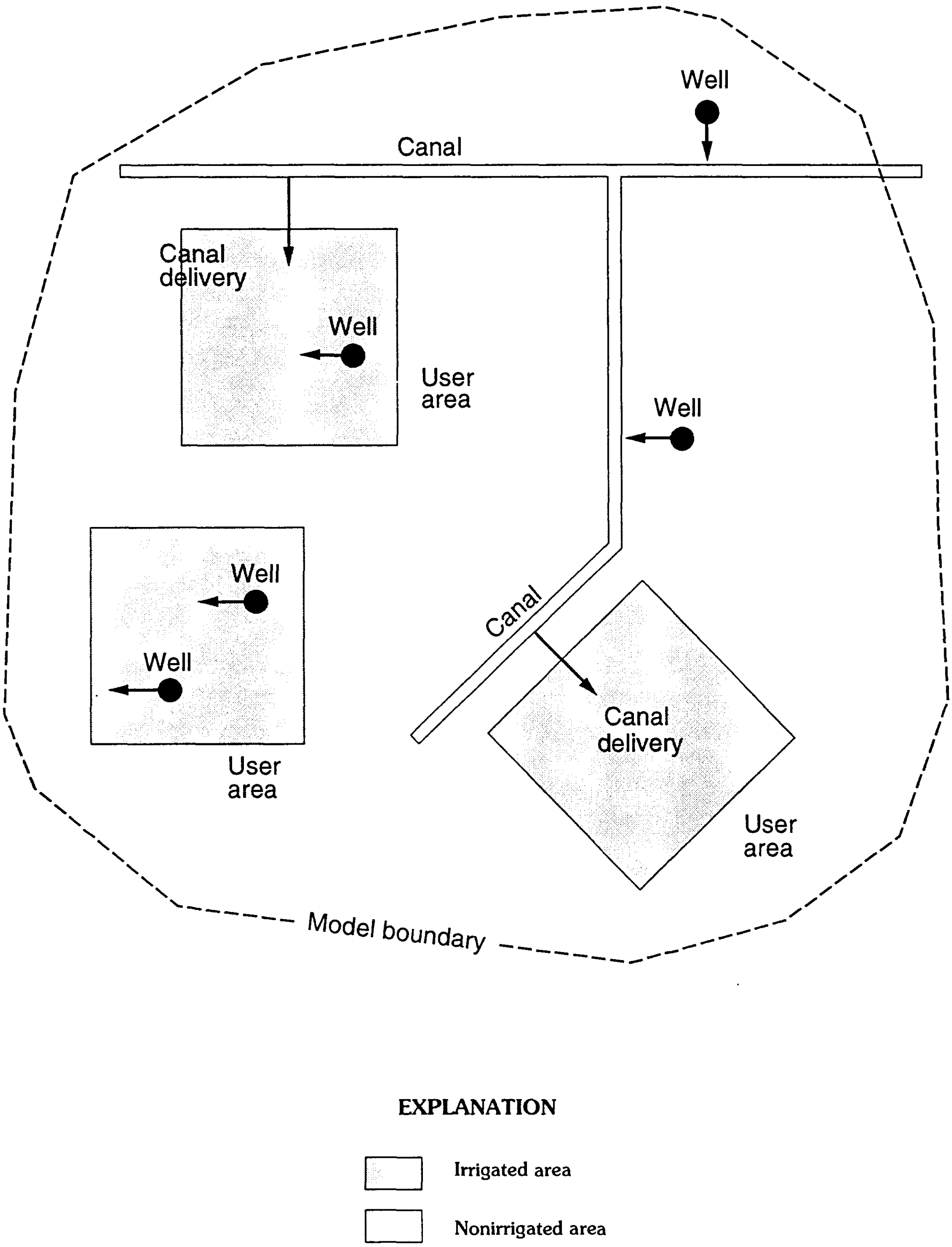


Figure 6. Representation of an irrigated agricultural system in subroutine *WATER*.

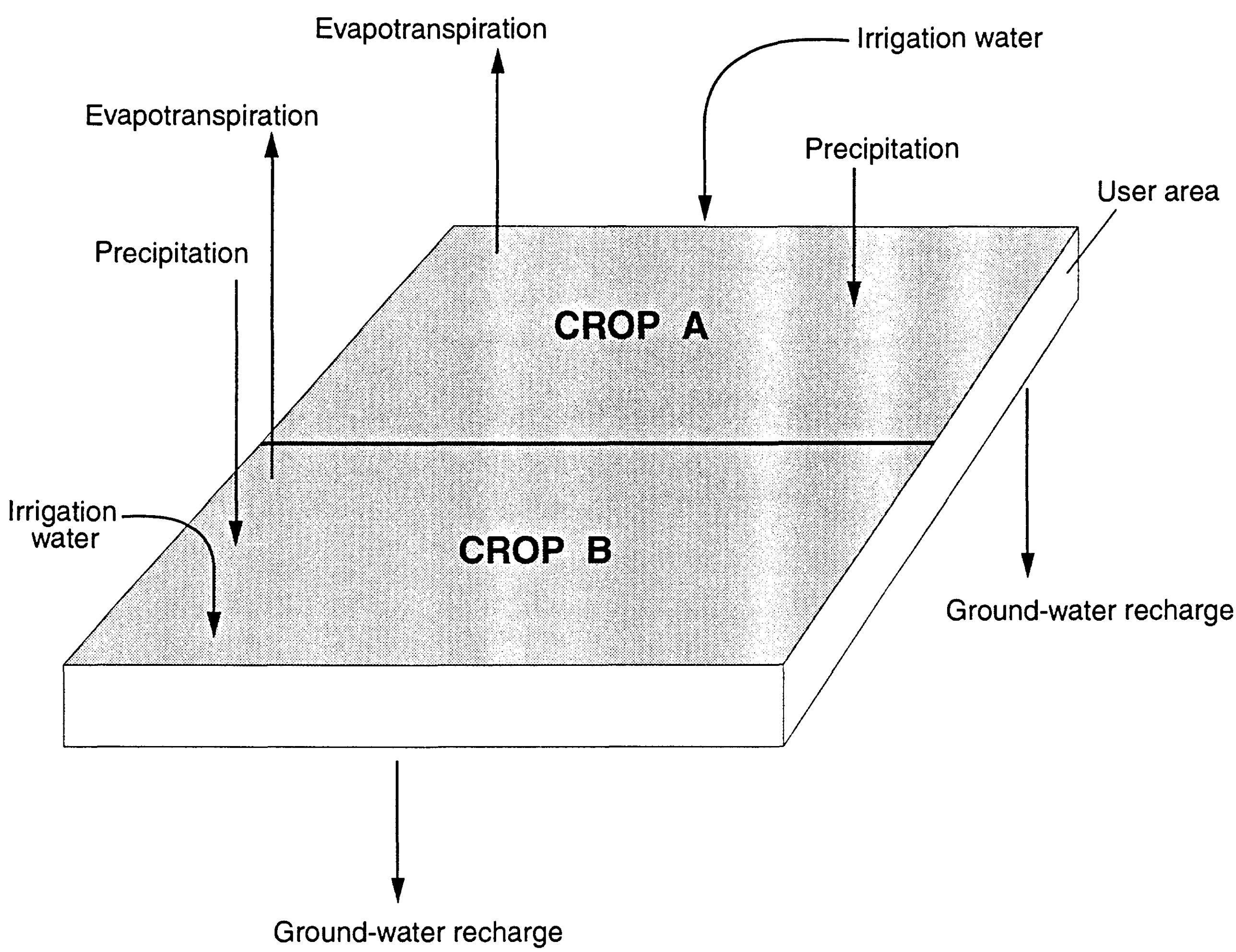


Figure 7. Representation of a user area in subroutine *WATER*.

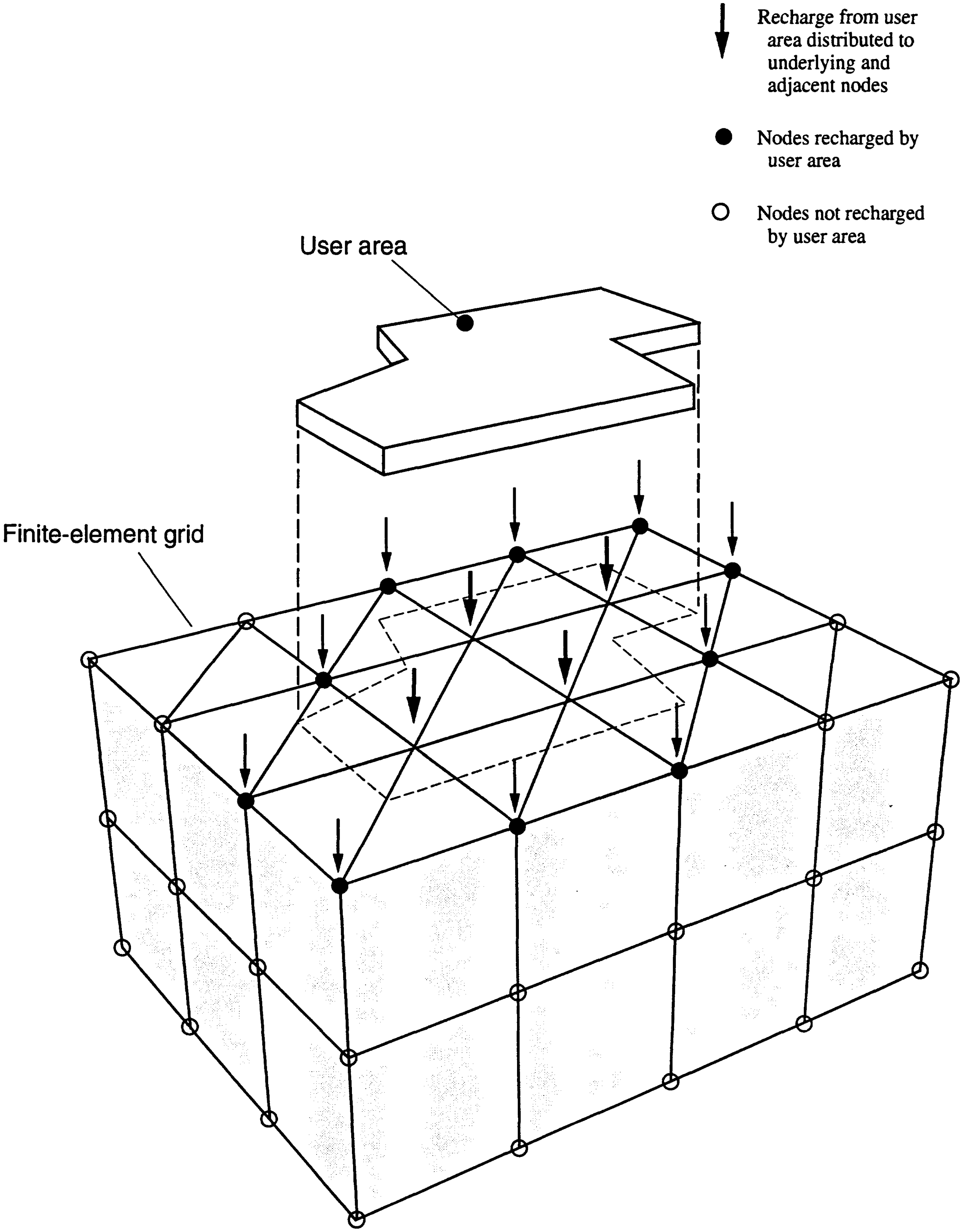


Figure 8. Representation of ground-water recharge in subroutine *WATER*.

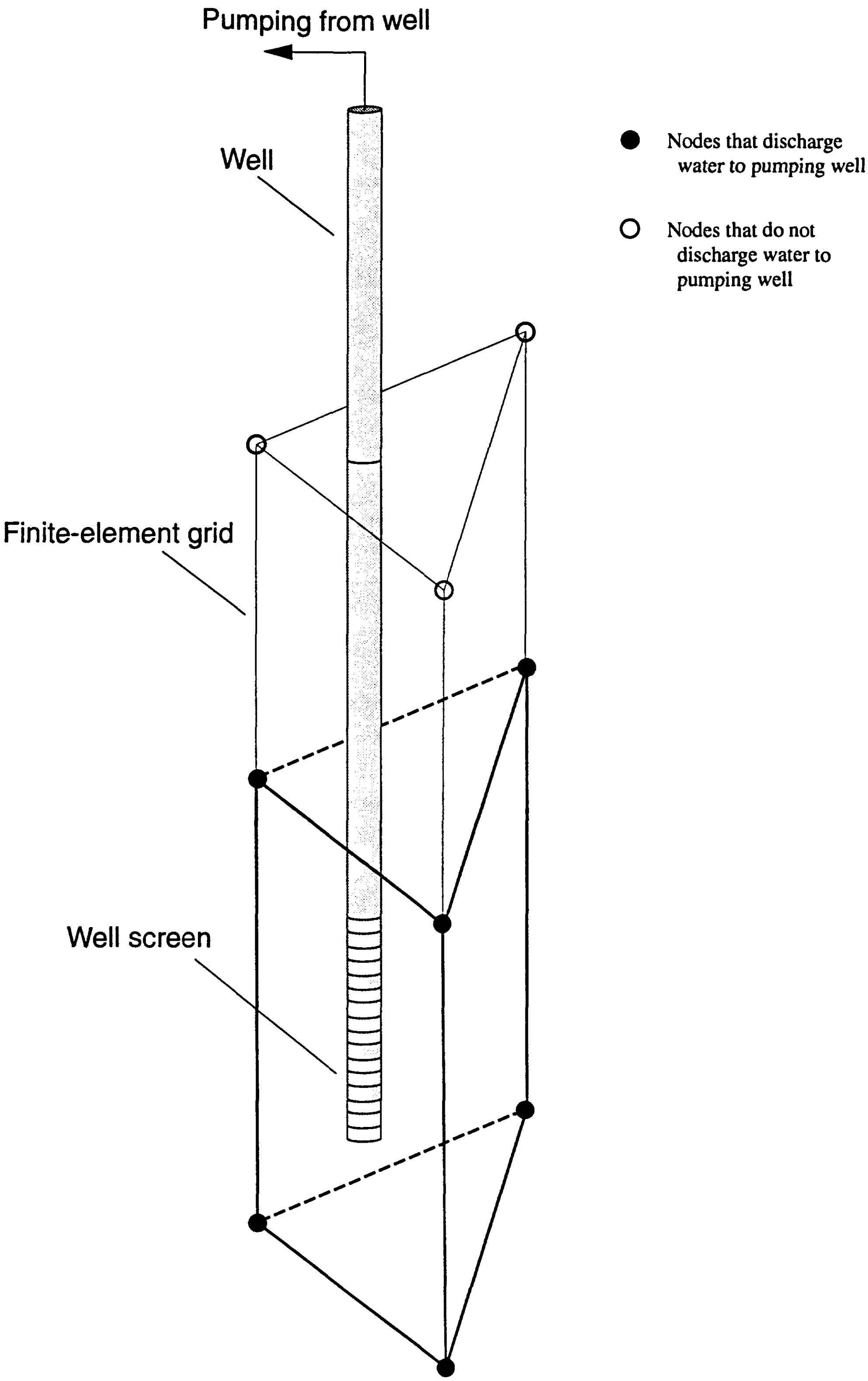


Figure 9. Representation of ground-water pumping in subroutine *WATER*.

3.10.2 Mathematical Basis

Subroutine *WATER* is based on the mass balance of water into and from a user area. That mass balance is given by the relation

$$Q_{sw} + Q_{Pump} - Q_{ET} - Q_{Rech} = Q_{Store}, \tag{3.10-1}$$

where

Q_{sw} is the surface-water delivery to a user area [L^3/t],
Q_{Pump} is the ground-water pumped to a user area [L^3/t],
Q_{ET} is the evapotranspiration from a user area [L^3/t],
Q_{Rech} is the ground-water recharge from the user area [L^3/t], and
Q_{Store} is the rate of change in soil-moisture storage [L^3/t].

3.10.3 Structure of Subroutine

Subroutine *WATER* is structured into two basic blocks. The first block, which is entered by a call to *WATER1*, is for the input and display of data for ground-water pumpage, surface-water deliveries, well locations, user-area boundaries, crop patterns, soil type, and potential evapotranspiration. The second block, which is entered by a call to *WATER2*, distributes ground-water pumpage and surface-water deliveries to the user areas, calculates the actual evapotranspiration through a call to subroutine *LAND*, and distributes ground-water pumpage and recharge to nodes in the finite-element grid.

In the first block, data representing the area to be modeled are organized into 15 files as follows:

1. **Well-site inventory.** The well-site inventory file contains information on the location and construction of wells and on the assignment of wells to particular user areas.

2. **Well-status codes.** The well-status code file contains information on the assignment of a well to a status group, which may change with time.

3. **Monthly well pumping.** The monthly pumping file contains information on monthly pumpage from individual wells.

4. **Well-pumping construction.** The pumping-construction file is used to set up a particular pumping scenario for a simulation. The pumping-construction file allows the user to set up a scenario by selecting pumpage listed in the monthly pumping file for individual wells or by selecting a pumpage value listed in the well-construction file for a group of wells.

5. **User inventory.** The user-inventory file contains information on the location and other characteristics of individual user areas.

6. **Monthly canal delivery.** The monthly delivery file contains information on monthly surface-water deliveries to individual user areas.

7. **Canal-delivery construction.** The canal-delivery construction file is used to set up a particular delivery scenario for a simulation. The canal-delivery construction file allows the user to set up a scenario by selecting delivery values from the monthly delivery file for individual user areas or by assigning a total delivery to a group of user areas with the same user-type code specified in the user-inventory file.

8. **Crop inventory.** The crop-inventory file contains information on the crops within individual user areas.

9. **Rooting depth.** The rooting-depth file contains information on the rooting depths of crops in the crop-inventory file.

10. **Monthly precipitation.** The monthly precipitation file contains monthly precipitation for a station. The precipitation station is assigned to user areas and adjusted by a factor in accordance with a specification in the user-inventory file.

11. **Monthly evapotranspiration.** The monthly evapotranspiration file contains information on the potential evapotranspiration for the crops in the crop-inventory file.

12. **Consumption construction.** The consumption-construction file is used to set up a particular evapotranspiration scenario for a simulation. The consumption-construction file allows the user to set up a scenario by selecting potential evapotranspiration values from the monthly evapotranspiration file and crop distribution from the crop-inventory file.

13. **Well-pumping destination.** The pumping-destination file identifies wells that pump into the surface-water distribution system. The identification is by the well-status code specified in the well-status code file.

14. **Recharge factor.** The recharge-factor file contains information on the proportion of deep percolation that becomes ground-water recharge. If the factor does not equal 1 for a particular area, only part of the deep percolation becomes ground-water recharge, and the remaining deep percolation is presumed to be removed from further interaction with the ground-water system. This would occur if part of the deep percolation entered an agricultural drainage system.

15. **Well exclusion.** The well-exclusion file identifies wells that are to be excluded from a simulation.

The relation of these 15 files to subroutine *WATER* and *LAND* is shown in figure 10.

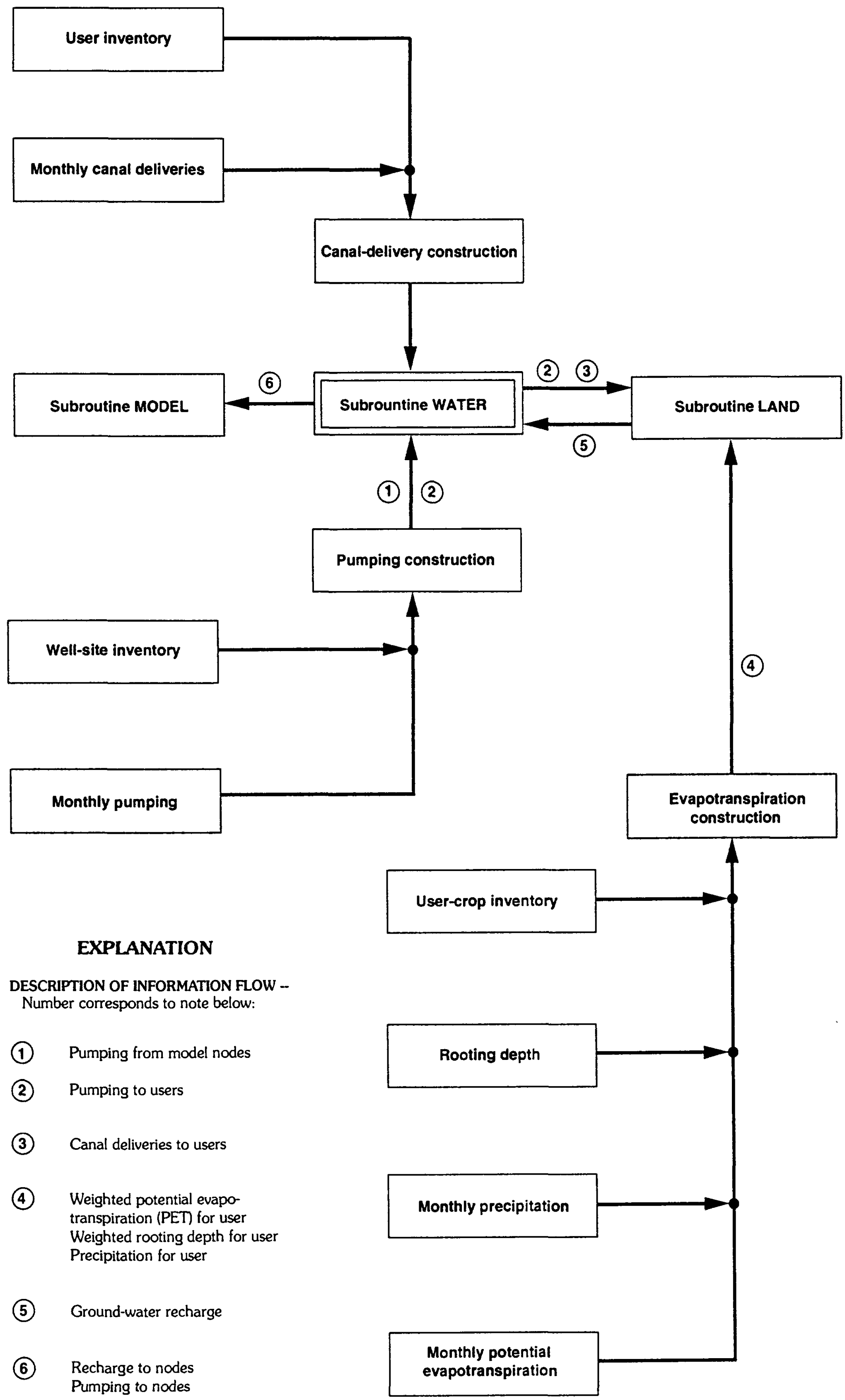


Figure 10. Files used in subroutine *WATER*.

In the second block of subroutine *WATER*, two vectors represent the distribution of ground-water recharge and pumpage to nodes within the finite-element grid. These vectors are $\{Q_R\}$ and $\{Q_P\}$ (ground-water recharge and the ground-water pumpage, respectively). Cumulative recharge and pumpage through all nodes are given by

$$Q_{WR} = \sum_{i=1}^{n} Q_{Ri} \tag{3.10-2}$$

and

$$Q_{WP} = \sum_{i=1}^{n} Q_{Pi} \, , \tag{3.10-3}$$

where

Q_{WR} is the cumulative recharge $[L^3/t]$,
Q_{Ri} is the recharge at the node i $[L^3/t]$,
Q_{WP} is the cumulative pumpage $[L^3/t]$, and
Q_{Pi} is the pumpage at node i $[L^3/t]$.

3.11 Subroutine *LAND*

3.11.1 Background

Subroutine *LAND* is used to calculate evapotranspiration and ground-water recharge that result from precipitation on cropped and noncropped areas and from irrigation applications on cropped areas.

3.11.2 Mathematical Basis

Evapotranspiration and ground-water recharge for a user area depend on the depth of water that infiltrates into soils. That infiltration, in turn, depends on (1) the acreage of the user area, (2) the effective precipitation on the user area, (3) the water delivered to the user area by surface-water deliveries and ground-water pumpage, (4) the delivery-system losses within a user area, and (5) the tailwater losses from fields.

3.11.2.1 Effective Precipitation

Effective precipitation is precipitation that infiltrates into the soil. In subroutine *LAND*, effective precipitation is calculated using a method by Jensen (1983) by which effective precipitation is the sum of increments of effective precipitation. In turn, the increments of effective precipitation are determined as a proportion of an increment of actual precipitation (table 1). The proportions range from 95 percent for the increment of precipitation less than 1 inch during a month to 5 percent for increments of precipitation of more than 7 inches during a month. This method of calculating effective precipitation is a simple approach that ignores much of the complexities of precipitation runoff.

Table 1. Effective precipitation based on increments of monthly rainfall

Precipitation increment range (inches)	Effective precipitation (percent)
0-1	95
1-2	90
2-3	82
3-4	65
4-5	45
5-6	25
6+	5

3.11.2.2 Infiltrated Water

Water that infiltrates the soil is given by the relation

$$Q_I = Q_{sw} + Q_{Pump} + A_u P_{Eff} - Q_{Tail} - Q_{Canal} , \qquad (3.11\text{-}1)$$

where

Q_I is the rate of infiltration within the user area $[L^3/t]$,
Q_{SW} is the surface-water deliveries into the user area $[L^3/t]$,
Q_{Pump} is the ground-water pumpage within the user area $[L^3/t]$,
A_u is the area of the user area $[L^2]$,
P_{Eff} is the effective precipitation [L/t],
Q_{Tail} is the tailwater losses from field $[L^3/t]$, and
Q_{Canal} is the conveyance loss within the user area $[L^3/t]$.

The conveyance loss, in turn, is given by the relation

$$Q_{Canal} = Q_{SW} f_{Canal} , \qquad (3.11\text{-}2)$$

where f_{Canal} is the proportion of surface-water deliveries that become conveyance losses [dimensionless]. The tailwater loss is given by the relation

$$Q_{Tail} = (Q_{sw} + Q_{Pump} - Q_{Canal}) f_{Tail} , \qquad (3.11\text{-}3)$$

where f_{Tail} is the proportion of the water that is applied to a field that becomes tailwater [dimensionless].

3.11.2.3 Distribution of Infiltrated Water

The water that infiltrates the soil eventually can (1) be consumed by evapotranspiration, (2) be stored as soil moisture in the root zone, and (3) past the root zone, flow as deep percolation. This deep percolation, in turn, can recharge to the ground-water system and discharge to agricultural drains. The distribution of infiltrated water is calculated on the basis of a set of rules that depends primarily on the comparison of the infiltration rate and the potential evapotranspiration rate. The distribution is also calculated on the basis of the relation between actual evapotranspiration, potential evapotranspiration, and soil moisture (fig. 11).

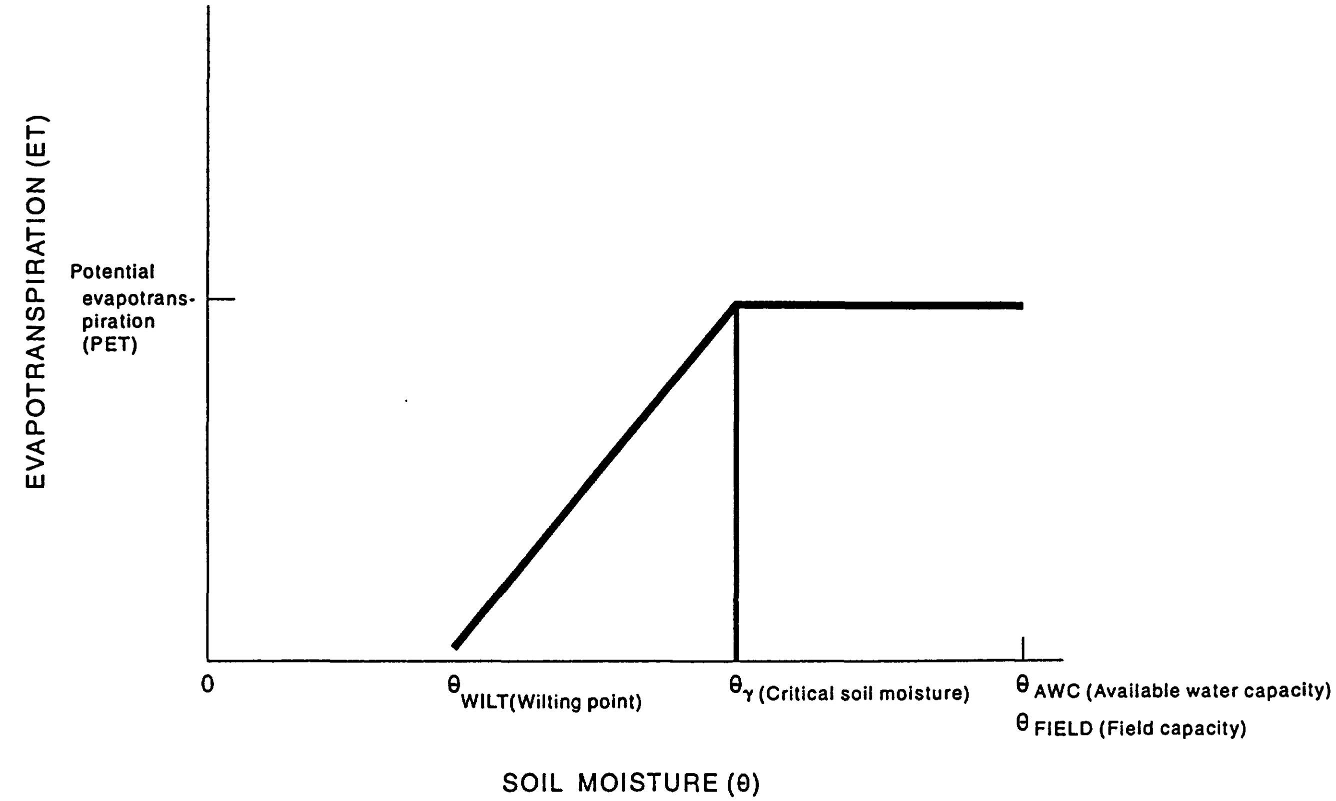


Figure 11. Relation between soil moisture and the evapotranspiration rate.

Actual evapotranspiration *(ET)* ranges from zero to a maximum value equal to the potential evapotranspiration *(PET)* for a range of volumetric soil moisture (θ), from the wilting point (θ_{Wilt}) to the field capacity (θ_{Field}) (fig. 11). The range for soil moisture can be expressed in terms of a scale that equals zero at the wilting point and equals available water capacity (θ_{AWC}) at the field capacity. *ET* increases linearly from a soil moisture equal to the wilting point to soil moisture equal to the critical soil moisture value ($\theta\gamma$). For higher soil moisture, *ET* continues to equal the *PET*. The distribution of infiltrated water is determined in subroutine *LAND* from rules that are triggered principally by the amount of infiltrated water relative to *PET*.

Infiltrated water greater than potential evapotranspiration.--If the infiltrated water is greater than or equal to the *PET*, then

$$\Delta\theta = \frac{(I - PET)}{R_d} \quad (3.11\text{-}4)$$

and

$$\theta_f = \theta_i + \Delta\theta , \quad (3.11\text{-}5)$$

unless θ_f is greater than θ_{AWC}. In which,

$$\theta_f = \theta_{AWC} , \quad (3.11\text{-}6)$$

$$\Delta\theta = \theta_{AWC} - \theta_i , \quad (3.11\text{-}7)$$

and

$$D = I - PET - (\theta_{awc} - \theta_i)\ R_d , \quad (3.11\text{-}8)$$

where

$\Delta\theta$	is the change in soil-moisture storage [dimensionless],
I	is the infiltrated water [L],
PET	is the potential evapotranspiration [L],
R_d	is the root-zone depth [L],
θ_f	is the final soil moisture [dimensionless],
θ_i	is the initial soil moisture at node *i* [dimensionless],
θ_{AWC}	is the available water capacity [dimensionless], and
D	is the deep percolation past the root zone [L].

Infiltrated water less than potential evapotranspiration.-- If the infiltrated water is less than the *PET*, then two conditions can occur relative to the *ET* rate (fig. 12). For the first condition, where the infiltrated water plus the stored soil moisture above $\theta\gamma$ is greater than or equal to the *PET*,

$$I + (\theta_i - \theta_\gamma)R_d \geq PET, \tag{3.11-9}$$

where

θ_γ is the soil moisture below which the actual evapotranspiration is less than the potential evapotranspiration,

then all *ET* will occur at the *PET* rate. For this condition,

$$\Delta\theta = \frac{(I - PET)}{R_d} \tag{3.11-10}$$

and

$$\theta_f = \theta_i + \Delta\theta . \tag{3.11-11}$$

For the second condition, where the infiltrated water plus the stored soil moisture above θ_γ is less than the *PET*, only some *ET* will occur at the *PET* rate, and an *ET* rate less than the *PET* rate will occur for part of the time period. If θ_i is greater than θ_γ,

$$\Delta t = 1 - \frac{[I + (\theta_i - \theta_\gamma)R_d]}{PET} , \tag{3.11-12}$$

Otherwise,

$$\Delta t = 1 - \frac{I}{PET} , \tag{3.11-13}$$

where Δt is the proportion of the time period for which the *ET* rate is less than the *PET* rate [dimensionless].

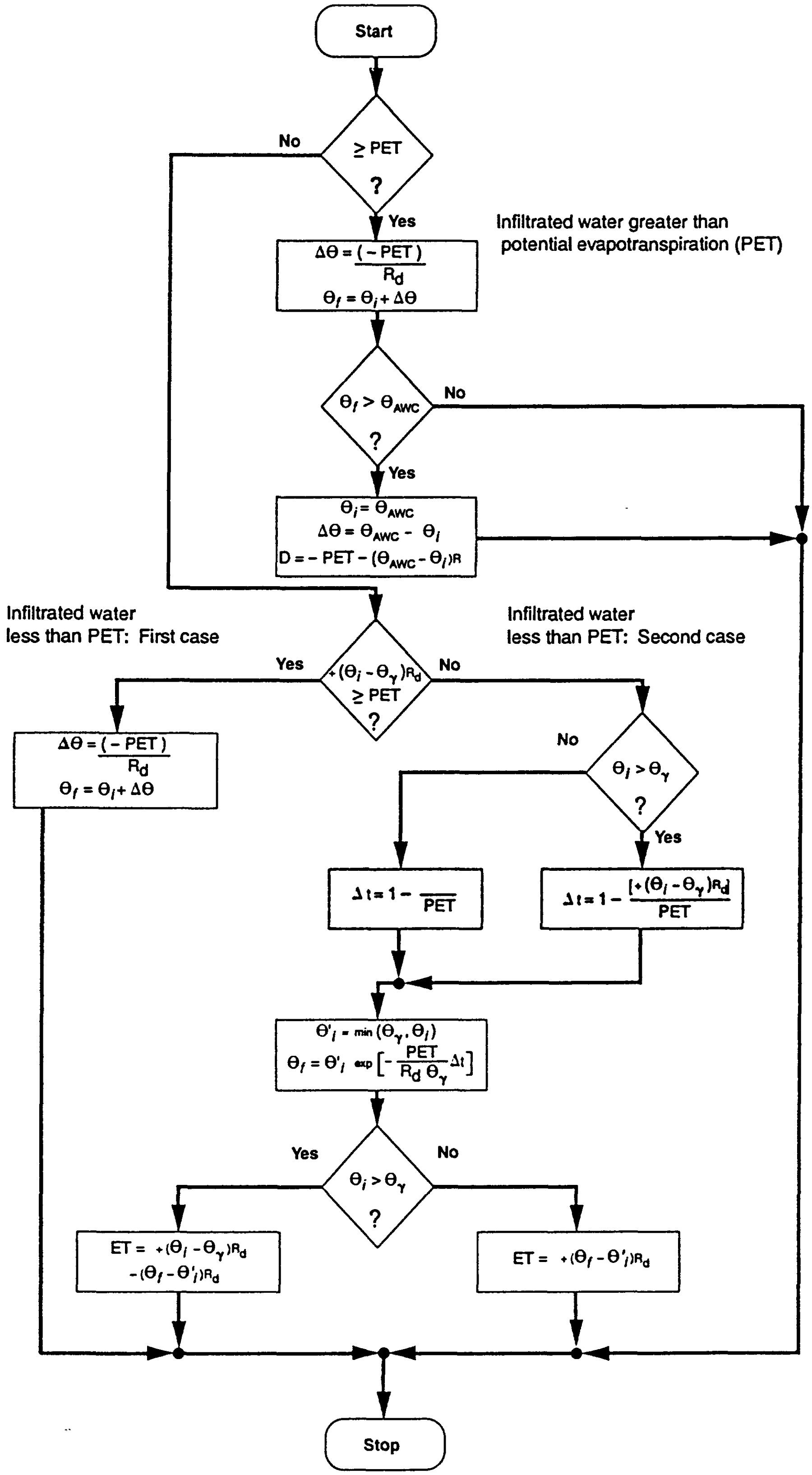


Figure 12. Calculations of evapotranspiration.

During this proportional time period Δt, the change in soil moisture with time is given by the ordinary differential equation

$$\frac{d\theta}{dt} = -\frac{ET}{R_d} = -\frac{1}{R_d}\left(\frac{\theta PET}{\theta_\gamma}\right), \tag{3.11-14}$$

which has the solution

$$\theta = \theta_i' \, exp\left[-\frac{PET}{R_d\theta_\gamma}t\right], \tag{3.11-15}$$

where

$$\theta_i' = min\ (\theta_\gamma, \theta_i), \tag{3.11-16}$$

and where, in Equations 3.11-14 and 3.11-15, the quantities ET and PET are expressed in depth per unit time.

The final soil moisture can be obtained from the relation

$$\theta_f = \theta_i' - \frac{1}{R_d}\int_o^{\Delta t} ETdt. \tag{3.11-17}$$

However, the *ET* rate is given by

$$ET = \theta\frac{PET}{\theta_\gamma}, \tag{3.11-18}$$

which can be substituted into Equation 3.11-17, along with Equation 3.11-15, to obtain

$$\theta_f = \theta_i' - \frac{PET\ \theta_i'}{R_d\ \theta_\gamma}\int_o^{\Delta t} exp\left[-\frac{PET}{R_d\ \theta_\gamma}t\right]dt, \tag{3.11-19}$$

which can be reduced to

$$\theta_f = \theta_i' \exp\left[-\frac{PET}{R_d\,\theta_\gamma}\Delta t\right]. \tag{3.11-20}$$

Total *ET* during the time period is

$$ET = I + (\theta_i - \theta_\gamma)\,R_d - (\theta_f - \theta_i')\,R_d, \tag{3.11-21}$$

if θ_i is greater than θ_γ. Otherwise,

$$ET = I - (\theta_f - \theta_i')R_d. \tag{3.11-22}$$

3.11.2.4 Nonuniform Applications

Subroutine *LAND* assumes that irrigation water is not applied uniformly over a user area to represent this condition. The user area is divided into five water-application zones, where water application is different in each zone according to the uniformity coefficient. Zone 3 receives an average water application, zones 1 and 2 receive more than average applications, and zones 4 and 5 receive less than average applications. These different applications are given by the relation (Jensen, 1983)

$$I_Z = I\left|1 + \frac{5}{6}(1 - U_c)(3 - Z)\right|, \tag{3.11-23}$$

where
- I_Z is the water application in the zone Z [L],
- I is the average water application for all zones [L],
- U_c is the uniformity coefficient [dimensionless], and
- Z is the zone number [dimensionless].

Values in the uniformity coefficient can range from 0.4 to 1. When U_c equals 1, the applied water is distributed equally to each zone. When U_c equals 0.4, the application in zone 1 is 200 percent of the average application, and the application in zone 5 is zero. The value of U_c for a particular nonuniformity of irrigation can be solved with Equation 3.11-23, but the equation would need to be algebraically rearranged. Additionally, values would need to be specified for the average water application, the zone number, and the water application in the zone.

3.11.3 Structure of Subroutine

Subroutine *LAND* is structured into a single block and is entered by a call to *LAND*. A call is made from subroutine *WATER* for each user. Subroutine *LAND* returns the deep percolation of infiltrated water.

3.12 Subroutine *BUDGET*

3.12.1 Background

Subroutine *BUDGET* is used to produce a water budget for the ground-water system at each time step.

3.12.2 Mathematical Basis

The water budget for the ground-water system is given by the relation

$$\sum_{i=1}^{n} Q_i = \Delta S, \qquad (3.12\text{-}1)$$

where

Q_i is a component of inflow or outflow for the ground-water system [L^3/t] and
ΔS is the rate of storage change within the ground-water system [L^3/t].

By expanding Equation 3.12-1, the inflows, the outflows, and the storage changes for the ground-water system are given by the expression

$$Q_P + Q_{WP} + Q_{WR} + Q_B + Q_E + Q_R + Q_V + Q_F = Q'_{Ske} + Q'_{Skv} + Q_{Ske}, \qquad (3.12\text{-}2)$$

where

Q_P is the cumulative discharge through specified-flux nodes from subroutine *FLUX* [L^3/t],
Q_{WP} is the cumulative pumpage from subroutine *WATER* L^3/t],
Q_{WR} is the cumulative recharge from subroutine *WATER* [L^3/t],
Q_B is the cumulative discharge through specified-head nodes from subroutine *CHEAD* [L^3/t],
Q_E is the cumulative evapotranspiration from subroutine *EVAP* [L^3/t],
Q_R is the cumulative discharge through river nodes from subroutine *RIVER* [L^3/t],
Q_V is the cumulative discharge through variable-flux nodes from subroutine *VFLUX* [L^3/t],
Q_F is the cumulative net discharge through fault-node pairs from subroutine *FAULT* [L^3/t],
Q'_{Ske} is the rate of elastic storage change within interbeds from subroutine *SINK* [L^3/t],
Q'_{Skv} is the rate of inelastic storage change within interbeds from subroutine *SINK* [L^3/t], and
Q_{Ske} is the rate of elastic storage change within noninterbeds from subroutine *BUDGET* [L^3/t].

Within subroutine *BUDGET*, the rate of elastic storage change within noninterbeds is determined from the relation

$$Q_{Ske} = \int_{\Omega} (1-p) S_S \frac{\Delta h}{\Delta t} d\Omega + \int_{\Gamma} S_y \frac{\Delta h}{\Delta t} d\Gamma, \tag{3.12-3}$$

where

p is the proportion of the local ground-water system that is occupied by interbeds [dimensionless],

S_s is the specific storage [1/L],

Δh is the average hydraulic-head change [L],

S_y is the specific yield at the water-table surface [dimensionless], and

Δt is the time interval [t].

Equation 3.12-3 is the complement to Equations 3.9-14 and 3.9-15, which relate to the rates of elastic and inelastic storage change within interbeds.

3.12.3 Structure of Subroutine

Subroutine *BUDGET* is structured into a single block and is entered by a call to *BUDGET*. Within that block, the rate of elastic storage change within the noninterbeds and the imbalance of the water budget are calculated and the water-budget components of Equation 3.12-2 are displayed. The water-budget imbalance is the difference between the right-hand and left-hand sides of Equation 3.12-2.

3.13 Subroutine *SHAPE*

3.13.1 Background

Subroutine *SHAPE* is used to construct the elemental matrices $[A^e]$ and $[B^e]$.

3.13.2 Mathematical Basis

The elemental matrices $[A^e]$ and $[B^e]$ are constructed using Equations 3.2-26, 3.2-27, 3.2-28, 3.2-35, and 3.2-36 in Section 3.2.2.3.

3.13.3 Structure of Subroutine

Subroutine *SHAPE* is structured into two basic blocks. The first block is entered by a call to *SHAPE1* and is used to construct the elemental matrices $[A^e]$ and $[B^e]$ using Equations 3.2-26, 3.2-27, and 3.2-28, which relate to volume integrations over tetrahedra elements. In the first block, the matrix $[B^e]$ relates to the specific storage of the ground-water system. The second block, which is entered by a call to *SHAPE2*, is used to construct the elemental matrix $[B^e]$ using Equations 3.2-35 and 3.2-36, which relate to surface integrations over triangular elements. In the second block, the matrix $[B^e]$ relates to the specific yield at the water table.

3.14 Subroutine *BAND*

3.14.1 Background

Subroutine *BAND* is used to solve the symmetric and banded system of linear algebraic equations that are represented by Equation 3.2-38. The left-hand side matrix of that system is

$$[L] \equiv [A] + \frac{1}{\Delta t}[B], \tag{3.14-1}$$

the right-hand side vector is

$$\{R\} \equiv \frac{1}{\Delta t}\,[B]\,\{H_t\} + \{F\}, \tag{3.14-2}$$

and the vector of unknowns is $\{H_{t+\Delta t}\}$. The resulting system of equations to be solved is

$$[L]\,\{H_{t+\Delta t}\} = \{R\}. \tag{3.14-3}$$

3.14.2 Mathematical Basis

The square-root or Cholesky method (Pinder and Gray, 1977, p. 22-23) is used to solve the system of linear equations. This method is based on the fact that, for the symmetric matrix $[L]$, the matrix $[\ell]$ exists such that

$$[L] = [\ell]\,[\ell]^T, \tag{3.14-4}$$

where $[\ell]$ is a lower triangular matrix. A lower triangular matrix is a matrix with coefficients of zero above the diagonal. The general ability to factor a matrix into upper and lower triangular matrices is the essential scheme for solving systems of linear equations. The Cholesky method is one of many such schemes.

<u>**Derivation of solution.**</u>--Setting aside the problem of finding $[\ell]$, Equation 3.14-3 can be solved using a three-step derivation. First, Equation 3.14-4 can be substituted into Equation 3.14-3 to obtain

$$[\ell]\,[\ell]^T\,\{Ht+\Delta t\} = \{R\}. \qquad (3.14\text{-}5)$$

However, an intermediate solution to Equation 3.14-5 can be defined as

$$[\ell]\{X\} = \{R\} \qquad (3.14\text{-}6)$$

where $\{X\}$ is the intermediate solution. Second, because of the triangular structure of $[\ell]$, Equation 3.14-6 can be readily solved for $\{X\}$ by direct back substitution. However, Equation 3.14-6 can be substituted into Equation 3.14-5 to obtain

$$[\ell][\ell]^T\,\{H_{t+\Delta t}\} = [\ell]\{X\}. \qquad (3.14\text{-}7)$$

Third, by premultiplying both sides of Equation 3.14-7 by $[\ell]^{-1}$, Equation 3.14-7 can be reduced to

$$[\ell]^T\,\{H_{t+\Delta t}\} = \{X\}. \qquad (3.14\text{-}8)$$

Again, because of the triangular structure of $[\ell]^T$, Equation 3.14-8 can be solved for $\{H_{t+\Delta t}\}$ by direct back substitution.

Finding lower triangularization.--Returning to the problem of finding *[ℓ]*, expressions for the coefficients of *[ℓ]* can be obtained first by writing expressions for the expansion of Equation 3.14-4 by the definition of matrix multiplication and then by solving for the coefficients *[ℓ]*. The expansion of Equation 3.14-4 yields

$$L_{jj} = \ell_{j1}^2 + \ell_{j2}^2 + \ldots + \ell_{jj}^2 \tag{3.14-9}$$

on the diagonal and yields

$$L_{ij} = \ell_{i1}\ell_{j1} + \ell_{i2}\ell_{j2} + \ldots + \ell_{ij}\ell_{jj} \tag{3.14-10}$$

below the diagonal. Equations 3.14-9 and 3.14-10 can be arranged to obtain

$$\ell_{jj} = \left(L_{jj} - \sum_{k=1}^{j-1} \ell_{jk}^2\right)^{1/2} \tag{3.14-11}$$

on the diagonal and

$$\ell_{ij} = \frac{\left(L_{ij} - \sum_{k=1}^{j-1} \ell_{ik}\ell_{jk}\right)}{\ell_{jj}} \tag{3.14-12}$$

below the diagonal. Equations 3.14-11 and 3.14-12 are used in subroutine *BAND* to extract the lower triangular matrix *[ℓ]* from the matrix *[L]*.

3.14.3 Structure of Subroutine

Subroutine *BAND* is structured into two basic blocks. The first block, which is entered by a call to *BAND1*, is for the lower triangularization of the matrix *[L]*. The second block, which is entered by a call to *BAND2*, is used to obtain $\{H_{t+\Delta t}\}$ using back substitution. $\{H_{t+\Delta t}\}$ is obtained first by solving for the intermediate solution *{X}* using Equation 3.14-6 and then by solving for the solution $\{H_{t+\Delta t}\}$ using Equation 3.14-8.

For some ground-water problems, the matrix *[L]* is identical for each time step of a simulation. This occurs when the governing equations are linear and the time-step length is the same in each time step. The governing equations are linear when the problem being simulated does not include drain nodes, ground-water evapotranspiration, stream-aquifer interactions, or land subsidence. For these conditions, the computational effort in solving Equation 3.14-3 can be reduced by performing the upper triangularization of *[L]* once and by performing the back substitution at each time step.

3.15 Subroutine *SOLVE*

3.15.1 Background

Subroutine *SOLVE* is used to solve the system of linear algebraic equations that are represented by Equation 3.2-38. The left-hand side matrix of that system is

$$[L] \equiv [A] + \frac{1}{\Delta t}[B], \tag{3.15-1}$$

the right-hand side vector is

$$\{R\} \equiv \frac{1}{\Delta t}[B]\{H_t\} + \{F\}, \tag{3.15-2}$$

and the vector of unknowns is $\{H_{t+\Delta t}\}$. The system of equations to be solved is

$$[L]\{H_{t+\Delta t}\} = \{R\}. \tag{3.15-3}$$

3.15.2 Mathematical Basis

The point overrelaxation method is used to solve the system of linear equations. This method is based such that the matrix $[L]$ can be expressed as

$$[L] = [D] - [l] - [u], \tag{3.15-4}$$

where
[D] is diagonal $\{L_{11}, L_{22}, \ldots, L_{nn}\}$,
[l] is a lower triangular matrix, and
[u] is a upper triangular matrix.

The entries of the matrices $[l]$ and $[u]$ are the negatives of the entries of $[L]$ below and above the main diagonal of $[L]$, respectively.

The substitution of Equation 3.15-4 into Equation 3.15-3 yields

$$([D]-[l]-[u])\{H\} = \{R\}, \qquad (3.15\text{-}5)$$

where the time subscript $t+\Delta t$ has been dropped for simplicity. Equation 3.15-5 can be rearranged to obtain

$$[D]\{H\} = ([l]+[u])\ \{H\}+\{R\}. \qquad (3.15\text{-}6)$$

Premultiplication of each term by $[D]^{-1}$ subsequently yields the expression

$$\{H\} = [D]^{-1}([l]+[u])\{H\}+[D]^{-1}\{R\}. \qquad (3.15\text{-}7)$$

From Equation 3.15-7, H_i can be calculated iteratively given estimates H_j for $j \neq I$ using the algebraic expression

$$H_i^{(k+1)} = -\sum_{j=1}^{i-1} \left(\frac{L_{ij}}{L_{ii}}\right) H_j^{(k+1)} - \sum_{j=e+1}^{n} \left(\frac{L_{ij}}{L_{ii}}\right) H_j^{(k)} + \frac{R_i}{L_{ii}}, \qquad (3.15\text{-}8)$$

where k is an iteration counter for each successive update of the vector $\{H\}$. In Equation 3.15-8, $H_i^{(k+1)}$ is calculated using the most current updated vector $\{H\}$.

The rate of convergence for the successive applications of Equation 3.15-8 can be slow; however, the rate can be accelerated by introducing the auxiliary vector $\{\tilde{H}\}$ in the form

$$L_{ii}\,\tilde{H}_i^{(k+1)} = -\sum_{j=1}^{i-1} L_{ij}H_j^{(k+1)} - \sum_{j=i+1}^{n} L_{ij}H_j^{(k)} + R_i. \tag{3.15-9}$$

Then, the quantity $H_i^{(k+1)}$ is a weighted mean of $H_i^{(k)}$ and $\tilde{H}_i^{(k+1)}$ in the form

$$H_i^{k+1} = (1-\omega)H_i^{(k)} + \omega\tilde{H}_i^{(k+1)}, \tag{3.15-10}$$

where ω is the overrelaxation factor for $1 \leq \omega \leq 2$.

Combining of Equations 3.15-9 and 3.15-10 yields the relation

$$H_i^{(k+1)} = H_i^{(k)} + \omega\left\{-\sum_{j=1}^{i-1}\left(\frac{L_{ij}}{L_{ii}}\right)H_j^{(k+1)} - \sum_{j=i+1}^{n}\left(\frac{L_{ij}}{L_{ii}}\right)H_j^{(k)} + \frac{R_i}{L_{ii}} - H_i^{(k)}\right\}. \tag{3.15-11}$$

Equation 3.15-11 is used iteratively in subroutine *SOLVE* to solve Equation 3.15-3. Iterations are continued until convergence is achieved, where convergence is defined by the relation

$$E^{(k+1)} = \max_{i} \left(|H_i^{(k+1)} - H_i^{(k)}| \right), \qquad (3.15\text{-}12)$$

such that

$$\varepsilon^{k+1} \leq \varepsilon_{max}, \qquad (3.15\text{-}13)$$

where ε_{max} is the closure criterion.

3.15.3 Structure of Subroutine

Subroutine *SOLVE* is structured into two basic blocks. The first block, which is entered by a call to *SOLVE1*, reads values for the relaxation factor, the closure criterion, and the maximum allowed number of iterations. The second block, which is entered by a call to *SOLVE2*, is for the iterative application of Equation 3.15-11.

3.16 Subroutine *SEARCH*

3.16.1 Background

Subroutine *SEARCH* is used to calibrate the ground-water model and to identify aquifer and river-bed parameters. The aquifer parameters are horizontal and vertical hydraulic conductivities, specific storage, and specific yield, and the river-bed parameters are river-bed hydraulic conductivities. Subroutine *SEARCH* also identifies parameter values for the aquifer and the river bed, such that the ground-water model best fits the measured ground-water levels and the available prior information on the aquifer and river-bed parameters. Prior parameter values are those developed outside the model calibration. To identify the set of parameter values that produce the best fit, two series of squared residuals are calculated within the subroutine. One series calculates the sum of weighted squared differences between the calculated and measured ground-water levels. The second series calculates the sum of weighted squared differences between the calculated and the prior parameter values. The best fit is defined as the set of parameter values that minimizes the combined sum of the measured and the simulated values.

The weighting within each of the sums of squared residuals is based on the uncertainty of the measured ground-water levels owing to measurement errors and to the lack of representativeness and on the uncertainty of previous estimates of the parameter values. Examples of measurement errors include mistakes in field measurements and uncertainty in the elevation of the land-surface measuring point. Examples of lack of representativeness include water-level measurements that are affected by nearby pumping and water-level measurements that are made in wells that are not representative of the conceptual basis of the ground-water model. Regardless of the source of uncertainty, the uncertainty of individual water-level measurements is expressed in terms of the standard deviation of the water-level measurements in subroutine *SEARCH*.

The uncertainty in the prior estimates of the parameter values is also expressed in terms of the standard deviation. However, the uncertainty of these estimates is expressed in log units, which represent the standard deviation of the log-transform of the parameter estimate. The sources of uncertainty in the prior estimates of parameter values include measurement errors and scale effects. Measurement errors can include errors in interpreting aquifer tests, boring logs, and other type of hydrologic information. Errors can occur when data are translated from one scale to another. For example, data from aquifer tests and boring logs, used to estimate parameter values, are essentially point data relative to the geographic scales represented in the model.

Model calibration is a technically complex procedure, and the structure of subroutine *SEARCH* is based, in part, on the assumption that the user understands the requirements for producing a valid model calibration. Because those requirements are beyond the scope of this report, the user is referred to Anderson and Woessner (1992) for direction on model calibration.

Subroutine *SEARCH* is used to identify aquifer parameters for the ground-water system using the Box-Kanemasu method of nonlinear least squares, which is a maximum a posteriori estimator (Beck and Arnold, 1977). This estimator uses prior information on the aquifer parameters and information on measurement errors. Inclusion of prior parameter information can reduce the variances of the parameter estimates.

Beck and Arnold (1977, p. 269-274) describe the Box-Kanemasu method in terms of a model that is linear in its parameters. However, ground-water models are nonlinear in their parameters and must be linearized to apply the method. This method identifies the sensitivity of the calculated water levels to the parameter values of the model. However, that sensitivity is dependent on particular parameter values, which is the phenomenon that makes ground-water models nonlinear in their parameters. A model is linearized by iteratively recalculating the sensitivities as the search for parameters proceeds.

The Box-Kanemasu method identifies parameter values that are not greatly different than the prior estimates of the parameters values and that reduce the difference between calculated water levels and measured water levels. The balance depends on the uncertainty in the measured water levels relative to the uncertainty in the prior estimates of the parameter values. For example, if the prior estimates of parameter values are highly certain, the method will identify parameter values that are not much different than the prior estimates but at the expense of allowing larger differences between calculated and measured water levels.

3.16.2 Mathematical Basis

3.16.2.1 Modified Box-Kanemasu Method

In subroutine *SEARCH*, aquifer and river-bed parameters are identified that minimize the weighted sum of squared deviations between measured and calculated water levels plus the weighted sum of squared deviations between the prior parameter values and the calibrated parameter rates. This parameter-identification problem is nonlinear, and an iterative process is needed to obtain a solution. Subroutine *SEARCH* uses a modified Box-Kanemasu iterative method.

Update the parameter estimates.--The Box-Kanemasu method (fig. 13) iteratively updates the initial estimate of the parameter vector for the *(k)* iteration by the relation (Beck and Arnold, 1977)

$$\{b\}^{(k)} = \{b\}^{(k-1)} + h^{(k)} \{\Delta b\}^{(k-1)}, \tag{3.16-1}$$

where

$\{b\}$ is a $[n_p \; x \; 1]$ vector of parameters [dimensionless],
k is an iteration counter,
h is a scaler interpolating factor [dimensionless],
$\{\Delta b\}$ is a $[n_p \; x \; 1]$ vector of parameter changes [dimensionless], and
n_p is the number of parameters [dimensionless].

By this relation, the updated estimates equal the previous estimates plus scaled parameter changes. The scaling factor h is selected to optimize the improvement of the parameter estimates, where the optimal value of $h^{(k)}$ minimizes the sum of squared deviations.

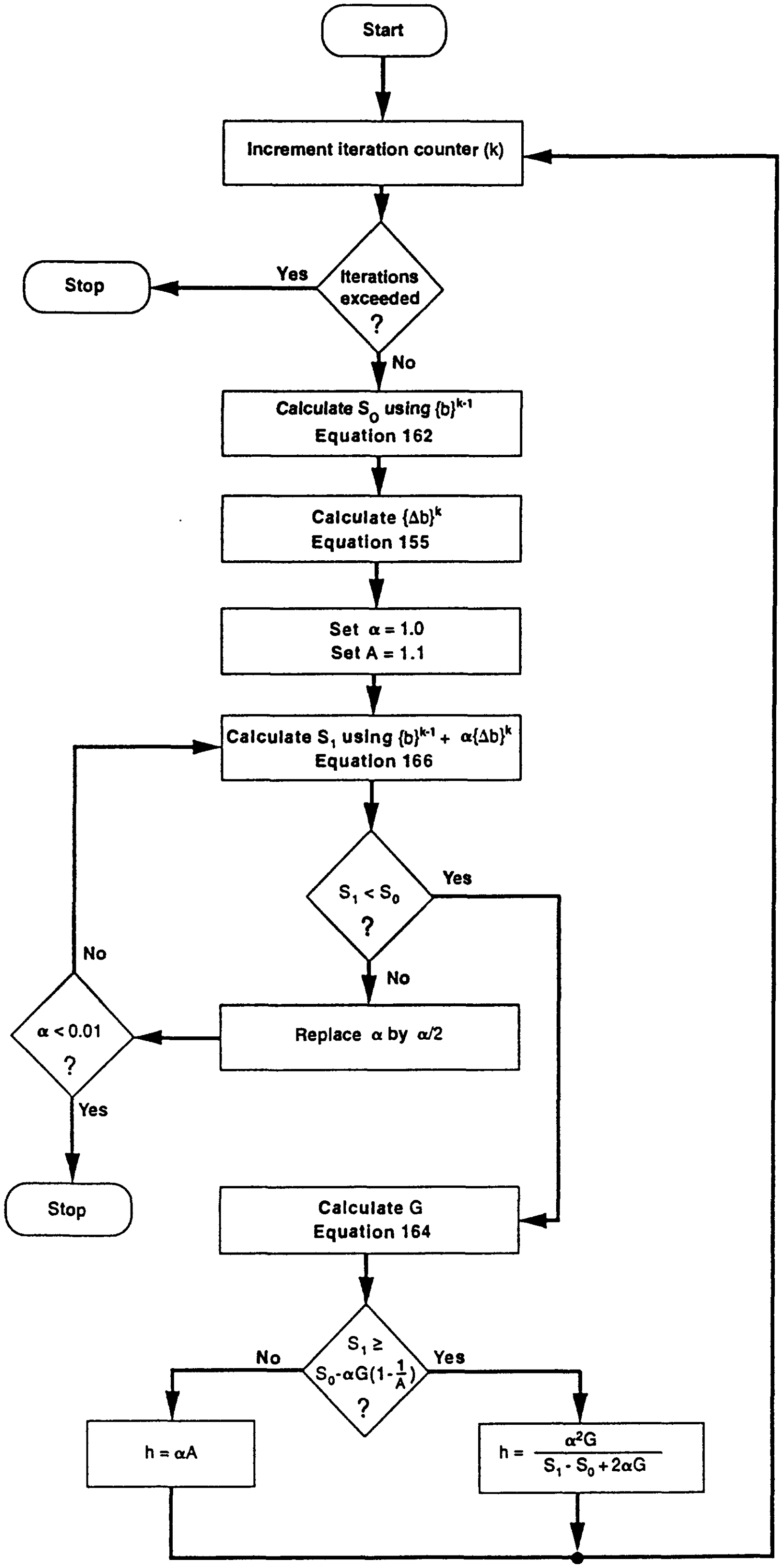


Figure 13. Parameter identification using the Box-Kanemasu method.

The vector of parameter changes is given by the relations (Beck and Arnold, 1977, Equations 6.6.6a and 6.6.6b)

$$\{\Delta b\}^{(k-1)} = [P]^{(k-1)} \left[[X]^{T(k-1)} [W] (\{Y\} - \{\eta\}^{(k-1)}) + [V](\{\mu\} - \{b\}^{(k-1)}) \right], \tag{3.16-2}$$

and

$$[P]^{-1(k-1)} \equiv [X]^{T(k-1)} [W] [X]^{(k-1)} + [V], \tag{3.16-3}$$

where

$[X]$ is a $[n_o \times n_p]$ sensitivity matrix [L],
$[W]$ is the $[n_o \times n_o]$ inverse of the covariance matrix of the water-level measurement errors $[1/L^2]$,
$\{Y\}$ is a $[n_o \times 1]$ vector of measured water levels [L],
$\{\eta\}$ is a $[n_o \times 1]$ vector of calculated water levels [L],
$[V]$ is the $[n_p \times n_p]$ inverse of the covariance matrix of the prior estimates of the aquifer parameters [dimensionless],
$\{\mu\}$ is the $[n_p \times 1]$ parameter vector from prior estimates [dimensionless],
η_o is the number of water-level measurements [dimensionless], and
η_p is the number of aquifer parameters [dimensionless].

A typical element of the sensitivity matrix is given by the relation

$$X_{ij} = \frac{\partial \eta_i}{\partial b_j}. \tag{3.16-4}$$

The elements of the sensitivity matrix are calculated by sequentially perturbing the parameter values to the model and calculating the resulting change in hydraulic heads.

Specification of uncertainty.--In subroutine *SEARCH*, the water-level-measurement error for a particular well is assumed to be independent of the measurement errors for another well. Accordingly, all off-diagonal terms of *[W]* equal zero. The diagonal of *[W]* is given by the relation

$$diag[W] = \left\{\frac{1}{\sigma_Y^2}\right\}, \tag{3.16-5}$$

where σ_Y^2 is the variance of the water-level measurement $[L^2]$. Likewise, the uncertainty in the prior estimate for a particular parameter is assumed to be independent of the uncertainty for another parameter. Accordingly, the off-diagonal terms of *[V]* all equal zero. The diagonal of *[V]* is given by the relation

$$diag[V] = \left\{\frac{1}{\sigma_b^2}\right\}, \tag{3.16-6}$$

where σ_b^2 is the variance of the prior parameter estimate [dimensionless].

Optimizing the scaling factor.--The factor h in Equation 3.16-1 is used to optimize parameter change $\{\Delta b\}$ at each iteration. This is done using the Box-Kanemasu method. This method is described in detail by Beck and Arnold (1977, p. 362-368). However, an overview of the method is given below.

The Box-Kanemasu method optimizes the parameter change $\{\Delta b\}$ at each iteration, finding the value of h that minimizes S by the relation (Beck and Arnold, 1977)

$$S = a_o + a_1 h + a_2 h^2 , \tag{3.16-7}$$

where a_o, a_1, and a_2 are coefficients of the quadratic equation.

This method approximates the weighted sum of squared deviations between measured and calculated water levels plus the weighted sum of squared deviations between the prior estimates and the estimates of the aquifer parameters.

The coefficient a_o is given by the relation

$$a_o = S_o, \tag{3.16-8}$$

where

$$\begin{aligned} S_o = (\{Y\} - \{\eta(\{b\})\})^T [W] (\{Y\} - \{\eta(\{b\})\}) \\ + (\{\mu\} - \{b\})^T [V] (\{\mu\} - \{b\}). \end{aligned} \tag{3.16-9}$$

The coefficient a_1 is given by the relation

$$a_1 = -2G, \tag{3.16-10}$$

where

$$G = [X]^T[W](\{Y\} - \{\eta(\{b\})\}) + [V](\{\mu\} - \{b\}). \tag{3.16-11}$$

The coefficient a_2 is given by the relation

$$a_2 = \frac{1}{\alpha} [S_1 - S_o + 2G\alpha], \tag{3.16-12}$$

where

$$\begin{aligned} S_1 = (\{Y\} - \{\eta(\{b\} + \alpha\{\Delta b\})\})^T[W](\{Y\} - \{\eta(\{b\} + \alpha\{\Delta b\})\}) \\ + (\{\mu\} - \{b\}^T[V](\{\mu\} - \{b\}), \end{aligned} \tag{3.16-13}$$

and where α is a parameter [dimensionless], which is small enough that S_1 is less than S_o.

The procedure by which h is obtained from Equations 3.16-7 to 3.16-13 is shown in figure 13. The iterations are required to find an appropriate value of the parameter α. Initially, the parameter α is set to the value 1.0 (fig. 13). However, if S_1 is not less than S_o, the parameter value is reduced progressively by one-half until S_1 is less than S_o.

3.16.2.2 Interface with Subroutine *MODEL*

Subroutine *SEARCH* passes a dimensionless parameter vector *{b}* to subroutine *MODEL*. The parameter values are determined from the relations

$$K_{xx}(j) = K^{o}_{xx}(j) \exp[b(i)], \tag{3.16-14}$$

where

$K_{xx}(j)$ is the hydraulic conductivity of prismatic element j in the finite-element grid [L/t], which has been assigned to the parameter i,
$K^{o}_{xx}(j)$ is the initial value of $K_{xx}(j)$ [L/t],
$b(i)$ is the parameter with index i in the vector *{b}* [dimensionless], and
j is the element index [dimensionless].

Similar relations apply to K_{yy}, K_{zz}, S_s, S_y, and K_i. Accordingly, the parameter values are determined by the initial values of K_{yy}, K_{zz}, S_s, S_s, and K_i and a factor that multiplies these initial values.

Rather than directly adjusting the parameter values, an exponential function is used as a parameter multiplier. This approach was selected for three reasons. First, the use of a multiplier allows elemental parameters to be adjusted as a group, identified by the element index. For example, a multiplier can be useful when applied to a group of elements that represents a single geologic formation because it allows uniform adjustment. Second, the use of an exponential function prevents the selection of a negative multiplier, which would produce physically implausible values. Third, exponential values reflect the typical log-normal distribution of hydraulic conductivities measured in aquifer materials. Subroutine *SEARCH* assumes that the parameter *{b}* is distributed as a log-normal probability distribution. Accordingly, K_{xx} in Equation 3.16-14 is assumed to be distributed as a log-normal distribution. Additionally, the variance $\{\sigma_b^2\}$ in Equation 3.16-6 is the variance of $ln(b)$.

3.16.3 Structure of Subroutine

Subroutine *SEARCH* is structured into a single block, which is entered by a call to *SEARCH*. The first part of the block is for the input and display of information on the parameter identification problems, including information on prior parameter estimates and measured ground-water levels. In the second part of the block, an iterative search for parameter values is done. Within those iterations, the sensitivity matrix is constructed by making multiple calls to subroutine *MODEL2*. Equation 3.16-2 is then solved for $\{\Delta b\}$, and Equations 3.16-7 through 3.16-13 are used to identify h. Iterations are repeated until the sum of squared deviations cannot be improved further.

3.17 Subroutine *SOLEQU*

3.17.1 Background

Subroutine *SOLEQU* is used to solve the system of linear equations represented by Equation 3.16-2. The left-hand side matrix of that system is

$$[L] \equiv [P], \tag{3.17-1}$$

the right-hand side vector is

$$\{R\} \equiv [X]^T[W](\{Y\} - \{\eta\}), \tag{3.17-2}$$

and the vector of unknowns is $\{\Delta b\}$. The resulting system of equations to be solved is

$$[L]\{\Delta b\} = \{R\}. \tag{3.17-3}$$

3.17.2 Mathematical Basis

The Gauss elimination method is used to solve the system of linear equations. This method is based on the fact that the matrices $[\ell]$ and $[u]$ exist such that

$$[\ell]\,[u] = [L], \tag{3.17-4}$$

where

$[\ell]$ is a lower triangular matrix, and

$[u]$ is an upper triangular matrix.

A lower triangular matrix is a matrix with coefficients of zero above the diagonal, and an upper triangular matrix is a matrix with coefficients of zero below the diagonal. The general ability to factor a matrix into upper and lower triangular matrices is the essential idea of all elimination schemes for solving systems of linear equations. The Gauss method is one of many such elimination schemes.

<u>Derivation of solution</u>.--Setting aside the problem of finding $[\ell]$ and $[u]$, the solution to Equation 3.17-3 can be obtained using a three-step derivation. First, Equation 3.17-4 can be substituted into Equation 3.17-3 to obtain

$$[\ell][u]\{\Delta b\} = \{R\}. \tag{3.17-5}$$

However, an intermediate solution to Equation 3.17-5 is

$$[\ell]\{X\} = \{R\}, \tag{3.17-6}$$

where $\{X\}$ is the intermediate solution. Second, because of the triangular structure of $[\ell]$, Equation 3.17-6 can readily be solved for $\{X\}$ by direct back substitution. However, Equation 3.17-6 can be substituted into Equation 3.17-5 to obtain

$$[\ell][u]\,\{\Delta b\} = [\ell]\{X\}. \tag{3.17-7}$$

Third, by premultiplying Equation 3.17-7 by $[\ell]^{-1}$, Equation 3.17-7 can be reduced to

$$[u]\{\Delta b\} = \{X\}. \tag{3.17-8}$$

Again, because of the triangular structure of $[u]$, Equation 3.17-8 can be used to solve $\{\Delta b\}$ by direct back substitution.

Finding lower and upper triangularizations.--Returning to the problem of finding *[ℓ]* and *[u]*, expressions for the coefficients of *[ℓ]* and *[u]* can be obtained first by writing expressions to expand Equation 3.17-4 using matrix multiplication and then by solving for the coefficients of *[u]* and *[ℓ]*. The expansion of Equation 3.17-4 yields

$$L_{ij} = \sum_{k=1}^{m} \ell_{ik} u_{kj} \tag{3.17-9}$$

where

m = min (i,j).

By letting $u_{ii} = L_{ii}$, equation 3.17-9 can be rearranged to obtain

$$u_{ij} = L_{ij} - \sum_{k=1}^{i-1} \ell_{ii} u_{kj} \tag{3.17-10}$$

where $j \geq i$. As $u_{ii} = \ell_{ii}$, it is necessary that $\ell_{ii} = 1$; for $j \leq i$, ℓ_{ij} becomes

$$\ell_{ij} = \frac{L_{ij} - \sum_{k=1}^{j-1} \ell_{ik} u_{kj}}{u_{jj}}. \tag{3.17-11}$$

3.17.3 Structure of Subroutine

Subroutine *SOLEQU* is structured into a single block, which is entered by a call to *SOLEQU*. Within this block, the matrix *[L]* is factored into upper and lower triangular matrices using Equations 3.17-10 and 3.17-11. Then, the solution $\{\Delta b\}$ is obtained by back substitution, which is done first by solving for the intermediate solution $\{X\}$ using Equation 3.17-6 and then by solving for the solution $\{\Delta b\}$ using Equation 3.17-8.

3.18 Subroutine *FMERGE*

3.18.1 Background

Subroutine *FMERGE* is used to construct input files from file segments. For example, the input file to subroutine *MODEL* includes data on the coordinates of the grid nodes, the specification of the nodes that makes up the grid elements, the aquifer parameter values, and other data. If the data on the nodal coordinates are in a particular file, the data on elements in a second file, the data on aquifer parameters in a third file, and the remaining data in a fourth file, subroutine F*MERGE* can be used to concatenate these files into the overall input file for subroutine *MODEL.* Likewise, subroutine *FMERGE* can be used to construct other input files.

3.18.2 Mathematical Basis

No calculations are done in subroutine *FMERGE.*

3.18.3 Structure of Subroutine

Subroutine *FMERGE* is structured into a single block. Within that block, the file segments are copied into the input file in a specified order. The file segments are not modified during this process.

4.0 MODEL VALIDATION

FEMFLOW3D was validated in four test problems using solutions obtained by independent methods:

1. Transient, confined, radial flow to a well (Theis, 1935).

2. Transient, unconfined, radial flow to a well (Neuman, 1974).

3. Transient, confined, linear flow to a ditch (Carslaw and Jaeger, 1959).

4. Transient compaction of an interbed owing to water-level changes (Leake and Prudic, 1988).

The following sections describe each of these four validation tests. Validation was done by simulating each test problem using *FEMFLOW3D* and then by comparing the results of the simulation with the results of the independent methods. For the first three test problems, the results of the independent method were obtained using analytical solutions (Theis, 1935; Neuman, 1974; Carslaw and Jaeger, 1959; respectively). For the fourth problem, the results of the independent method were obtained using another model (Leake and Prudic, 1988).

4.1 Transient, Confined, Radial Flow to a Well

4.1.1 Problem Description

For the first problem, *FEMFLOW3D* was used to simulate flow to a fully penetrating well in an infinite confined aquifer; the well was pumped at a constant rate. The simulation was then compared with the Theis (1935) analytical solution, which is described in Lohman (1979). This solution has the form

$$s = \frac{Q}{4\pi KB} W(u), \tag{4.1-1}$$

where

- s is the drawdown in the aquifer at a specified time and distance from the well [L],
- Q is the pumping rate [L^3/t],
- K is the radial hydraulic conductivity of the aquifer [L/t],
- B is the aquifer thickness [L],
- $W()$ is the well function [dimensionless], and
- u is a parameter [dimensionless].

The parameter u is given by the relation

$$u = \frac{r^2 S_s}{4Kt}, \tag{4.1-2}$$

where

- r is the radial distance from the well [L],
- t is the time since the start of pumping [t], and
- S_s is the specific storage of the aquifer [1/L].

4.1.2 Model Simulation

The finite-element grid used for the first test problem (fig. 14) is radially symmetric and represents a 24-degree wedge. The radial length of the grid is approximately 21,600 ft, which is sufficient to avoid adverse boundary effects. The radius of the well is 1.0 ft, and the grid spacing in the radial direction increases by a factor of 1.1 with distance from the well. The vertical thickness of the grid is 100 ft.

This first test problem has one transient time-step period that is divided into 21 geometrically expanding time steps. The initial time step is 0.01 day in duration. The duration of each successive time step is expanded by a factor of 1.5. The total duration of the simulation is about 99.7 days.

The initial and boundary conditions for *FEMFLOW3D* are

1. For initial conditions everywhere, drawdown equals zero.

2. For the boundary condition at a radial distance from the well equal to 1.0 ft, flow toward the well equals the pumping rate for times greater than zero. (The well radius and the inward boundary of the finite-element grid are equal to 1.0 ft.)

3. For the boundary condition at a radial distance from the well equal to 21,600 ft (which is equal to the outward boundary of the finite-element grid), no flow occurs for all times.

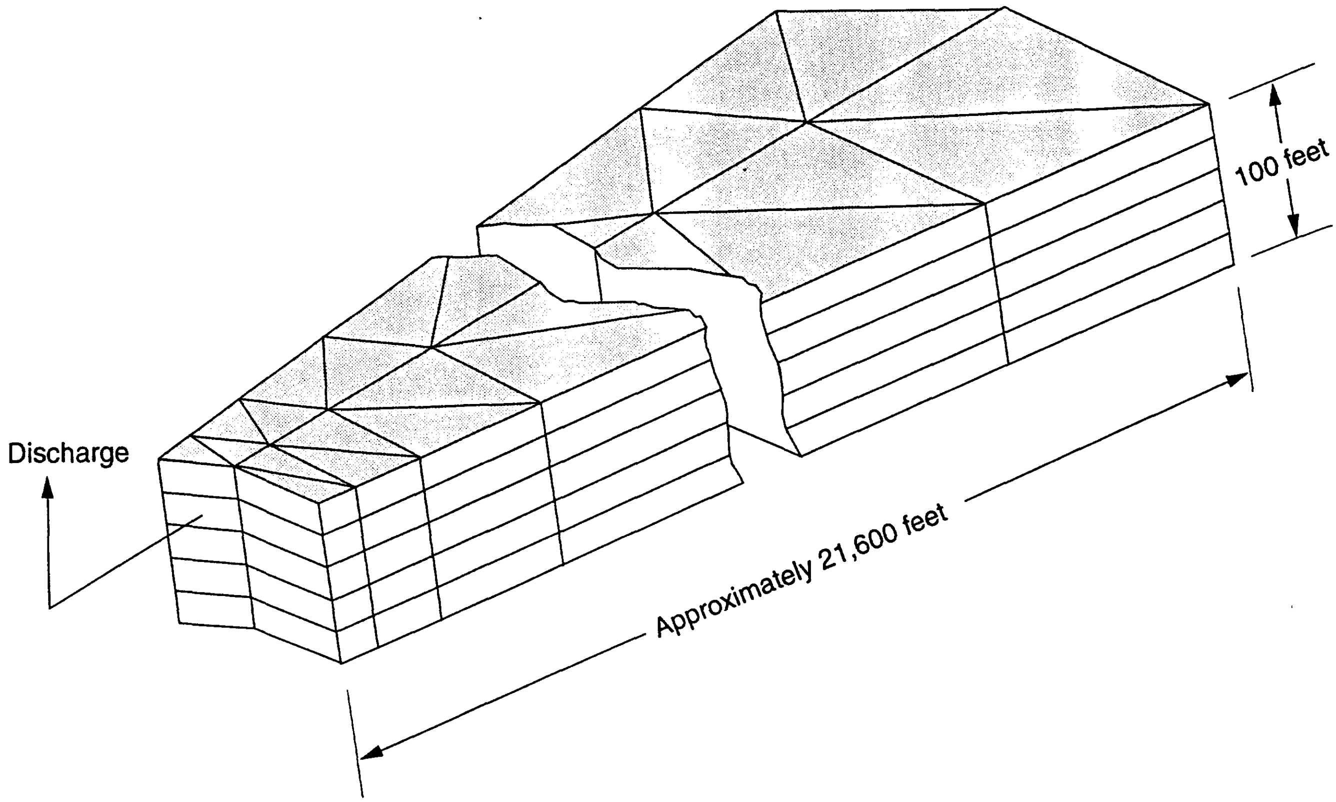


Figure 14. Finite-element grid for Theis problem.

The initial and boundary conditions for the Theis solution are

1. For initial conditions everywhere, drawdown equals zero.

2. For the boundary condition as the radial distance from the well approaches zero, flow toward the well equals the pumping rate for times greater than zero.

3. For the boundary condition as the radial distance from the well approaches infinity, drawdown equals zero for all times.

The hydraulic parameters used in *FEMFLOW3D* are

Hydraulic conductivity *(K)*	10.0 ft/d
Specific storage *(S)*	1.0×10^{-4} ^{1}/ft
Aquifer thickness	100.0 ft
Pumping rate *(Q)*	0.25 ft^3/s

These parameters were also used in the Theis solution, except that the pumping rate was increased to represent the full circumference of the well, while the finite-element grid represents only a 24-degree wedge. Accordingly, the pumping rate is

Pumping rate *(Q)*	3.75 ft^3/s

The input and the output files for *FEMFLOW3D* are on the diskette in the pocket at the back of the report. The input files are THEIS.FLS and THEIS.IN. The output files are THEIS.OUT, THEIS.PLT and THEIS.BUD. The formats of the input files are described in Section 5.0.

4.1.3 Results

The results from the *FEMFLOW3D* simulation and from the Theis solution are shown on a plot of drawdown against $log_{10}(r^2/t)$ (fig. 15). The output file from *FEMFLOW3D* is listed in appendix B for two time steps. The parameter $log_{10}(r^2/t)$ allows various combinations of distance from the well and of elapsed time to be compared on a single graph. *FEMFLOW3D* can very closely replicate the results of the Theis solution for a range of distances and times (fig. 15).

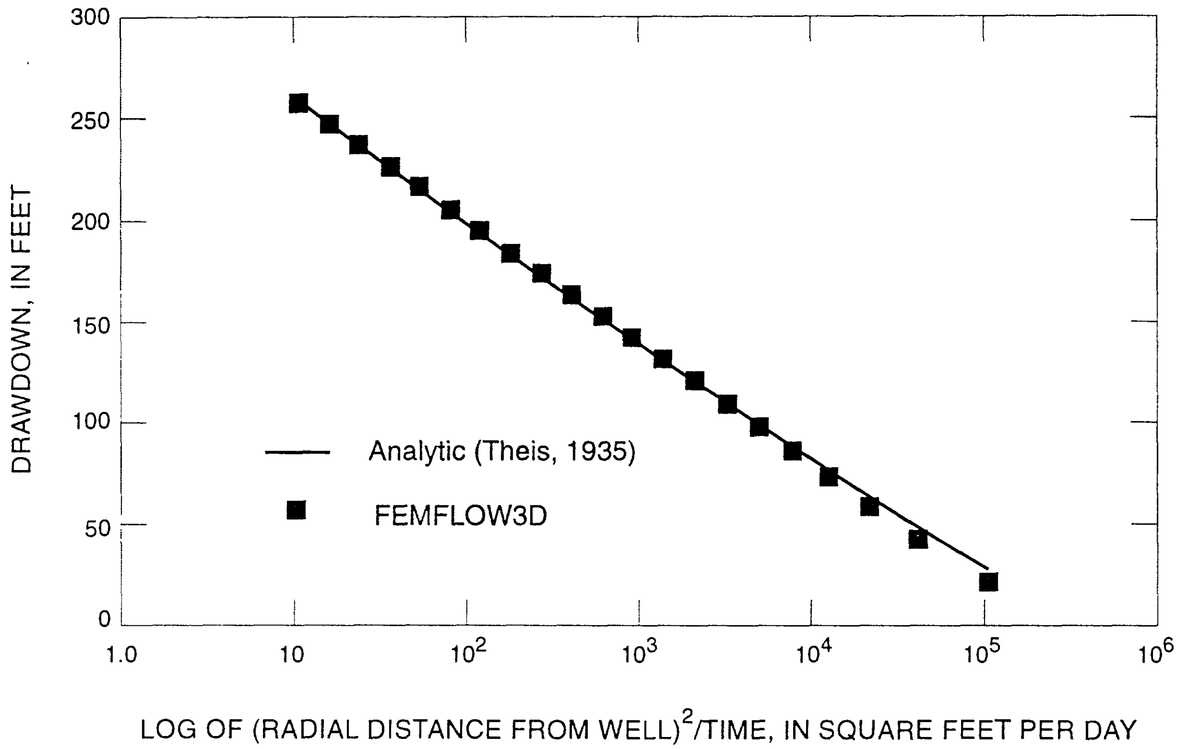


Figure 15. Comparison of *FEMFLOW3D* results with Theis solution.

4.2 Transient, Unconfined, Radial Flow to a Well

4.2.1 Problem Description

For the second test problem, *FEMFLOW3D* was used to simulate flow to a partially penetrating well in an infinite unconfined aquifer; the well was pumped at a constant rate. This problem simulated, in part, the release of stored ground water owing to changes in the position of the water table. The results of the *FEMFLOW3D* simulation were compared with the results of Neuman's (1974) analytical solution, which has the form

$$s = \frac{Q}{4\pi KB} W(\sigma, \beta, z_D, \ell_{D_s}\, d_D, t_s), \tag{4.2-1}$$

where

- s is the drawdown at a specified time, radial distance from the well, and depth within the aquifer [L],
- Q is the pumping rate [L^3/t],
- K is the hydraulic conductivity of the aquifer [L/t],
- B is the thickness [L], and
- $W()$ is the well function [dimensionless].

The dimensionless parameters of the well function are

$$\sigma = \frac{S_s}{S_y}, \tag{4.2-2}$$

$$\beta = \frac{K_z}{K_r}\frac{r^2}{B^2}, \tag{4.2-3}$$

$$z_D = \frac{z}{r}, \tag{4.2-4}$$

$$\ell_D = \frac{\ell}{B}, \tag{4.2-5}$$

$$d_D = \frac{d}{B}, \tag{4.2-6}$$

and

$$t_s = \frac{KBt}{S_s r^2}, \tag{4.2-7}$$

where

- S_s is the specific storage of the aquifer [1/L],
- S_y is the specific yield [dimensionless],
- K_z is the vertical hydraulic conductivity [L/t],
- K_r is the radial hydraulic conductivity [L/t],
- r is the radial distance from the well to the observation point [L],
- z is the height above the bottom of the aquifer to the observation point [L],
- ℓ is the distance from the top of the aquifer to the bottom of the well perforations [L],
- d is the distance from the top of the aquifer to the top of the well perforations [L], and
- t is the elapsed time [t].

4.2.2 Model Simulation

The finite-element grid used for the second test problem (fig. 16) is radially symmetric and represents a 24-degree wedge. The radial length of the grid is 9,974 ft, which is sufficient to avoid adverse boundary effects. The radius of the well is 1.0 ft, and the grid spacing in the radial direction increases by a factor of 1.5 with distance from the well. The vertical thickness of the grid is 160 ft, which represents the initial saturated thickness of the aquifer. The well is perforated within the depth interval between 60 to 160 ft below the initial water-table position.

The second test problem has one transient time-step period that is divided into 21 geometrically expanding time steps. The initial time step is 0.01 day in duration, and the duration of each successive time step is expanded by a factor of 1.5. The total duration of the simulation is about 99.7 days.

The initial and boundary conditions for *FEMFLOW3D* are

1. For initial conditions everywhere, drawdown equals zero.

2. For the boundary condition at a radial distance from the well equal to 1.0 ft, flow toward the well equals the pumping rate for times greater than 0. (The well radius and the inward boundary of the finite-element grid are equal to 1.0 ft.)

3. For the boundary condition at a radial distance from the well equal to 9,974 ft (which is equal to the outward boundary of the finite-element grid), no flow occurs for all times.

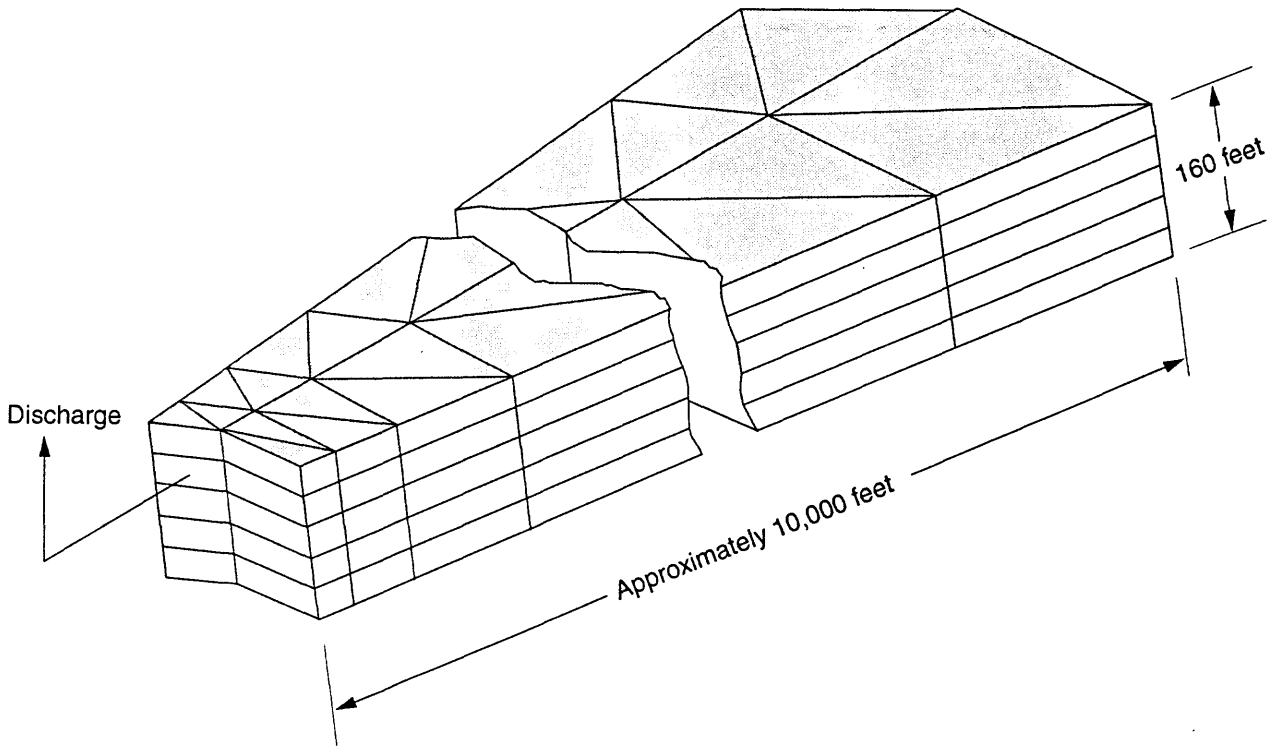


Figure 16. Finite-element grid for Neuman problem.

The initial and boundary conditions for the Neuman (1974) solution are

1. For initial conditions everywhere, drawdown equals zero.

2. For the boundary condition as the radial distance from the well approaches zero, flow toward well equals the pumping rate for times greater than zero.

3. For the boundary condition as the radial distance from the well approaches infinity, drawdown equals zero for all times.

The hydraulic parameters used in *FEMFLOW3D* are

Specific yield (S_y)	0.1 ft {dimensionless]
Specified storage (S_s)	1.0×10^{-6} ^{1}/ft
Radial hydraulic conductivity (K_r)	50.0 ft/d
Vertical hydraulic conductivity (K_z)	50.0 ft/d
Aquifer thickness (B)	160.0 ft
Depth to top of well perforations (d)	60.0 ft
Depth to bottom of well perforations (ℓ)	160.0 ft
Pumping rate (Q)	0.297 ft^3/s
Height above aquifer bottom to observation point (z)	0.0 ft

These parameters were also used in the Neuman solution, except that the pumping rate was increased to represent the full circumference of the area affected by the well, while the finite-element grid represents only a 24-degree wedge. Accordingly, the pumping rate is

Pumping rate (Q)	4.46 ft^3/s

The input and the output files for *FEMFLOW3D* are on the diskette in the pocket at the back of the report. The input files are NEUMAN.FLS and NEUMAN.IN. The output files are NEUMAN.OUT, NEUMAN.PLT, and NEUMAN.BUD. The formats of the input files are described in Section 5.0.

4.2.3 Results

The results of the *FEMFLOW3D* simulation and of the Neuman (1974) solution are shown on a plot of drawdown against $log_{10}(r^2/t)$ (fig. 17), This is the same type of plot used for the Theis test problem shown on figure 15. *FEMFLOW3D* can very closely replicate the results of the Neuman solution for a range of distances and times.

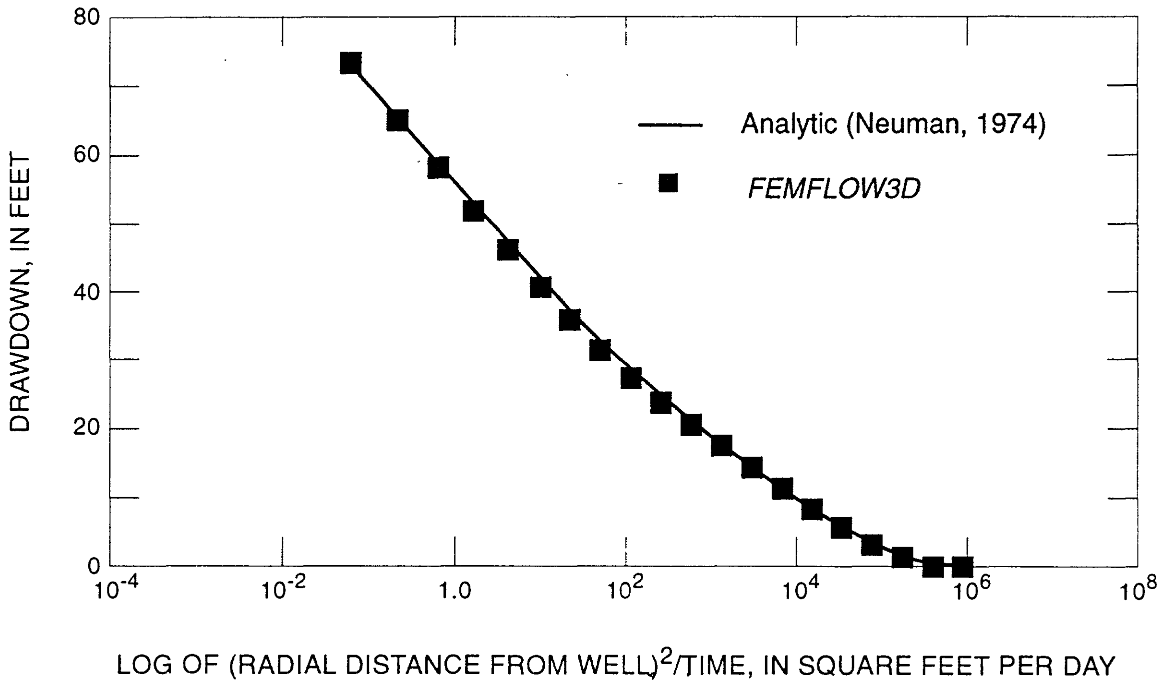


Figure 17. Comparison of *FEMFLOW3D* results with Neuman solution.

4.3 Transient, Confined, Linear Flow to a Ditch

4.3.1 Problem Description

For the third test problem, *FEMFLOW3D* was used to simulate flow to a fully penetrating ditch at the boundary of a semi-infinite aquifer as the result of a step change in the water level within the ditch. The simulation was then compared with the analytical solution of Carslaw and Jaeger (1959), which has the form

$$\Delta h = \Delta h_o \, erfc \left[\frac{x}{2\sqrt{\frac{Kt}{S_s}}} \right], \quad \textbf{(4.3-1)}$$

where

- Δh is the change in hydraulic head in the aquifer at a specified time and distance from the ditch [L],
- Δh_o is the change in the water level in the ditch [L],
- K is the hydraulic conductivity of the aquifer [L/t],
- S_s is the specific storage of the aquifer [^{1}/L],
- x is the distance from the ditch [L], and
- t is the elapsed time [t].

4.3.2 Model Simulation

The finite-element grid for the third test problem is linear, with a length of 5,800 ft, a width of 40 ft, and a vertical thickness of 10 ft (fig. 18).

The third test problem has two time-step periods. Steady-state heads are calculated in the first period. The second transient time period is divided into 40 geometrically expanding time steps. The initial time step is 0.25 day in duration, and the duration of each succesive time step is expanded by a factor of 1.30. The total duration of the transient simulation is about 80 years.

The initial and boundary conditions for *FEMFLOW3D* are

1. For initial conditions everywhere, the change in hydraulic head equals zero.

2. For the boundary condition at the ditch (x equals zero), the change in hydraulic head equals the change in water level in the ditch for times greater than zero.

3. For the boundary condition at a distance of 5,800 ft from the ditch, the change in hydraulic head is calculated using a variable-flux boundary condition. (See Section 3.7 for a discussion of variable-flux boundary conditions.) This boundary condition has the effect of extending the finite-element grid to infinity, where the boundary condition at an infinite distance from the ditch is such that the change in hydraulic head equals zero for all times.

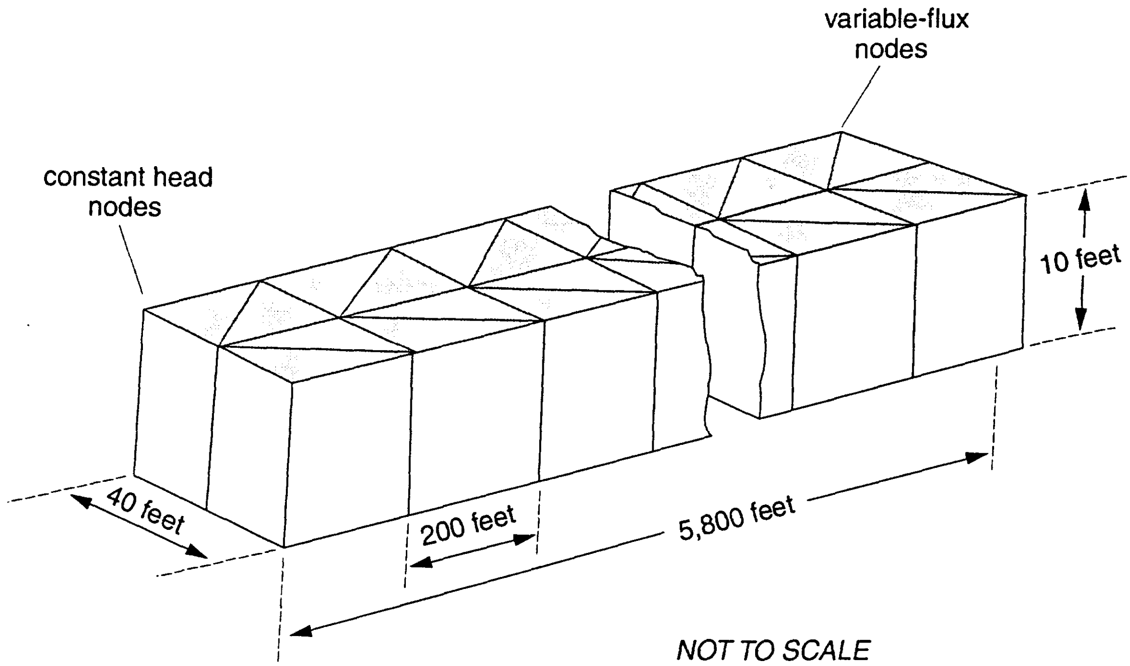


Figure 18. Finite-element grid for Carslaw and Jaeger problem.

The corresponding initial and boundary conditions for the Carslaw and Jaeger (1959) solution are

1. For initial conditions everywhere, the change in hydraulic head equals zero.

2. For the boundary condition at the ditch (x equals zero), the change in hydraulic head equals the change in water level in the ditch for times greater than zero.

3. For the boundary condition at an infinite distance from the ditch, the change in hydraulic head equals zero for all times.

The hydraulic parameters used in *FEMFLOW3D* are

Hydraulic conductivity (K)	100.0 ft/d
Specific storage (S_s)	1.0 x 10^{-3} ^{1}/ft
Water-level step (Δh_o)	10.0 ft

These parameters were also used in the Carslaw and Jaeger (1959) solution.

The input and the output files for *FEMFLOW3D* are on the diskette in the pocket at the back of this report. The input files are CARSLAW.FLS and CARSLAW.IN. The output files are CARSLAW.OUT, CARSLAW.OUT, CARSLAW.PLT, and CARSLAW.BUD. The formats for the input files are described in Section 5.0.

4.3.3 Results

The results from the *FEMFLOW3D* simulation and the Carslaw and Jaeger (1959) solution are given in figure 19, which shows Δh against $log_{10}(t)$ for x equals 5,800 ft. These results indicate that the variable-flux boundary condition implemented in *FEMFLOW3D* replicates the effect of a semi-infinite aquifer. This boundary condition provides a practical alternative to extending the finite-element grid, where stresses reach the boundaries of the finite-element grid and the boundary condition cannot dependably be modeled as a specified-flux or specified-head boundary.

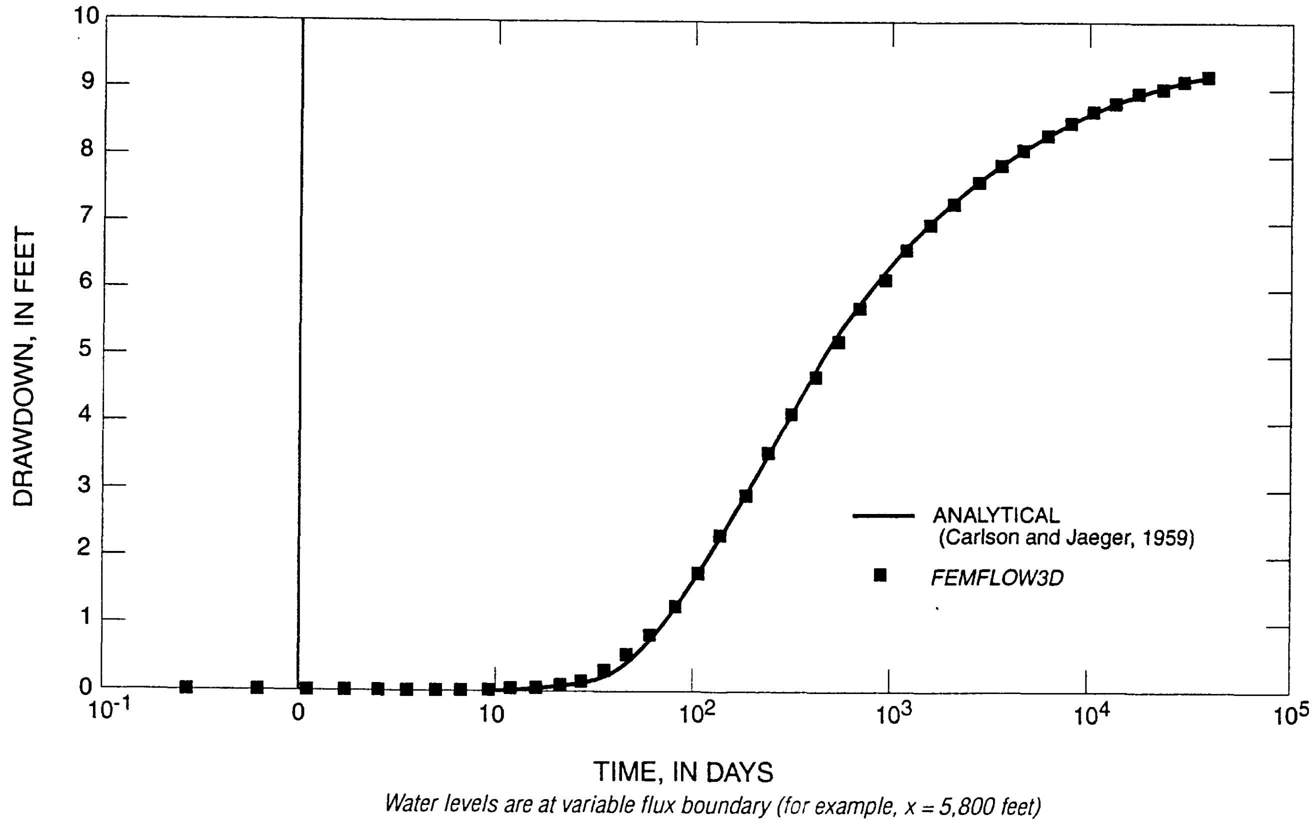


Figure 19. Comparison of *FEMFLOW3D* results with Carslaw and Jaeger solutions.

4.4 Transient Compaction Owing to Water-Level Changes

4.4.1 Problem Description

For the fourth test problem, *FEMFLOW3D* was used to simulate compaction of a compressible interbed owing to changes in the hydraulic head at the boundary of the interbed. This problem, which was taken from Leake and Prudic (1988), simulated the cumulative compaction of 100 identical interbeds. Hydraulic head at the boundaries of the interbeds was specified to decline linearly for 180 days to 10 ft below the starting value. Following the decline, the head recovered linearly for 180 days to the original value. Five successive cycles of declines and recoveries were simulated. The *FEMFLOW3D* simulation was then compared with the solution from the one-dimensional compaction model *COMPAC1* (Helm, 1975; 1984), which is described by Leake and Prudic (1988).

4.4.2 Model Simulation

The finite-element grid used for the fourth test problem (fig. 20) represents one-half of a single 10-foot thick interbed. The grid represents a column within the interbed with a thickness of 5 ft, a width of 10 ft, and a breadth of 10 ft. Each element in the grid has a thickness of 0.5 ft. It was assumed that the interbeds drain from above and below, because the test problem simulates drainage from the one-half interbed from above only. Therefore, the compaction of 100 identical interbeds was obtained by multiplying the compaction for one-half of an interbed by 200 (fig. 20).

The fourth test problem has one transient time-step period that is divided into 180 uniform time steps. Each time step is 10 days in duration. The total duration of the simulation is 1,800 days.

The initial and boundary conditions for *FEMFLOW3D* and *COMPAC1* are

1. For initial conditions everywhere, hydraulic heads and preconsolidation heads equal zero.

2. For the boundary condition at the top surface of the interbed, hydraulic heads are specified (fig. 21) for times greater than zero.

3. For the boundary condition at the center of the interbed, no-flow occurs for all times.

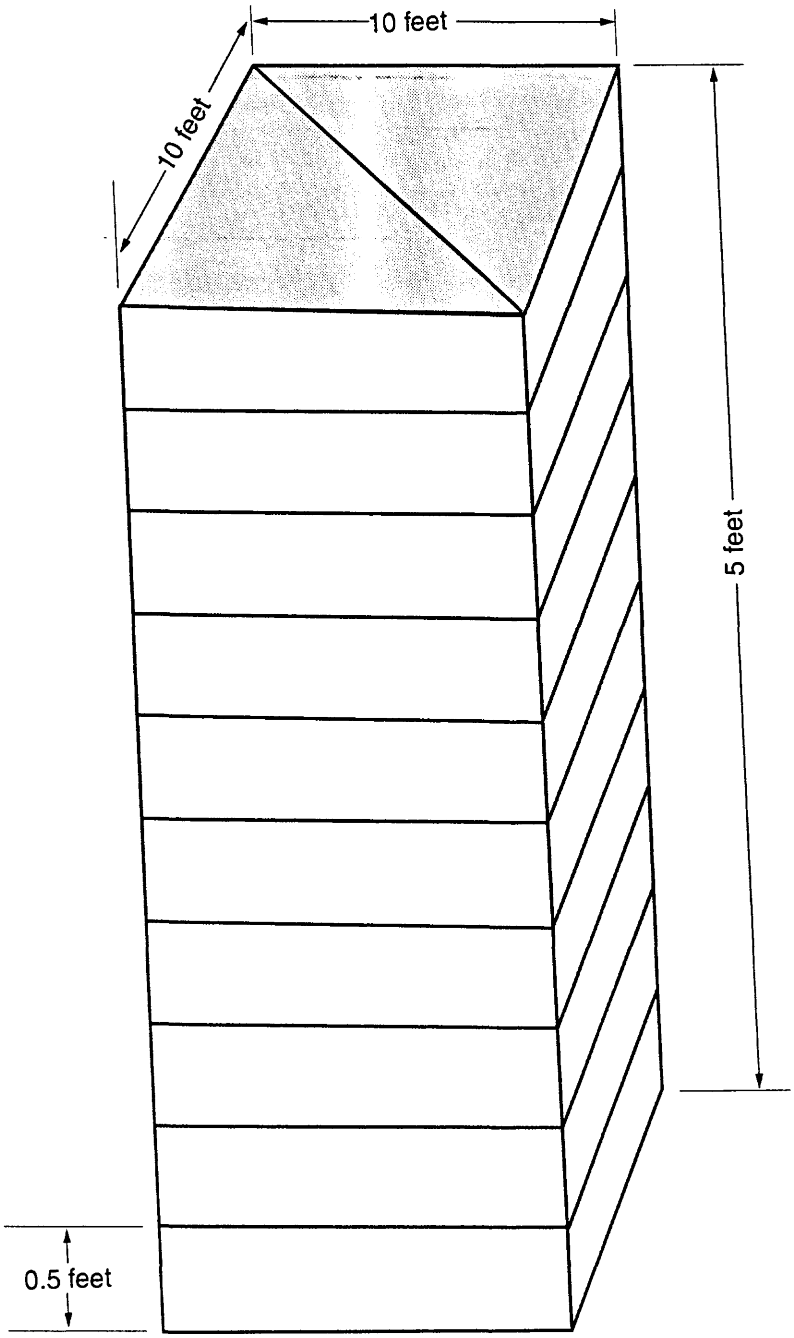


Figure 20. Finite-element grid for compaction problem.

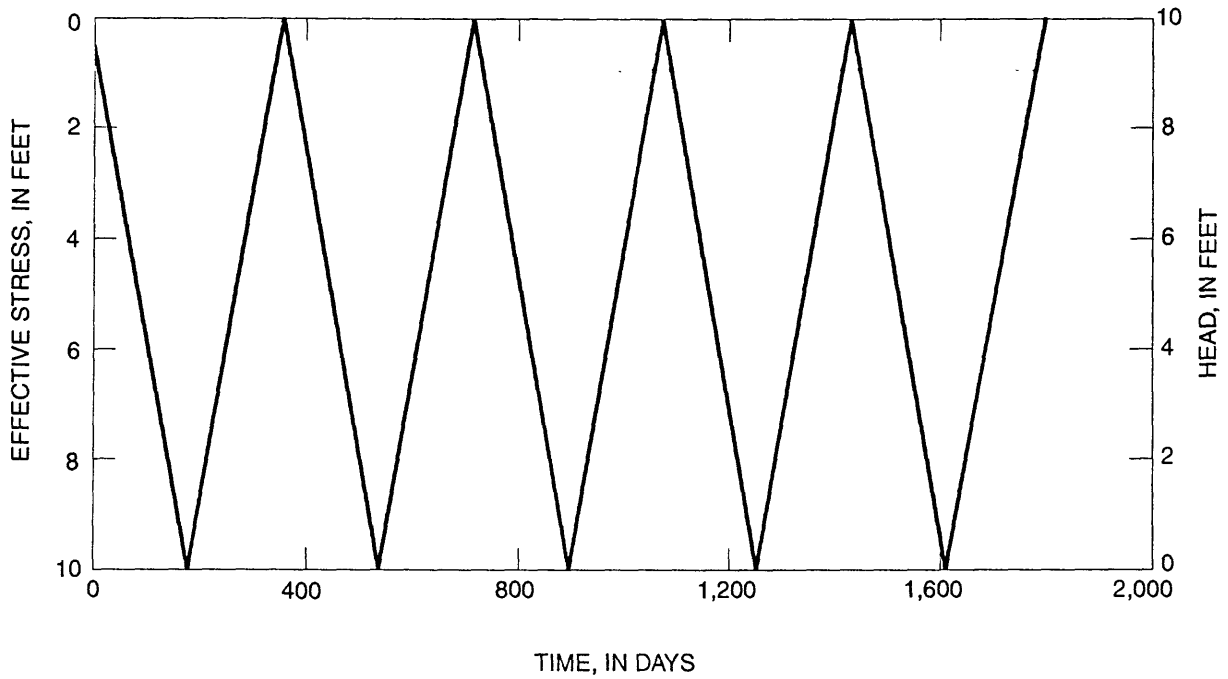


Figure 21. Boundary heads for compaction problem.

The hydraulic parameters for *FEMFLOW3D* and *COMPAC1* are

Vertical hydraulic conductivity *(K_z)*	2.7×10^{-6} ft/d
Elastic specific storage *(S_{Ske})*	1.0×10^{-6} ^{1}/ft
Inelastic specific storage *(S_{Skv})*	1.0×10^{-4} ^{1}/ft
Half-thickness of interbed *(B/2)*	5.0 ft

With these properties, the elastic and the inelastic time constraints (Equation 3.9-5) are 9 days and 900 days, respectively.

The input and the output files for *FEMFLOW3D* are on the diskette in the pocket at the back of the report. The input files are LEAKE.FLS and LEAKE.IN. The output files are LEAKE.OUT, LEAKE.PLT, and LEAKE.BUD. The formats for the input files are described in Section 5.0.

4.4.3 Results

The results from the *FEMFLOW3D* simulation and the results for *COMPAC1* (Leake and Prudic, 1988) are shown on a plot of cumulative compaction with time. *FEMFLOW3D* can very closely replicate the results of the one-dimensional compaction model *COMPAC1* (fig. 22).

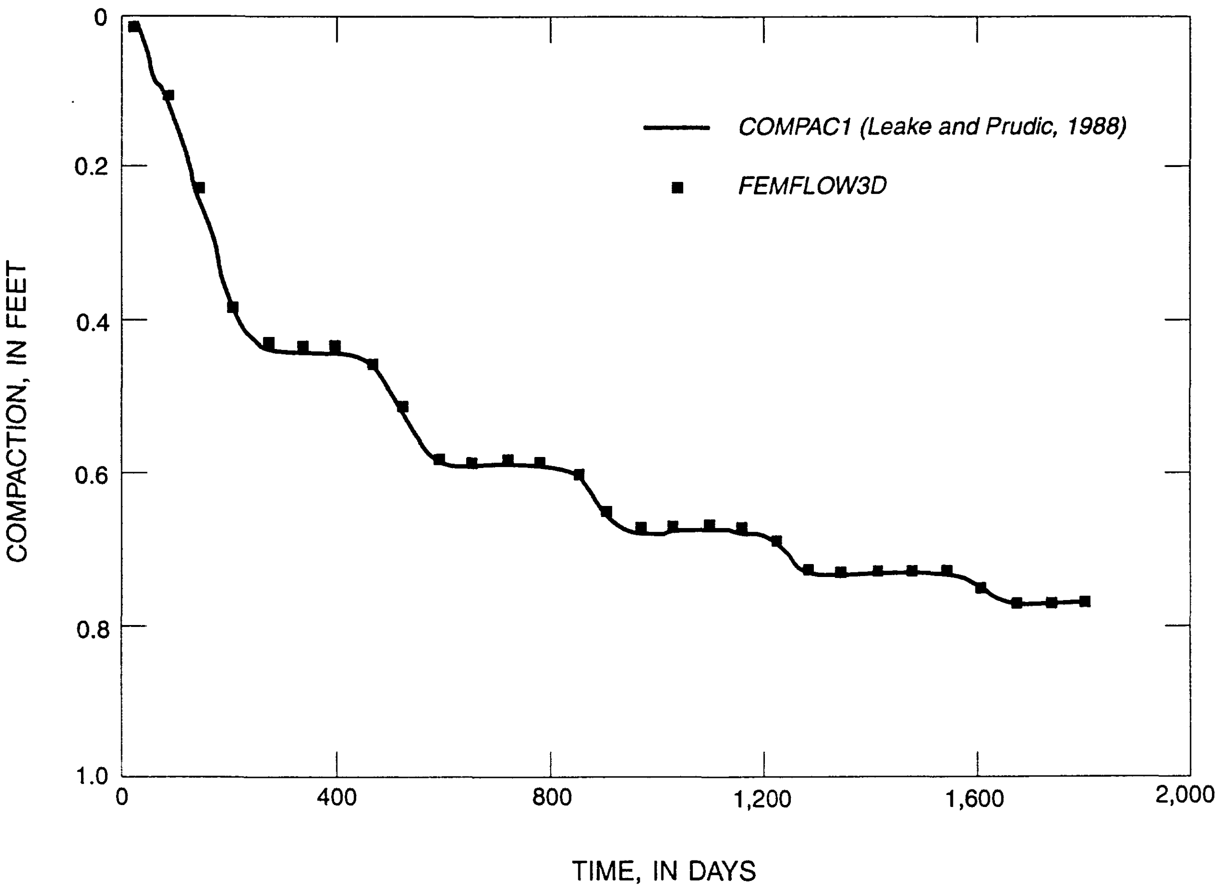


Figure 22. Comparison of *FEMFLOW3D* results with *COMPAC1* results.

5.0 INPUT FORMATS

The inputs to *FEMFLOW3D* are organized into four groups of files, with each group consisting of one or more files. These files are described in subsections 5.1 through 5.4. The first group is the file-specification file, which identifies the input and the output files for a simulation. The second group is the model-input file, which specifies the ground-water problem to be simulated, but excludes inputs relating to the simulation of an irrigated-agricultural system. The third group consists of the irrigated-agricultural system files, which describe the source and fate of irrigation water. The fourth group consists of the parameter-identification files, which specify the parameter identification problem to be solved.

Unless otherwise specified, all files and variables must be included in every simulation and must follow the order and the format specified in this section. Provisions for omitting unnecessary input for subroutines that are not used in any specific application are described in each of the subsections.

Section 5.0 concludes with a description of the procedure for specifying array dimensions (Section 5.5), which is a necessary step in using the program for any specific application.

5.1 File-Specification File

The file-specification file contains the names of input, output, and plot files for program *FEMFLOW3D*. For reference, each of these files has been assign a name, which is shown in parentheses in the text. Two input files are associated with subroutine *MODEL*; they specify the ground-water problem to be solved and the parameter indexes to be used in parameter identification. Five files provide output data for each simulation (MODEL.OUT, MODEL.PLT, MODEL.BUD, MODEL.FLX, AND MODEL.RIV). Fifteen input files are associated with subroutine *WATER*; they specify the source and use of irrigation water (WELL.DAT, WSTAT.DAT, PUMP.DAT, CPUMP.DAT, USER.DAT, DELIVER.DAT, CDELIVER.DAT, CROP.DAT, ROOT.DAT, PRECIP.DAT, PET.DAT, CCROP.DAT, HARDPAN.DAT, DESTIN.DAT, and EXCLUDE.DAT). Fifteen files provide output data related to subroutine *WATER* (WATER.OUT, WELL.OUT, PUMP.OUT, CPUMP.OUT, USER.OUT, DELIVER.OUT, CDELIVER.OUT, CROP.OUT, ROOT.OUT, PRECIP.OUT, PET.OUT, CCROP.OUT, HARDPAN.OUT, DESTIN.OUT, AND EXCLUDE.OUT). In addition to the parameter identification file (MODEL.IND), the file-specification file also contains the names of the other files needed for subroutine *SEARCH*, the input file SEARCH.DAT and the output file SEARCH.OUT.

FEMFLOW3D has a file-handing facility that allows each input file to be specified as a single file or as a series of file segments that are concatenated into a single input file. File names are specified for each input file and associated file segments. Subroutine *MERGE* performs the concatenation by copying the file segments into the input file.

The file names for a *FEMFLOW3D* simulation are listed sequentially in the file-specification file (FILES.DAT). *FEMFLOW3D* asks for the name of this file-specification file on the CRT screen prompt, and the file name is entered using the keyboard. *FEMFLOW3D* then reads the names for the input and the output files and opens those files. The format for the file-specification file are listed in the next section. (See table 2 for a formatted example of these inputs.)

Table 2. Example of inputs for file-specification file (Records 1 through 57 for FILES.DAT)

Record	Input records	Remarks
Input files for subroutine MODEL		
	FILE1 A60	NFILE I10
1	MODEL.DAT	12
	XFILE A60	If NFILE is greater than zero, the file segments that make up the input file are listed. This example show 12 file segments for the MODEL input file.
2	STEP.DAT	
2	GRID.DAT	
2	AQUIFER.DAT	
2	HEAD.DAT	
2	CHEAD.DAT	
2	FLUX.DAT	
2	RIVER.DAT	
2	EVAP.DAT	
2	VFLUX.DAT	
2	FAULT.DAT	
2	SINK.DAT	
2	WATER.DAT	
Output files for subroutine MODEL		
	FILE20 A60	
3	RIVER.OUT	
	FILE2 A60	
4	MODEL.OUT	
	FILE3 A60	
5	MODEL.PLT	
	FILE4 A60	
6	MODEL.BUD	
	FILE6 A60	
7	MODEL.FLX	

Table 2. Example of inputs for file-specification file (Records 1 through 57 for FILES.DAT)--Continued

Record	Input records		Remarks
Input file for parameter-index file.			
	FILE5 A60	NFILE I10	
8	MODEL.IND	0	IF NFILE is zero, no file segments are listed and the record for XFILE is omitted.
Input files for subroutine WATER1			
	FILE2 A60	NFILE I10	
10	WELL.DAT	0	
	FILE29 A60	NFILE I10	
12	WSTAT.DAT	0	
	FILE3 A60	NFILE I10	
14	PUMP.DAT	0	
	FILE4 A60	NFILE I10	
16	CPUMP.DAT	0	
	FILE5 A60	NFILE I10	
18	USER.DAT	0	
	FILE6 A60	NFILE I10	
20	DELIVER.DAT	0	
	FILE7 A60	NFILE I10	
22	CDELIVER.DAT	0	
	FILE8 A60	NFILE I10	
24	CROP.DAT	0	

Table 2. Example of inputs for file-specification file (Records 1 through 57 for FILES.DAT)--Continued

Record	Input records	Remarks
Input files for subroutine WATER1--Continued		
	FILE9 A60	NFILE I10
26	ROOT.DAT	0
	FILE10 A60	NFILE I10
28	PRECIP.DAT	0
	FILE11 A60	NFILE I10
30	PET.DAT	0
	FILE12 A60	NFILE I10
32	CCROP.DAT	0
	FILE25 A60	NFILE I10
34	HARDPAN.DAT	0
	FILE27 A60	NFILE I10
36	DESTIN.DAT	0
	FILE30 A60	NFILE I10
38	EXCLUDE.DAT	0
Output files for subroutine WATER1		
	FILE13 A60	
40	WATER.OUT	
	FILE14 A60	
41	WELL.OUT	

Table 2. Example of inputs for file-specification file (Records 1 through 57 for FILES.DAT)--Continued

Record	Input records	Remarks
Output files for subroutine WATER1--Continued		
	FILE15 A60	
42	PUMP.OUT	
	FILE16 A60	
43	CPUMP.OUT	
	FILE17 A60	
44	USER.OUT	
	FILE18 A60	
45	DELIVER.OUT	
	FILE19 A60	
46	CDELIVER.OUT	
	FILE20 A60	
47	CROP.OUT	
	FILE21 A60	
48	ROOT.OUT	
	FILE22 A60	
49	PRECIP.OUT	
	FILE23 A60	
50	PET.OUT	
	FILE24 A60	
51	CCROP.OUT	

Table 2. Example of inputs for file-specification file (Records 1 through 57 for FILES.DAT)--Continued

Record	Input records		Remarks
Output files for subroutine WATER1--Continued			
	FILE26 A60		
52	HARDPAN.OUT		
	FILE28 A60		
53	DESTIN.OUT		
	FILE31 A60		
54	EXCLUDE.OUT		
Input file for parameter identification			
	FILE1 A60	NFILE I10	
55	SEARCH.DAT	2	This example shows two file segments, SEARCH.PAR and LEVELS.DAT, which will be concatenated and copied into the file SEARCH.DAT.
	XFILE A60		
56	SEARCH.PAR		
56	LEVELS.DAT		
Output file for parameter identification			
	FILE2 A60		
57	SEARCH.OUT		

5.1.1 Files for Subroutine *MODEL*

Subroutine *MODEL* opens six files. These include two input files and four output files as follows:

Record	Columns	Format	Variable	Variable Description
1	1-60	A60	FILE1	Name of model-input file (MODEL.DAT).
	61-70	I10	NFILE	Number of file segments that make up the model-input file.
2	1-60	A60	XFILE(*i*)	Name of segment for model-input file.
3	1-60	A60	FILE20	Name of river-output file (MODEL.RIV).
4	1-60	A60	FILE2	Name of main output file (MODEL.OUT).
5	1-60	A60	FILE3	Name of model-plot output file (MODEL.PLT).
6	1-60	A60	FILE4	Name of water-budget output file (MODEL.BUD).
7	1-60	A60	FILE6	Name of flux-output file (MODEL.FLX).
8	1-60	A60	FILE5	Name of parameter-index input file (MODEL.IND).
	61-70	I10	NFILE	Number of file segments that make up the parameter-index file.
9	1-60	A60	XFILE(*i*)	Name of segment for parameter-index file.

Notes:

1. If NFILE is zero, file segments (XFILE) are omitted. If NFILE is greater than zero, the record for reading a file-segment name is repeated NFILE times. For example, for Records 1 and 2, Record 2 is repeated NFILE times as specified in Record 1. Accordingly, the NFILE files named in XFILE(*i*) are concatenated and copied into the file named in FILE1, and that file is opened.

2. If any of the model output files are not needed, it can be named "NUL" and no output file will be written.

3. The parameter index file (FILE5) can be specified as "NUL" if the *SEARCH* subroutine is not used and Record 9 is omitted.

4. The file names in parentheses in the "variable description" column indicate suggested file names only and may be named differently for each application.

5. Input files should be "write protected" because errors in constructing the file-specification file can result in writing to input files.

5.1.2 Files for Subroutine WATER

Subroutine *WATER* opens 30 files and their corresponding file segments. These files are listed below. The first 15 files are the input files, and the remaining 15 files are the output files. If subroutine *WATER* is not used, these files are omitted.

Record	Columns	Format	Variable	Variable Description
10	1-60	A60	FILE2	Name of well-site inventory file (WELL.DAT).
	61-70	I10	NFILE	Number of file segments that makes up the well-site inventory file.
11	1-60	A60	XFILE(*i*)	Name of segment for well-site inventory file.
12	1-60	A60	FILE29	Name of well-status file (WSTAT.DAT).
	61-70	I10	NFILE	Number of file segments that makes up the well-status file.
13	1-60	A60	XFILE(*i*)	Name of segment for well-status file.
14	1-60	A60	FILE3	Name of well-pumping file (PUMP.DAT).
	61-70	I10	NFILE	Number of file segments that makes up the well-pumping file.
15	1-60	A60	XFILE(*i*)	Name of segment for well-pumping file.
16	1-60	A60	FILE4	Name of pumping-construction file (CPUMP.DAT).
	61-70	I10	NFILE	Number of file segments that makes up the pumping-construction file.
17	1-60	A60	XFILE(*i*)	Name of segment for well-pumping file.
18	1-60	A60	FILE5	Name of water-user inventory file (USER.DAT).
	61-70	I10	NFILE	Number of file segments that makes up the water-user inventory file.
19	1-60	A60	XFILE(*i*)	Name of segment for water-user inventory file.
20	1-60	A60	FILE6	Name of water-delivery file (DELIVER.DAT).
	61-70	I10	NFILE	Number of file segments that makes up the water-delivery file.
21	1-60	A60	XFILE(*i*)	Name of segment for water-delivery file.
22	1-60	A60	FILE7	Name of delivery-construction file (CDELIVER.DAT).
	61-70	I10	NFILE	Number of segments that makes up the delivery-construction file.
23	1-60	A60	XFILE(*i*)	Name of segments for delivery-construction file.
24	1-60	A60	FILE8	Name of crop-inventory file (CROP.DAT).
	61-70	I10	NFILE	Number of segments that makes up the crop-inventory file.
25	1-60	A60	XFILE(*i*)	Name of segment for crop-inventory file.
26	1-60	A60	FILE9	Name of rooting-depth file (ROOT.DAT).
	61-70	I10	NFILE	Number of segments that makes up the rooting-depth file.
27	1-60	A60	XFILE(*i*)	Name of segment for rooting-depth file.
28	1-60	A60	FILE10	Name of precipitation file (PRECIP.DAT).
	61-70	I10	NFILE	Number of segments that makes up the precipitation file.
29	1-60	A60	XFILE(*i*)	Name of segment for precipitation file.

Record	Columns	Format	Variable	Variable Description
30	1-60	A60	FILE11	Name of potential evapotranspiration file (PET.DAT).
	61-70	I10	NFILE	Number of segments that makes up the potential evapotranspiration file.
31	1-60	A60	XFILE(*i*)	Name of segment for potential evapotranspiration file.
32	1-60	A60	FILE12	Name of crop-construction file (CCROP.DAT).
	61-70	I10	NFILE	Number of segments that makes up the crop-construction file.
33	1-60	A60	XFILE(*i*)	Name of segment for crop-construction file.
34	1-60	A60	FILE25	Name of recharge-fraction file (HARDPAN.DAT).
	61-70	I10	NFILE	Number of segments that makes up the recharge-fraction file.
35	1-60	A60	XFILE(*i*)	Name of segment for recharge-fraction file.
36	1-60	A60	FILE27	Name of pumping-destination file (DESTIN.DAT).
	61-70	I10	NFILE	Number of segments that makes up the pumping-destination file.
37	1-60	A60	XFILE(*i*)	Name of segment for the pumping-destination file.
38	1-60	A60	FILE30	Name of well-exclusion file (EXCLUDE.DAT).
	61-70	I10	NFILE	Number of file segments that makes up the well-exclusion file.
39	1-60	A60	XFILE(*i*)	Name of segment for well-exclusion file.
40	1-60	A60	FILE13	Name of water-output file (WATER.OUT).
41	1-60	A60	FILE14	Name of well-site inventory output file (WELL.OUT).
42	1-60	A60	FILE15	Name of well-pumping output file (PUMP.OUT).
43	1-60	A60	FILE16	Name of pumping-construction output file (CPUMP.OUT).
44	1-60	A60	FILE17	Name of water-user inventory output file (USER.OUT).
45	1-60	A60	FILE18	Name of water-delivery output file (DELIVER.OUT).
46	1-60	A60	FILE19	Name of delivery-construction output file (CDELIVER.OUT).
47	1-60	A60	FILE20	Name of crop-inventory output file (CROP.OUT).
48	1-60	A60	FILE21	Name of rooting-depth output file (ROOT.OUT).
49	1-60	A60	FILE22	Name of precipitation output file (PRECIP.OUT).
50	1-60	A60	FILE23	Name of potential evapotranspiration output file (PET.OUT).
51	1-60	A60	FILE24	Name of crop-construction output file (CCROP.OUT).
52	1-60	A60	FILE26	Name of recharge-factor output file (HARDPAN.OUT).
53	1-60	A60	FILE28	Name of pumping-destination output file (DESTIN.OUT).
54	1-60	A60	FILE31	Name of well-exclusion output file (EXCLUDE.OUT)

Notes:

1. If NFILE is zero, file segments (XFILE) are omitted. IF NFILE is greater than zero, the record for reading a file-segment name is repeated NFILE times. For example, for Records 1 and 2, Record 2 is repeated NFILE times as specified in Record 1. Accordingly, the NFILE files named in XFILE(*i*) are concatenated and copied into the file named in FILE1, and that file is opened.

2. If subroutine *WATER* is not being used in the simulation, Records 10 through 54 can be omitted.

3. The file names in parentheses in the "variable description" column indicate suggested file names only and may be named differently for each application.

4. Input files should be "write protected" because errors in constructing the file-specification file can result in writing to input files.

5.1.3 Files for Subroutine *SEARCH*

Subroutine *SEARCH* opens two files, one input file and one output file. These files are listed below. If subroutine *SEARCH* is not used, these files can be omitted and not listed in the file-specification file.

Record	Columns	Format	Variable	Variable Description
55	1-60	A60	FILE1	Name of parameter-identification input file (SEARCH.DAT).
	61-70	I10	NFILE	Number of file segments that makes up the parameter-identification file.
56	1-60	A60	XFILE(*i*)	Name of segment for parameter-identification file.
57	1-60	A60	FILE2	Name of parameter-identification output file (SEARCH.OUT).

Notes:

1. If NFILE is zero, file segments (XFILE) are omitted. IF NFILE is greater than zero, the record for reading a file-segment name is repeated NFILE times. For example, for Records 1 and 2, Record 2 is repeated NFILE times as specified in Record 1. Accordingly, the NFILE files named in XFILE(*i*) are concatenated and copied into the file named in FILE1, and that file is opened.

2. If parameter identification is not done for a simulation, then FILE5 in Record 8 is named "NUL," NFILE equals zero, and Record 9 is omitted.

3. The file names in parentheses in the "variable description" column indicate suggested file names only and may be named differently for each application.

4. Input files should be "write protected" because errors in constructing the file-specification file can result in writing to input files.

The example of the files-specification file given in table 2 represents a model simulation that includes an irrigated agricultural system represented by the input files for subroutine *WATER*. Additionally, the example represents a model simulation in which model parameters are identified to fit calculated ground-water levels to measured water levels, as represented by the input files for subroutine *SEARCH*. In the example, only the input files for subroutine *MODEL* and subroutine *SEARCH* are represented by file segments that will be concatenated into the files MODEL.DAT and SEARCH.DAT. respectively. However, any of the other input files could be concatenated from model segments.

If a model simulation does not invoke the use of subroutine *WATER* or the use of subroutine *SEARCH* to perform parameter identification, some of the files in table 2 are not applicable, and the names of the inapplicable files can be deleted from the file-specification file. If subroutine *WATER* is not used, the input and the output files for subroutine *WATER* must be deleted. If parameter identification is not going to be done, the input and the output files for parameter identification must also be deleted. However, there must be a file name for the parameter-index file, with NUL used for the dummy-file name.

The input and the output files listed in the file-specification file may reside in any volume or directory of the computer. If a file is in a directory other than the directory from which *FEMFLOW3D* is executed, the path name for the file must be included.

5.2 Subroutine *MODEL* Input File

The model-input file (MODEL.DAT) contains specifications for the simulation parameters, time-step scheme, output frequency controls, matrix solution options, the finite-element grid, aquifer parameters, initial conditions, constant-head boundary conditions, internal sources and sinks, stream-aquifer parameters, ground-water evapotranspiration, variable-flux boundary conditions, fault conditions, subsidence parameters, and controls for the use of the subroutine *WATER.*

5.2.1 Simulation Parameters (*MODEL*)

The simulation parameters represent switches that select basic options in *FEMFLOW3D* for use in a simulation. These inputs are listed below. (See table 3 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1-78	A78	TITLE	Title of simulation.
2	1-10	I10	MAXKNS	Maximum number of time steps.
	11-20	I10	IDELT	Switch for type of time-step scheme.
	21-30	I10	NITER	Maximum number of iterations.
	31-40	F10.0	CLOSE	Closure criterion for iterations [ft].
	41-50	I10	ISKIP	Switch for skipping updates when a problem is linear.
	51-60	I10	IFIT	Switch for doing parameter identification.
	61-70	I10	ISS	Switch for calculating steady-state heads in the first time step.
	71-80	I10	ISOLVE	Switch for selecting solution method.

Table 3. Example of inputs for simulation parameters (Records 1 and 2 for MODEL.DAT)

Record	Input records							
	TITLE A78							
1	FEMFLOW3D - Western Basin - Run 1							
	MAXKNS I10	IDELT I10	NITER I10	CLOSE F10.0	ISKIP I10	IFIT I10	ISS I10	ISOLVE I10
2	9	2	10	0.01	0	1	1	1

Notes:

1. One of two time-step schemes can be selected by setting the switch IDELT. If IDELT equals 1, a scheme with geometrically expanding time steps is selected. If IDELT equals 2, a scheme with integer-multiple time steps is selected. If subroutine *WATER* is used in the simulation, IDELT must equal 2.

2. The maximum number of iterations NITER should equal 1 if (1) the finite-element grid is fixed over time, (2) the "drainage-node" form of the specified-head boundary condition is not used, (3) stream-aquifer interactions are not simulated, (4) evapotranspiration is not simulated, and (5) land subsidence is not simulated. Otherwise, NITER should equal about 10.

3. The closure criterion CLOSE is the maximum absolute difference in calculated hydraulic heads between two iterations (see discussion in Section 3.2.2.5). CLOSE should equal about 0.05 to 0.005 ft.

4. The switch ISKIP must equal zero (no skipping) if the simulation is nonlinear or has unequal time steps. ISKIP can equal one (skipping) if the simulation is linear and has equal time steps, as discussed in Section 3.2.3.3

5. The switch IFIT is for selecting parameter identification. If IFIT equals 1, parameter identification is performed; if IFIT equals zero, it is not performed.

6. The switch ISS is for selecting a steady state for the first time step. If ISS equals 1, then the calculated heads for the first time step are the steady-state heads, and the transient-state simulation starts with the second time step. In this case, the specified initial conditions are not used. If ISS equals zero, then the transient-state simulation starts with the first time step. In this case, the specified initial conditions are used.

7. If ISOLVE equals 1, the direct solution method is used and no other input is needed. If ISOLVE equals 2, the iterative solution method is used and Record 8 (Section 5.2.4) must be included.

5.2.2 Time-Step Scheme (*MODEL*)

The inputs for the time-step scheme describe the discretization of time within the simulation. One of two types of time-step schemes can be selected. These inputs are listed below. (See table 4 for formatted examples of these inputs.)

Record	Columns	Format	Variable	Variable Description
3	1-10	I10	NSTART	Number of time-step periods.
4	1-10	I10	KNS1	First time step of period.
	11-20	I10	KNS2	Last time step of period.
	21-30	F10.0	DELT0	Initial time step for period [days].
	31-40	F10.0	FDELT	Factor for expansion of time steps during period [dimensionless].
5	1-10	F10.0	DELT0	Basic time step [days].
6	1-10	I10		Time step (for reference only).
	11-20	I10	ICMO1(*i*)	Start of time step in integer multiples of DELT0.
	21-30	I10	ICMO2(*i*)	End of time step in integer multiples of DELT0.

Table 4. Examples of inputs for alternative time-step schemes (Records 3 through 6 for MODEL.DAT)

Record	Input records				Remarks
Example for IDELT = 1 and MAXKNS = 50:					
	NSTART I10				
3	9				
	KNS1 I10	KNS2 I10	DELTO F10.0	FDELT F10.0	
4	1	10	0.05	1.2	Expanding time steps.
4	11	50	0.31	1.0	Constant time steps.
Example for IDELT = 2, ISS= 0, and MAXKNS = 8:					Steady-state conditions are not calculated.
	DELTO F10.0				
5	30.4				
	time step I10	ICMO1 I10	ICMO2 I10		
6	1	1	3		Start transient time steps.
6	2	4	6		
6	3	7	9		
6	4	10	12		
6	5	13	15		
6	6	15	18		
6	7	19	21		
6	8	22	24		
Example for IDELT = 2 ISS = 1, and MAXKNS = 9:					Steady-state initial condition.
	DELTO F10.0				
5	30.4				
	time step I10	ICMO1 I10	ICMO2 I10		
6	1	1	12		Start steady-state time step.
6	2	1	3		Proceed to transient time steps.
6	3	4	6		
6	4	7	9		
6	5	10	12		
6	6	13	15		
6	7	16	18		
6	8	19	21		
6	9	22	24		

Notes:

1. If IDELT in Record 2 equals 1, omit Records 5 and 6. If IDELT equals 2, omit Records 3 and 4.

2. If IDELT equals 1, repeat Record 4 NSTART times.

3. If IDELT equals 2 and subroutine *WATER* is used in the simulation, DELTO must equal 1/12 year in days (30.42 days).

4. If IDELT equals 2, repeat Record 6 MAXKNS times, where MAXKNS appears in Record 2. Time steps are integer multiples of DELTO and are based on the specifications of ICMO1(*i*) and ICMO2(*i*).

Examples of three alternative time-step schemes are given in table 4. If subroutine *WATER* is not used, either the geometrically expanding time-step scheme, where IDELT = 1, or the integer-multiple time-step scheme, where IDELT = 3, may be used. If subroutine *WATER* is used, the integer-multiple time-step scheme, where IDELT = 2, must be used.

The first example is for IDELT = 1 and MAXKNS = 50. For this example, two time-step periods were used (NSTART = 2). For the first period, there were 10 time steps (KNS1 = 1 and KNS2 = 10), where the initial time step was 5 days (DELTO = 0.05), and the time-step length was increased by a factor of 1.2 for each following step (FDELT = 1.2). For the second period, there were 40 time steps (KNS1 = 11 and KNS2 = 50), where the initial time step for the period was 31 days (DELTO = 0.31), and the time-step length was constant for each following time step (FDELT = 1.0).

The second example is for IDELT = 2, ISS = 0, and MAXKNS = 8. The basic time step for this example was 30.4 days, (DELTO = 30.4, which is the standard time step used in subroutine *WATER* for the input of water-use and other data. However, the time steps used in the ground-water model are integer multiples of the basic time step. For the second example, each of the eight time steps were three basic time steps in length. The first time step included basic time steps 1 through 3 (ICMO1 = 4 and ICMO2 = 6). In the second example, each time step was equal in length; however, by selecting different values of ICMO1 and ICMO2, each step can vary in length, as needed.

The third example is for IDELT = 2, ISS = 1, and MAXKNS = 9. For this example, the time-step schemes for steps 2 through 9 were the same time-step schemes used for steps 1 through 8 for the second example. For the third example, however, time step 1 represented the steady-state condition, and the initial steady-state conditions were calculated using pumpage and ground-water recharge for basic steps 1 through 12 (ICMO1 = 1 and ICMO2 = 12).

The interactions between subroutines *WATER* and *MODEL* are controlled by the time-step schemes. Subroutine *WATER* assumes that the basic time-step is 1 month (DELTO = 30.4 days), and all data inputs are specified for particular years and months. Within the time-step loop of subroutine *MODEL*, subroutine *WATER* is called for each time step, which is an integer multiple of the basic time step of 1 month. Subroutine *WATER* calculates average pumpage and ground-water recharge for the time step on the basis of a 1-month time step within subroutine *WATER*. Consequently, when subroutine *WATER* is used, *FEMFLOW3D* operates on two time-step schemes: the 1-month time step in subroutine *WATER* and the integer-multiple time step in subroutine *MODEL*.

Subroutine *WATER* operates on an 1-month time step. The other subroutines operate on the time step specified in the time-step scheme input, whether IDELT = 1 or IDELT = 2, and include subroutines for specified-head boundaries, stream-aquifer interactions, ground-water evapotranspiration, variable-flux boundaries, fault effects, and land subsidence. All data inputs for these subroutines must correspond to the time steps in subroutine *MODEL*.

5.2.3 Output Controls (*MODEL*)

The inputs for output controls determine the frequency of time-step outputs. These inputs are listed below. (See table 5 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
7	1-10	I10	NPRINT	Frequency of output to the river model output file, the model output file, and the water-budget file.
	11-20	I10	NPLOT	Frequency of output to model plot file.
	21-30	I10	NFLUX	Frequency of output to model flux file.

Table 5. Example of inputs for output frequency controls (Record 7 for MODEL.DAT)

Record	Input records		
	NPRINT	NPLOT	NFLUX
	I10	I10	I10
7	10	1	5

Notes:

1. NPRINT specifies the time-step frequency that the calculated heads and other information are written to the model output file.

2. NPLOT specifies the time-step frequency that the calculated heads are written to the model plot file.

3. NFLUX specifies the time-step frequency that the inflow to and outflow from each node are written to the model flux file.

5.2.4 Matrix Solution Parameters *(SOLVE)*

The matrix solution parameters define the maximum number of iterations, the relaxation factor, and the closure criterion for the iterative solution method. These inputs are listed below. (See table 6 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
8	1-10	I10	NITER	Maximum number of iterations.
	11-20	F10.0	OMEGA	Relaxation factor.
	21-30	F10.0	CLOSE	Closure criterion [ft].

Notes:

1. If the switch ISOLVE in Record 2 equals 1, Record 8 is omitted and the direct solution method is used.
2. The run time for *FEMFLOW3D* is minimized when OMEGA, the relaxation factor, is assigned a value between 1.2 and 1.7.

Table 6. Example of inputs for the matrix solution parameters (Record 8 for MODEL.DAT)

Record	Input records			Remarks
	NITER I10	OMEGA F10.0	CLOSE F10.0	
8	500	1.7	0.001	If the switch ISOLVE in Record 2 equals 1, Record 8 is omitted and the direct solution method is used.

5.2.5 Finite-Element Grid Coordinates (*MODEL*)

The inputs of the finite-element coordinate define the three-dimensional position of the nodes within the grid. These inputs are listed below. (See table 7 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
9	1-10	I10	NN	Number of nodes.
	11-20	I10	NE	Number of prismatic elements.
10	1-10	F10.0	FACX	Multiplication factor for *x* coordinates [dimensionless]
	11-20	F10.0	FACY	Multiplication factor for *y* coordinates [dimensionless].
	21-30	F10.0	FACZ	Multiplication factor for *z* coordinates [dimensionless].
	31-40	I10	IECHO1	Switch for echo of node coordinates.
11	1-10	I10		Node number (for reference only).
	11-20	F10.0	X(*i*)	*X* coordinate of node [ft].
	21-30	F10.0	Y(*i*)	*Y* coordinate of node [ft].
	31-40	F10.0	Z(*i*)	*Z* coordinate of node [ft].

Table 7. Example of inputs for finite-element grid coordinates (Records 9 through 11 for MODEL.DAT)

Record	Input records			
	NN I10	NE I10		
9	698	946		
	FACX F10.0	FACY F10.0	FACZ F10.0	IECHO1 I10
10	1.0	1.0	1.0	0
	node number	X F10.0	Y F10.0	Z F10.0
11	1	1212153.	427015.	0
11	2	1212153.	427015.	-45
11	3	1212718.	428532.	0
11	4	1212718.	428532.	-50
11	5	1213313.	430068.	0
11	6	1213313.	430068.	-50
11	7	1213843.	431569.	0
11	8	1213843.	431569.	-50
11	9	1214298.	432775.	0
11	10	1214298.	432775.	-45
11	11	1217064.	441786.	0
11	12	1217064.	441786.	-60
11	13	1217064.	441786.	-104
11	14	1217064.	441786.	-180
11	15	1217551.	443443.	0
11	16	1217551.	443443.	-60
11	17	1217551.	443443.	-104
11	18	1217551.	443443.	-180
11	19	1217732.	444098.	0
11	20	1217732.	444098.	-60
11	21	1217732.	444098.	-104
11	22	1217732.	444098.	-125
11	23	1217886.	444598.	0
11	24	1217886.	444598.	-60
11	25	1217886.	444598.	-104
11	26	1217886.	444598.	-125

Notes:

1. If the switch IECHO1 equals 1, displays occur. If the switch equals zero, no display occurs.
2. Record 11 is repeated for NN nodes.

5.2.6 Finite-Element Grid Incidences (*MODEL*)

The inputs of the finite-element grid incidences describe the definition of elements with regard to nodes. These inputs are listed below. (See table 8 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
12	1-10	I10	IECHO2	Switch for display of incidences for prismatic elements.
	11-20	I10	IECHO3	Switch for display of incidences for tetrahedral elements.
13	1-10	I10		Element number (for reference only).
	11-70	6I10	IPRISM(*i*)	Node incidences for a prismatic element.

Table 8. Example of inputs for the finite-element grid incidences (Records 12 and 13 for MODEL.DAT)

Record Input records

	IECHO2	IECHO3
	I10	I10
12	1	1

Regular prismatic elements have six node incidences:

	element number	IPRISM	IPRISM	IPRISM	IPRISM	IPRISM	IPRISM
		I10	I10	I10	I10	I10	I10
13	1	11	45	15	12	46	16
13	2	15	45	49	16	46	50
13	3	49	19	15	50	20	16
13	4	49	53	19	50	54	20
13	5	19	53	23	20	54	24
13	6	53	57	23	54	58	24
13	7	23	57	27	24	58	28
13	8	27	57	61	28	58	62
13	9	41	75	45	42	76	46
13	10	75	49	45	76	50	46
13	11	75	77	49	76	78	50

Elements can also be irregular to represent geologic features that pinch out. For example, elements with one zero-height side have five node incidences:

	element number	IPRISM	IPRISM	IPRISM	IPRISM	IPRISM	IPRISM
		I10	I10	I10	I10	I10	I10
15	203	645	641	667	646	642	0

Element with two zero-height sides have four node incidences:

	element number	IPRISM	IPRISM	IPRISM	IPRISM	IPRISM	IPRISM
		I10	I10	I10	I10	I10	I10
13	204	641	638	667	642	0	0

Notes:

1. Record 13 identifies the six nodes that define a prismatic element. The nodes are numbered counterclockwise around the top and then counterclockwise around the bottom.

2. Zero-height edges are identified by a node number of zero. For one zero-height edge, the zero-node number must be listed as the sixth node incidence in Record 13. For two zero-height edges, the zero-node number must be listed as the fifth and sixth node incidences in Record 13.

3. Record 13 is repeated for each element specified by NE.

5.2.7 Aquifer Parameters (*MODEL*)

The inputs for aquifer parameters describe the hydraulic properties of the ground-water system. These inputs are listed below. (See table 9 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
14	1-10	F10.0	FACKXX	Multiplication factor for hydraulic conductivity in the *x* direction [dimensionless].
	11-20	F10.0	FACKYY	Multiplication factor for hydraulic conductivity in the *y* direction [dimensionless].
	21-30	F10.0	FACKZZ	Multiplication factor for hydraulic conductivity in the *z* direction [dimensionless].
	31-40	F10.0	FACSS	Multiplication factor for specific storage [dimensionless].
	41-50	F10.0	FACSY	Multiplication factor for specific yield [dimensionless].
	51-60	I10	IECHO4	Switch for display of aquifer parameters.
	61-70	I10	IECHO5	Switch for display of parameter indexes.
15	1-10	I10		Element number (for reference only).
	11-20	F10.0	KXX0	Initial value of hydraulic conductivity in *x* direction [ft/d].
	21-30	F10.0	KYY0(*i*)	Initial value of hydraulic conductivity in *y* direction [ft/d].
	31-40	F10.0	KZZ0(*i*)	Initial value of hydraulic conductivity in *z* direction [ft/d].
	41-50	F10.0	SS0(*i*)	Initial specific storage [1/ft].
	51-60	F10.0	SY0(*i*)	Initial specific yield [dimensionless].
	61-70	I10	ITOP	Flag to identify prismatic elements that make up the top of grid.

Table 9. Example of inputs for aquifer parameters (Records 14 and 15 for MODEL.DAT)

Record	Input records						
	FACKXX F10.0	FACKYY F10.0	FACKZZ F10.0	FACSS F10.0	FACSY F10.0	IECHO4 I10	IECHO5 I10
14	1.0	1.0	1.0	1.0	1.0	1	1
	element number	KXXO F10.0	KYYO F10.0	KZZO F10.0	SSO F10.0	SYO F10.0	ITOP I10
15	1	4.35E+01	4.35E+01	1.20E-02	4.11E-04	7.14E-02	1
15	2	2.86E+01	2.86E+01	2.10E-01	4.34E-05	0.00E+00	0
15	3	6.19E+01	6.19E+01	3.19E-01	4.34E-05	0.00E+00	0
15	4	4.29E+01	4.29E+01	1.19E-02	4.11E-04	7.08E-02	1
15	5	2.84E+01	2.84E+01	2.10E-01	4.34E-05	0.00E+00	0
15	6	6.18E+01	6.18E+01	3.19E-01	4.34E-05	0.00E+00	0
15	7	4.38E+01	4.38E+01	1.20E-02	4.11E-04	7.18E-02	1
15	8	2.80E+01	2.80E+01	2.09E-01	4.34E-05	0.00E+00	0
15	9	6.18E+01	6.18E+01	3.19E-01	4.34E-05	0.00E+00	0
15	10	4.15E+01	4.15E+01	1.18E-02	4.11E-04	6.94E-02	1
15	11	2.71E+01	2.71E+01	2.08E-01	4.34E-05	0.00E+00	0
15	12	6.13E+01	6.13E+01	3.18E-01	4.34E-05	0.00E+00	0
15	13	4.48E+01	4.48E+01	1.21E-02	4.11E-04	7.29E-02	1
15	14	2.73E+01	2.73E+01	2.08E-01	4.34E-05	0.00E+00	0
15	15	6.13E+01	6.13E+01	3.18E-01	4.34E-05	0.00E+00	0
15	16	4.38E+01	4.38E+01	1.20E-02	4.11E-04	7.18E-02	1
15	17	2.73E+01	2.73E+01	2.08E-01	4.34E-05	0.00E+00	0
15	18	6.12E+01	6.12E+01	3.18E-01	4.34E-05	0.00E+00	0

Notes:

1. Record 15 is repeated for each element specified by NE in Record 9.
2. If the switches IECHO4 or IECHO5 equal 1, then displays occur. If the switches equal zero, no displays occur.
3. The flag ITOP equals 1 for elements that form the water table; otherwise, ITOP equals zero. If a water-table condition is not being simulated, ITOP equals zero everywhere.
4. Values for initial specific yield SY0(*i*) are needed only for elements with ITOP equals 1.
5. Initial aquifer-parameter values are modified during the calibration process, if parameter identification is going to be done. Otherwise, the initial values are maintained throughout the simulation.

5.2.8 Initial Hydraulic Heads (*MODEL*)

The inputs for initial hydraulic head describe initial conditions for the simulation. These inputs are listed below. (See table 10 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
16	1-10	I10	IECHO6	Switch for display of initial hydraulic heads.
17	1-50	5F10.0	H0(*i*)	Initial hydraulic heads [ft].

Notes:

1. Record 17 is repeated for each node, specified by NN in Record 9, until NN values of H0(*i*) are input.

2. If the switch IECHO6 equals 1, a display occurs. If IECHO6 equals zero, no displays occur.

Table 10. Example of inputs for initial hydraulic heads (Records 16 and 17 for MODEL.DAT)

Record	Input records					Remarks
	IECHO6 I10					
16	1					
	H0 F10.0	H0 F10.0	H0 F10.0	H0 F10.0	H0 F10.0	(5F10.0)
						The file lists an initial head for each node in the model. If a calculation for steady-state heads is included in the simulation, initial heads may be set to zero or any arbitrary value.
17	5.0	5.0	5.0	5.0	5.0	
17	5.0	5.0	5.0	5.0	5.0	
17	5.0	5.0	5.0	5.0	5.0	
17	5.0	5.0	5.0	5.0	5.0	
17	5.0	5.0	5.0	5.0	5.0	
17	5.0	5.0	5.0	5.0	5.0	
17	32.4	32.2	31.1	31.0	29.9	
17	30.1	29.9	30.1	30.8	30.8	
17	13.7	13.7	13.6	13.6	12.4	
17	12.4	11.8	11.8	12.7	12.7	
17	12.0	12.0	13.2	13.1	12.0	
17	12.0	12.4	12.4	12.0	12.0	
17	12.1	12.1	11.9	11.9	39.0	
17	38.8	38.3	37.9	37.2	36.7	
17	36.6	36.2	36.6	36.7	15.4	
17	15.4	14.6	14.6	15.2	15.2	
17	14.6	14.6	15.2	15.2	14.3	
17	14.3	15.1	15.1	14.9	14.9	
17	15.0	15.0	43.4	41.8	42.4	
17	40.2	40.3	37.9	38.2	36.7	
17	36.1	34.4	34.0	25.1	22.7	
17	22.2	22.7	21.3	21.3	20.9	
17	20.9	21.5	21.5	20.8	20.8	
17	20.8	20.8	20.5	20.5	20.0	
17	20.0	19.8	19.8	36.3	35.2	
17	35.2	34.2	33.8	33.1	32.1	
17	32.0	32.3	28.1	24.6	24.3	
17	25.5	31.1	24.1	23.5	23.5	
17	23.6	26.2	26.2	24.1	24.1	
17	27.2	27.2	24.5	24.5	26.6	
17	26.6	24.6	24.6	32.5	31.7	
17	31.3	31.2	30.0	30.0	31.2	
17	29.8	29.7	29.6	31.1	36.1	

5.2.9 Specified-Head Boundary Condition (*CHEAD*)

The inputs for specified-head boundary conditions describe, in part, how the simulation relates to conditions outside the modeled volume. These inputs are listed below. (See table 11 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
18	1-10	I10	NCHN	Number of specified-head nodes.
	11-20	I10	IECHO1	Switch for display of inputs.
	21-30	I10	IECHO2	Switch for display of fluxes through specified-head nodes.
19	1-10	I10	CHNODE(i)	Identity of specified-head node.
	11-20	F10.0	CHEAD0(*i*)	Specified head [ft].
	21-30	F10.0	K(*i*)	Hydraulic conductivity of membrane [ft/d].
	31-40	F10.0	B(*i*)	Thickness of membrane [ft].
	41-50	F10.0	AREA(*i*)	Area of membrane that is associated with node [ft^2].
	51-60	I10	TYPE(i)	Switch for type of specified-head node.
	61-70	I10	TABLE(*i*)	Identity of table of specified heads.
20	1-10	I10	NTABLE	Number of tables of specified heads.
21	1-10	I10	NPT(*i*)	Number of entries in a table.
22	1-10	F10.0	TTAB(*i,j*)	Time for entry in table [days].
	11-20	F10.0	CHTAB(*i,j*)	Specified head for entry in table [ft].

Table 11. Example of inputs for specified-head boundary conditions (Records 18 through 22 for MODEL.DAT)

Record	Input records							Remarks
	NCHN	IECHO1	IECHO2					
	I10	I10	I10					
18	15	1	1					
	CHNODE	CHEAD0	K	B	AREA	TYPE	TABLE	
	I10	F10.0	F10.0	F10.0	F10.0	I10	I10	
19	321	130.	0.001	100.	3.0E+07	0		Lake
19	325	130.	0.001	100.	3.0E+07	0		Lake
19	393	130.	0.001	100.	3.0E+07	0		Lake
19	397	130.	0.001	100.	3.0E+07	0		Lake
19	65	154.	40.	20.	1.5E+06	1	1	River
19	93	152.	40.	20.	1.5E+06	1	1	River
19	125	149.	40.	20.	1.5E+06	1	1	River
19	161	145.	40.	20.	1.5E+06	1	1	River
19	193	136.	40.	20.	1.5E+06	1	1	River
19	229	129.	40.	20.	1.5E+06	1	1	River
19	261	125.	40.	20.	1.5E+06	1	1	River
19	237	130.	10.	2640.	1.0E+05	2	2	Subsurface drain
19	241	131.	10.	2640.	1.0E+05	2	2	Subsurface drain
19	269	126.	10.	2640.	1.0E+05	2	2	Subsurface drain
19	273	127.	10.	2640.	1.0E+05	2	2	Subsurface drain
	NTABLE							
	I10							
20	2							
	NPT							
	I10							
21	9							
	TTAB	CHTAB						
	F10.0	F10.0						
22	0.00	3.03						Changes in river stage
22	365.25	3.15						
22	730.50	0.55						
22	1095.75	1.59						
22	1461.00	3.68						
22	1826.25	1.65						
22	2191.50	0.12						
22	2556.75	-0.90						
22	2922.00	2.19						
	NPT							
	I10							
21	9							
	TTAB	CHTAB						
	F10.0	F10.0						
22	0.00	-5.00						Depth of drain below land surface
22	365.25	-5.00						
22	730.50	-5.00						
22	1095.75	-5.00						
22	1461.00	-5.00						
22	1826.25	-5.00						
22	2191.50	-5.00						
22	2556.75	-5.00						
22	2922.00	-5.00						

Notes:

1. If NCHN equals zero, then Records 19 through 22 are omitted.

2. If the switches IECH01 and IECH02 equal 1, displays occur. If the switches equal zero, then no displays occur.

3. If the switch TYPE(*i*) equals zero, unlimited discharge can occur in either direction through the specified-head nodes. This boundary condition represents constantly flowing rivers with a hydraulic connection to the water table. If the switch equals 1, unlimited discharge can occur from the model domain through the specified-head nodes, and only limited discharge can occur into the model domain, where the maximum inward discharge is discharge that occurs with a unit hydraulic gradient through the membrane. This second boundary condition represents rivers or lakes with a limited capacity for recharging the ground-water system. If the switch equals 2, unlimited discharge can occur from the model domain through the specified-head nodes, but no discharge can occur into the model domain. This third boundary condition is used to represent drainage systems.

4. Record 19 is repeated for each specified-head node (NCHN times).

5. If NTABLE equals zero, Records 21 and 22 are omitted.

6. Record 22 is repeated for NPT(*i*) table entries. Records 21 and 22 are repeated for NTABLE tables as a group.

5.2.10 Internal and Boundary Fluxes (*FLUX*)

The inputs for internal and boundary fluxes describe the flux sources and sinks within and on the boundaries of the modeled volume. These inputs are listed below. (See table 12 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
23	1-10	I10	NQSET	Number of data sets of node fluxes.
	11-20	I10	MAXKNS	Number of time steps.
	21-30	F10.0	FACQ	Multiplication factor for node fluxes [dimensionless].
	31-40	I10	IECHO1	Switch for display of input fluxes.
	41-50	I10	IECHO2	Switch for display of time-step fluxes.
24	1-10	I10	NQ	Number of node-flux values in data set.
25	1-10	I10	QNODE(*i*)	Identity of node in data set.
	11-20	F10.0	QSET(*i,j*)	Flux through node [ft^3/s].
26	1-10	I10		Time step (for reference only).
	11-60	5F10.0	FACSET(*i,j*)	Multiplication factor for data set and time step.

Table 12. Example of inputs for internal and boundary fluxes (Records 23 through 26 for MODEL.DAT)

Record	Input records						Remarks
	NQSET	MAXKNS	FACQ	IECHO1	IECHO2		
	I10	I10	F10.0	I10	I10		
23	2	9	1.0	1	1		
	NQ						
	I10						
24	2						
	QNODE	QSET					
	I10	F10.0					
25	483	1.32E+00					
25	565	1.32E+00					
	NQ						
	I10						
24	3						
	QNODE	QSET					
	I10	F10.0					
25	483	1.32E+00					
25	565	1.32E+00					
25	587	1.32E+00					
	time step	FACSET	FACSET	FACSET	FACSET	FACSET	
		F10.0	F10.0	F10.0	F10.0	F10.0	The format of 5F10.0 allows as many as five multiplication factors (FACSET) for the data sets (NQSET) to be listed per line. In this example, only two data sets are specified, and only two corresponding FACSETs are shown.
26	1	1.00	1.00				
26	2	1.00	1.00				
26	3	1.00	1.10				
26	4	1.00	1.20				
26	5	1.00	1.30				
26	6	1.00	1.40				
26	7	1.00	1.50				
26	8	1.00	1.60				
26	9	1.00	1.70				

Notes:

1. If NQSET equals zero, Records 24 through 26 are omitted.

2. If the switches IECHO1 and IECHO2 equal 1, displays occur. If the switches equal zero, no displays occur.

3. The value of MAXKNS in Record 23 must be greater than or equal to the value of MAXKNS in Record 2.

4. Record 25 is repeated for each node-flux value in the data set (NQ times). Records 24 and 25 are repeated for each data set (NQSET times) as a group.

5. Record 26 is repeated until NQSET values of FACSET(*i,j*) have been entered, and Record 26 is repeated for MAXKNS time steps.

5.2.11 River-Aquifer Interactions (*RIVER*)

The inputs for river-aquifer interactions determine the ground-water recharge and discharge that results from or produces riverflow within a network of river channels. Each river channel consists of a series of reaches associated with nodes in the model. These inputs are listed below. (See table 13 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
27	1-10	I10	NRCH	Number of river channel reaches.
	11-20	I10	NINF	Number of river inflow input locations.
	21-30	I10	NSTEP	Number of time steps.
	31-40	I10	IECHO1	Switch for display of river parameters.
	41-50	I10	IECH02	Switch for display of river inflows.
	51-60	I10	IECHO3	Switch for display of riverflow routing.
28	1-10	F10.0	AD	Depth coefficient $[(t/L^2)^{m_D}]$.
	11-20	F10.0	BD	Depth exponent [dimensionless].
	21-30	F10.0	AW	Width coefficient $[(t/L^2)^{m_w}]$.
	31-40	F10.0	BW	Width exponent [dimensionless].
29	1-10	I10	NRN(*i*)	Number of nodes for the reaches of the river channel.
	11-20	F10.0	KFACT	Multiplication factor for river-bed hydraulic conductivity [dimensionless].
30	1-10	I10	RNODE(*i,j*)	Identity of node for the reach of the river channel.
	11-20	F10.0	HBED(*i,j*)	Elevation of river bed at the node [ft].
	21-30	F10.0	LENGTH(*i,j*)	Length of reach of the river channel at the node [ft].
	31-40	F10.0	KBED(*i,j*)	Hydraulic conductivity of river bed at node [ft/d].
	41-50	F10.0	BBED(*i,j*)	Thickness of river bed at node [ft].
	51-60	I10	JOIN(*i,j*)	Identifier for river channel to be joined at this node.
	61-70	I10	SET(*i,j*)	Identifier for river inflow data set to be used at this river node.
	71-80	I10	ITABLE(*i,j*)	Identifier for table of the cross-sectional geometry of the reach to be used for this river node.
	81-90	I10	INDRIV(*i,j*)	Parameter identification index for assignment of hydraulic conductivity of the river bed.
31	1-10	I10	NTABLE	Number of reach-geometry tables.
32	1-10	I10	NPT(*i*)	Number of entries in reach-geometry table.
33	1-10	F10.0	QTAB(*i,j*)	Riverflow [ft^3/s].
	11-20	F10.0	WTAB(*i,j*)	Width of riverflow [ft].
	21-30	F10.0	DTAB(*i,j*)	Depth of flow at lowest point on cross section of the reach [ft].
34	1-10	F10.0	RFACT	Multiplication factor for riverflow inputs [dimensionless].
35	1-50	5F10.0	QIN(*i,j*)	River inflow input (ft^3/s).

Table 13. Example of inputs for river-aquifer interactions (Records 27 through 35 for MODEL.DAT)

Record	Input records									Remarks
	NRCH I10	NINF I10	NSTEP I10	IECHO1 I10	IECHO2 I10	IECHO3 I10				
27	3	5	10	1	1	1				
	AD F10.0	BD F10.0	AW F10.0	BW F10.0						
28	5.0	0.4	20.0	0.5						
	NRN I10	KFACT F10.0								
29	3	1.0								Channel 1
	RNODE I10	HBED F10.0	LENGTH F10.0	KBED F10.0	BBED F10.0	JOIN I10	SET I10	ITABLE I10	INDRIV I10	
30	10	100.0	4000.0	2.0	5.0	0	1	0	18	
30	12	90.0	4000.0	2.0	5.0	0	0	0	18	
30	14	80.0	4000.0	2.0	5.0	0	2	0	18	
	NRN I10	KFACT F10.0								
29	2	1.0								Channel 2
	RNODE I10	HBED F10.0	LENGTH F10.0	KBED F10.0	BBED F10.0	JOIN I10	SET I10	ITABLE I10	INDRIV I10	
30	30	110.0	2000.0	3.0	10.0	0	4	0	19	
30	28	100.0	2000.0	3.0	10.0	0	0	0	19	
	NRN I10	KFACT F10.0								
29	4	1.0								Channel 3
	RNODE I10	HBED F10.0	LENGTH F10.0	KBED F10.0	BBED F10.0	JOIN I10	SET I10	ITABLE I10	INDRIV I10	
30	50	110.0	3000.0	2.0	5.0	0	3	1	20	
30	52	100.0	3000.0	2.0	5.0	0	0	2	20	
30	54	90.0	3000.0	2.0	5.0	2	0	2	20	
30	56	80.0	3000.0	2.0	5.0	0	5	2	20	
	NTABLE I10									
31	2									

Table 13. Example of inputs for river-aquifer interactions (Records 27 through 35 for MODEL.DAT)--Continued

Record	Input records					Remarks
	NPT I10					
32	5					
	QTAB F10.0	WTAB F10.0	DTAB F10.0			
33	0.000	0.000	0.000			
	0.116	5.549	0.185			
	3.169	19.164	0.640			
	51.416	54.479	1.819			
	780.869	114.605	4.879			
	NPT I10					
32	5					
	QTAB F10.0	WTAB F10.0	DTAB F10.0			
33	0.000	0.000	0.000			
	0.052	4.772	0.159			
	1.569	17.150	0.573			
	26.158	49.254	1.645			
	402.788	109.228	4.426			
	RFACT F10.0					
34	1.00					
	QIN F10.0	QIN F10.0	QIN F10.0	QIN F10.0	QIN F10.0	(5F10.0)
35	37.472	266.579	140.913	30.334	46.358	
35	33.000	26.300	15.500	15.000	50.873	
	RFACT F10.0					
34	1.00					
	QIN F10.0	QIN F10.0	QIN F10.0	QIN F10.0	QIN F10.0	(5F10.0)
35	5.479	6.587	4.420	6.164	5.871	
35	4.017	1.952	0.000	0.000	0.000	

Table 13. Example of inputs for river-aquifer interactions (Records 27 through 35 for MODEL.DAT)--Continued

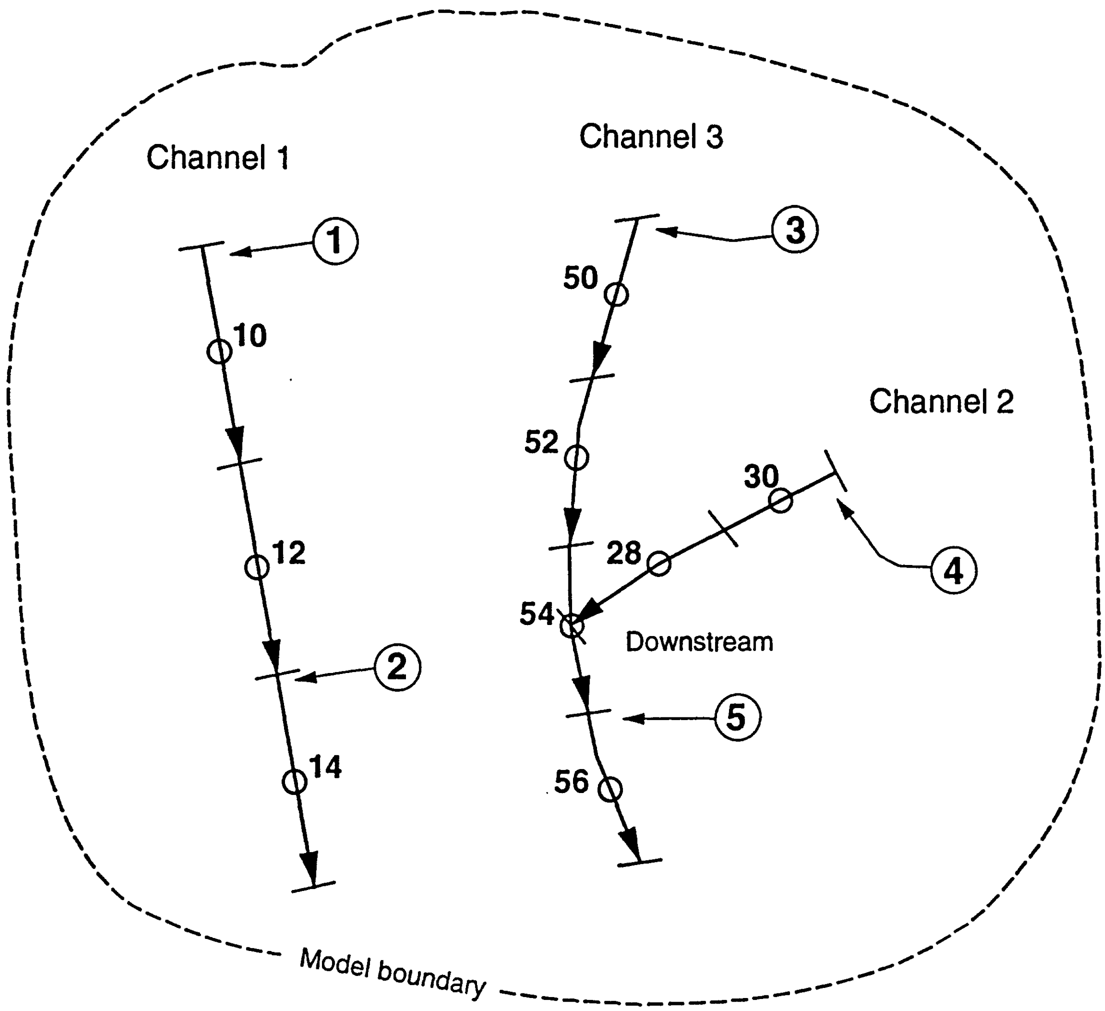
Record	Input records					Remarks
	RFACT F10.0					
34	1.00					
	QIN F10.0	QIN F10.0	QIN F10.0	QIN F10.0	QIN F10.0	(5F10.0)
35	62.000	193.189	126.457	82.000	75.849	
35	59.000	71.000	67.000	49.000	18.000	
	RFACT F10.0					
34	1.00					
	QIN F10.0	QIN F10.0	QIN F10.0	QIN F10.0	QIN F10.0	(5F10.0)
35	55.382	81.000	77.000	49.000	46.000	
35	20.000	2.000	19.000	25.000	39.843	
	RFACT F10.0					
34	1.00					
	QIN F10.0	QIN F10.0	QIN F10.0	QIN F10.0	QIN F10.0	(5F10.0)
35	4.958	2.098	2.420	0.699	3.562	
35	6.184	0.309	0.000	0.781	0.813	

Notes:

1. If NRCH equals zero, then Records 28 through 35 are omitted.
2. If the switches IEHCO1, IECHO2, and IECHO3 equal 1, then displays occur. If the switches equal zero, then no displays occur.
3. Two options specify the relations between the geometry of a reach and the flow in the river at the reach. If ITABLE (*i,j*) in Record 30 equals zero, the coefficients in Record 28 are used for the reach. The coefficients and exponents in Record 28 are described in Section 3.5.2. If these coefficients are applied to all of the reaches of the river, NTABLE in Record 31 equals zero and Records 32 and 33 are omitted. Otherwise, a table number is assigned to the river reach in Record 30 and a reach-geometry table is used. The reach-geometry table is specified in Records 32 and 33. If a reach-geometry table is specified for all reaches in a river, the coefficients and exponents in Record 28 are set equal to zero.
4. Record 30 is repeated for NRN(*i*) river nodes in a reach. Then, Records 29 and 30 are repeated for NRCH river reaches as a group.
5. The river nodes RNODE(*i,j*) in Record 30 must not be shared. In other words, a node in the finite-element grid that is assigned to a node reach within any channel reach cannot also be assigned to another node reach within any channel reach.
6. INDRIV in Record 30 corresponds to the global parameters within the subroutine *SEARCH*, described in Section 5.4.
7. If NTABLE equals zero, then Records 32 and 33 are omitted.
8. Record 33 is repeated for each entry in the reach-geometry table [NPT(*i*) times], and Records 32 and 33 are repeated for each reach-geometry table (NTABLE times) as a group.
9. Record 35 is repeated for each time step until NSTEP values of QIN(*i,j*) are input. Record 34 and 35 are repeated for each river inflow input location (NINF times) as a group.

An example of inputs for a layout of a river system is given in table 13. This example is based on the layout shown in figure 23, which includes three channels that make up two separate river systems. The first system includes channel 1; the second system includes channels 2 and 3, where channel 2 is tributary to channel 3. Lateral inflows occur at five locations. Inflow inputs 1 and 2 occur within channel 1, inflow input 4 occurs within channel reach 2, and inflow inputs 3 and 5 occur within channel 3. Each channel is divided into node reaches: channel 1 has three node reaches, channel 2 has two node reaches, and channel 3 has four node reaches.

The ordering of channel reaches in the input data set (table 13) must follow specific ordering rules. Within subroutine *RIVER*, the routing of riverflow is done in a specific order for each channel and then for each node reach. Riverflow is routed down channels in order. If a channel is tributary to another channel, flow must be routed down the tributary channel before flow is routed down the receiving channel. Therefore, the ordering rule specifies that a channel tributary to another channel must appear in the input data set prior (but not necessarily just prior) to the receiving channel. If a channel is not tributary, the input data set can be in any order. In the example in table 13, flow must be routed down channel 2 before channel 3. However, channel 1, which receives no tributary flow, could have been listed anywhere in the order.



EXPLANATION

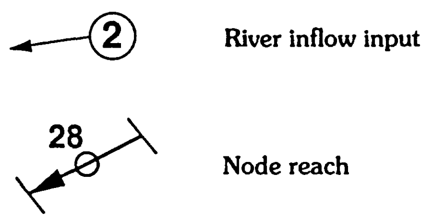


Figure 23. Representation of river network for river-aquifer interaction.

Two alternatives can be used for specifying reach geometry. In channels 1 and 2, the exponential relations for riverflow depth and width (Record 28) are used for each of the node reaches, and ITABLE in Record 30 is set to zero. In channel 3, tabulated values of width and depth are used. ITABLE in Record 30 is set to 1 for the first node reach in channel 3. The other three node reaches in channel 3 use table 2. Records 32 and 33 specify the reach geometry.

5.2.12 Ground-Water Evapotranspiration (*EVAP*)

The inputs for ground-water evapotranspiration describe the consumption of ground water by phreatophytes or by bare-soil evaporation. These inputs are listed below. (See table 14 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
36	1-10	I10	NETN	Number of evapotranspiration nodes.
	11-20	I10	MAXKNS	Number of time steps.
	21-30	I10	IECH01	Switch for display of inputs.
	31-40	I10	IECH02	Switch for display of discharges from evapotranspiration nodes.
37	1-10	I10	ETNODE(*i*)	Identity of evapotranspiration node.
	11-20	F10.0	ETAREA(*i*)	Area of evapotranspiration for node [acres].
	21-30	F10.0	LAND(*i*)	Land-surface elevation at node [ft].
	31-40	F10.0	DELH0(*i*)	Extinction depth for node [ft].
	41-50	F10.0	ETMAX(*i*)	Maximum evapotranspiration rate for node [ft/yr].
38	1-10	I10		Number of time step (for reference only).
	11-20	F10.0	FACSET(*i*)	Multiplication factor of maximum evapotranspiration for time step [dimensionless].

Table 14. Example of inputs for ground-water evapotranspiration (Records 36 through 38 for MODEL.DAT)

Record	Input records				
	NETN I10	MAXKNS I10	IECHO1 I10	IECHO2 I10	
36	6	24	1	1	
	ETNODE I10	ETAREA F10.0	LAND F10.0	DELHO F10.0	ETMAX F10.0
37	49	62.0	10.0	15.0	3.15
37	57	62.0	10.0	15.0	3.15
37	77	45.0	12.5	15.0	3.15
37	85	121.0	12.5	15.0	3.15
37	108	44.0	19.5	15.0	3.15
37	116	44.0	19.5	15.0	3.15
	time step	FACSET F10.0			
38	1	0.99			
38	2	0.69			
38	3	0.50			
38	4	0.51			
38	5	0.61			
38	6	0.86			
38	7	1.07			
38	8	1.27			
38	9	1.38			
38	10	1.49			
38	11	1.44			
38	12	1.19			
38	13	0.99			
38	14	0.69			
38	15	0.50			
38	16	0.51			
38	17	0.61			
38	18	0.86			
38	19	1.07			
38	20	1.27			
38	21	1.38			
38	22	1.49			
38	23	1.44			
38	24	1.19			

Notes:

1. If NETN equals zero, Records 37 and 38 are omitted.
2. If the switches IECHO1 and IECHO2 equal 1, displays occur. If the switches equal zero, no displays occur.
3. The number of time steps MAXKNS in Record 36 must be greater than or equal to MAXKNS in Record 2.
4. Record 37 is repeated for each evapotranspiration node (NETN times).
5. Record 38 is repeated for each time step (MAXKNS times).

5.2.13 Variable-Flux Boundary Conditions (*VFLUX*)

The inputs for variable-flux boundary conditions describe, in part, how the simulation relates to conditions outside the modeled volume. These inputs are listed below. (See table 15 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
39	1-10	I10	NVFB	Number of variable-flux nodes.
	11-20	I10	IECH01	Switch for display of inputs.
	21-30	I10	IECH02	Switch for display of discharges through variable-flux nodes.
40	1-10	I10	VFNODE(*i*)	Identity of variable-flux node.
	11-20	F10.0	K(*i*)	Hydraulic conductivity of the extended ground-water system [ft/d].
	21-30	F10.0	SS(*i*)	Specific storage of the extended ground-water system [^{1}/ft].
	31-40	F10.0	WIDTH(*i*)	Width of extended ground-water system for node [ft].
	41-50	F10.0	HEIGHT(*i*)	Height of extended ground-water system for node [ft].

Table 15. Example of inputs for variable-flux boundary conditions (Records 39 and 40 for MODEL.DAT)

Record	Input records				
	NVFB	IECHO1	IECHO2		
	I10	I10	I10		
39	6	1	1		
	VFNODE	K	SS	WIDTH	HEIGHT
	I10	F10.0	F10.0	F10.0	F10.0
40	5	33.0	1.0E-05	5280.0	80.0
40	9	33.0	1.0E-05	5280.0	85.0
40	13	33.0	1.0E-05	5280.0	90.0
40	17	33.0	1.0E-05	5280.0	95.0
40	21	33.0	1.0E-05	5280.0	100.0
40	25	33.0	1.0E-05	5280.0	105.0

Notes:

1. If NVFB equals zero, Record 40 is omitted.
2. If the switches IECH01 and IECH02 equal 1, displays occur. If the switches equal zero, no displays occur.
3. Record 40 is repeated for each variable-flux node (NVFB times).

An example of the layout of variable-flux boundaries is shown in figure 24. For this boundary condition, an extended ground-water system is associated with a node on the boundary surface (VFNODE). The extended ground-water system has a finite width (WIDTH) and height (HEIGHT), but it extends an infinite distance away from the boundary surface.

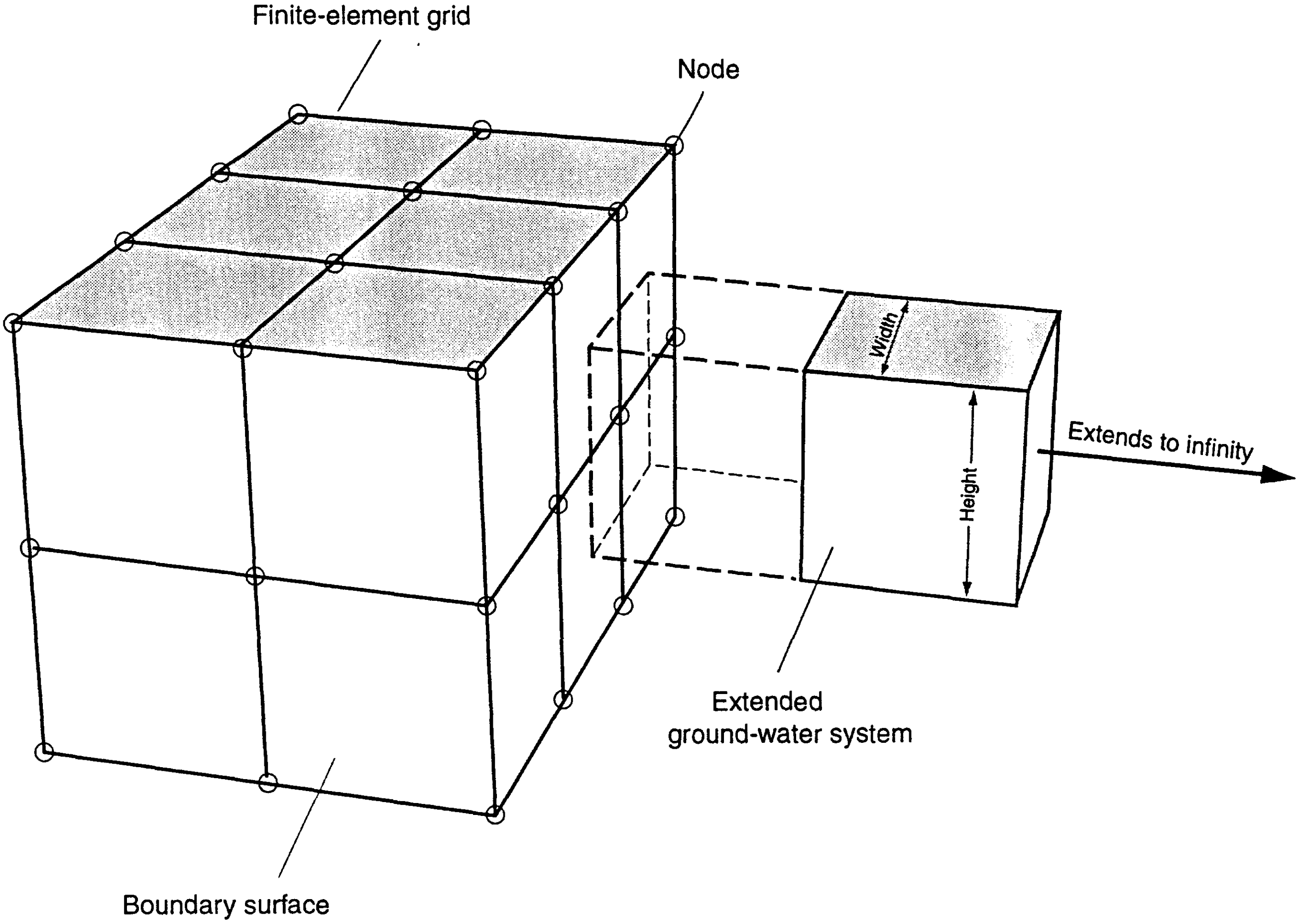


Figure 24. Representation of variable-flux boundary condition.

In the expected application of the variable-flux boundary condition, the width and height of the extended ground-water system extends one-half the distance from the associated node to the adjacent nodes on the boundary surfaces. Each node on the boundary surface has an extended ground-water system with a height and width such that the entire boundary surface is included. The sum of the end areas of the extended ground-water systems should equal the total area of the boundary surface.

5.2.14 Fault Internal Condition (*FAULT*)

The inputs for fault internal conditions represent the flow of ground water through highly transmissive features, such as fault zones or well casings. These inputs are listed below. (See table 16 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
41	1-10	I10	NFN	Number of node pairs.
	11-20	I10	IECH01	Switch for display of inputs.
	21-30	I10	IECH02	Switch for display of discharges between node pairs.
42	1-10	I10	FNODE1(*i,j*)	First node of node pair.
	11-20	I10	FNODE2(*i,j*)	Second node of node pair.
	21-30	F10.0	TRAN(*i*)	Transmissivity of feature for node pair parallel to node link, which is the connecting line between the node pair [ft^2/d].
	31-40	F10.0	LENGTH(*i*)	Length of node link [ft].
	41-50	F10.0	HEIGHT(*i*)	Height or width of the part of the feature represented by the node pair [ft].

Table 16. Example of inputs for fault internal conditions (Records 41 and 42 for MODEL.DAT)

Record Input records

```
Example for fault link:

              NFN    IECHO1    IECHO2
              I10       I10       I10

41              2         1         1

           FNODE1    FNODE2      TRAN    LENGTH    HEIGHT
              I10       I10     F10.0     F10.0     F10.0

42             51        61    1000.0     400.0     500.0
42             41        42    1000.0     500.0     400.0

Example for well link

              NFN    IECHO1    IECHO2
              I10       I10       I10

41              2         1         1

           FNODE1    FNODE2      TRAN    LENGTH    HEIGHT
              I10       I10     F10.0     F10.0     F10.0

42             30        31   2.0E+06     500.0       1.0
42             31        32   2.0E+06     500.0       1.0
```

Notes:

1. If NFN equals zero, Record 42 is omitted.
2. If the switches IECH01 equal 1, displays occur. If the switches equal zero, no displays occur.
3. Record 42 is repeated for each node pair (NFN times).
4. If the highly permeable feature is a fault, two sets of fault links must be specified to represent both horizontal and vertical ground-water flow within a fault plane: one set links nodes in the horizontal direction, and the other set links nodes in the vertical direction. Each set of fault links should separately represent the total area of the fault surface.
5. If the highly permeable feature is a well casing, the product of transmissivity times the width (diameter of the well) is the conductivity of the well casing to vertical flow.

The node-pair links allow the representation of ground-water flow through a highly permeable feature, where the feature can transmit significantly more water than the adjacent aquifer continuum.

Two examples of inputs for the layout of the fault internal conditions are given in table 16. The examples are based on the layouts shown in figure 25. The first example is for fault links and the second example is for well links (figs. 25*A* and 25*B*, respectively).

In the example for fault links, a single layer of nodes define the geometry of the fault plane. The height of the fault link extends one-half the distance above and below the line between the two nodes that define the link to the adjacent node pairs on the fault surface.

In the first example, two fault links are represented (NFN = 2); the fault links link nodes 51 and 61 (FNODE1 = 51 and FNODE2 = 61) and nodes 41 and 42 (FNODE1 = 41 and FNODE2 = 42). The height of the first fault link is 500 ft (HEIGHT = 500.0) and the length is 400 ft (LENGTH = 400.0). The transmissivity of the fault link is 1,000 ft^2/d (TRAN = 1000.0). This transmissivity represents the product of a unit thickness normal to the plane of the fault and the hydraulic conductivity of the fault zone parallel to the fault node link. If the thickness of a fault is 2 ft, the hydraulic conductivity that corresponds to the transmissivity would be 500 ft/d.

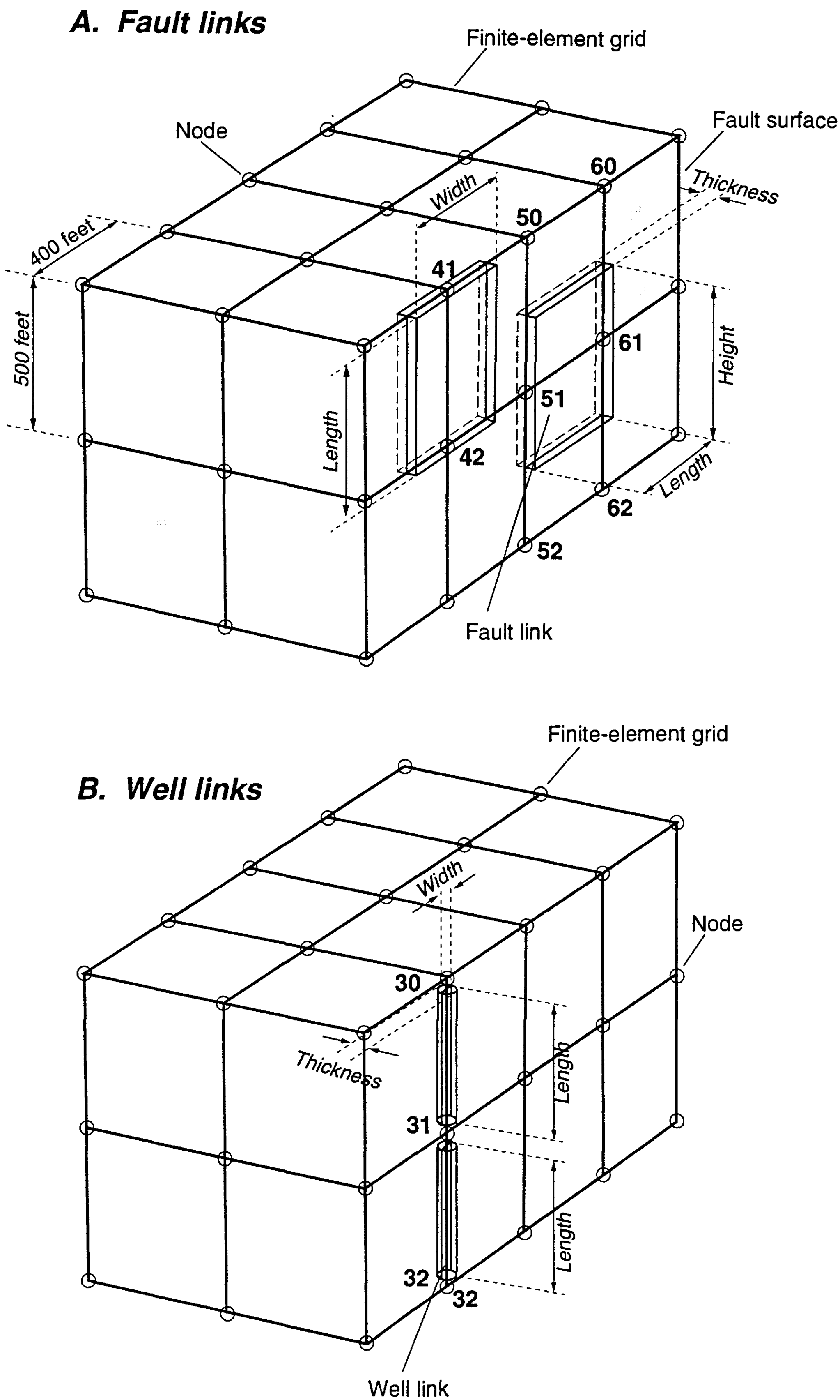


Figure 25. Representation of fault internal condition.

In the first example, the first fault link is oriented horizontally, and the second fault is oriented vertically (fig. 25A). In both examples, the length of the fault link is the length between the two nodes that define the link. However, the connecting line is horizontal in the first example and vertical in the second. In the first example, the height is a vertical dimension of the fault link normal to the line between the two nodes that define the link. In the second example, the width is a horizontal dimension. In both examples, transmissivity represents the resistance to flow between the node pairs.

In the second example, the highly permeable feature is represented by a well casing in which a single line of nodes define the length of the casing perforations. A well link represents the preferential flow of ground water through a well casing. For a well that is perforated within different layers of the finite-element grid, well links can be used to represent hydraulic head that is nearly equal throughout a well casing, and hydraulic head within the ground-water system that is nearly equal adjacent to the perforated interval of the well.

In the second example, two well links are represented (NFN = 2), the well links link nodes 30 and 31 (FNODE1 = 30 and FNODE2 = 31) and nodes 31 and 32 (FNODE1 = 31, and FNODE2 = 32). The width of the well links is 1 ft (WIDTH = 1.0) and the length is 500 ft (LENGTH = 500.0). The transmissivity of the well links is 2.0E+06 ft^2/d. This represents a transmissivity that is large enough so that the differential in hydraulic head across the well link is small. If the well was pumping at 2 ft^3/s, the transmissivity would produce a differential of 0.086 ft, where the head differential equals the pumping rate divided by the transmissivity using a consistent set of dimensional units.

5.2.15 Land Subsidence (*SINK*)

The inputs for land subsidence describe the simulation of land subsidence owing to water-level declines. These inputs are listed below. (See table 17 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
43	1-10	I10	ISIN	Switch for simulating land subsidence.
	11-20	I10	IECH01	Switch for display of aquifer parameters.
	21-30	I10	IECH02	Switch for display of initial preconsolidation heads.
	31-40	I10	IECH03	Switch for display of preconsolidation heads at end of time step.
	41-50	I10	IECH04	Switch for display of element columns.
	51-60	I10	IECH05	Switch for display of land subsidence for each element column.
44	1-10	F10.0	FACSKE	Multiplication factor for elastic specific storage of interbeds [dimensionless].
	11-20	F10.0	FACSKV	Multiplication factor for inelastic specific storage of interbeds [dimensionless].
	21-30	F10.0	FACPOR	Multiplication factor for proportion of element occupied by interbeds [dimensionless].
45	1-10	I10		Element number (for reference only).
	11-20	F10.0	SSKE(*i*)	Elastic specific storage for interbeds [$^1/ft$].
	21-30	F10.0	SSKV(*i*)	Inelastic specific storage for interbeds [$^1/ft$].
	31-40	F10.0	POR(*i*)	Proportion of prismatic element occupied by interbeds [dimensionless].
46	1-50	5F10.0	HP(*i*)	Initial preconsolidation heads [ft].
47	1-10	I10	NCOL	Number of element columns.
	11-20	I10	NSTACK	Maximum number of elements in element column.
48	1-10	I10		Number of element column (for reference only).
	11-60	5I10	COL(*i,j*)	Identification of elements in an element column.

Table 17. Example of inputs for land subsidence (Records 43 through 48 for MODEL.DAT)

Record	Input records						Remarks
	ISINK I10	IECHO1 I10	IECHO2 I10	IECHO3 I10	IECHO4 I10	IECHO5 I10	
43	1	1	1	1	1	1	
	FACSKE F10.0	FACSKV F10.0	FACPOR F10.0				
44	1.0	1.0	1.0				
	element number	SSKE F10.0	SSKV F10.0	POR F10.0			
45	1	1.0E-06	5.51E-05	1.0			
45	2	1.0E-06	5.51E-05	1.0			
45	3	1.0E-06	5.51E-05	1.0			
45	4	1.0E-06	5.51E-05	1.0			
45	5	1.0E-06	5.51E-05	1.0			
45	6	1.0E-06	5.51E-05	1.0			
45	7	1.0E-06	5.51E-05	1.0			
45	8	1.0E-06	5.51E-05	1.0			
45	9	1.0E-06	5.51E-05	1.0			
45	10	1.0E-06	5.51E-05	1.0			
45	11	1.0E-06	5.51E-05	1.0			
45	12	1.0E-06	5.51E-05	1.0			
45	13	1.0E-06	5.51E-05	1.0			
45	14	1.0E-06	5.51E-05	1.0			
45	15	1.0E-06	5.51E-05	1.0			
45	16	1.0E-06	5.51E-05	1.0			
45	17	1.0E-06	5.51E-05	1.0			
45	18	1.0E-06	5.51E-05	1.0			
45	19	1.0E-06	5.51E-05	1.0			
45	20	1.0E-06	5.51E-05	1.0			
	HP F10.0	HP F10.0	HP F10.0	HP F10.0	HP F10.0		(5F10.0)
	0.0	0.0	0.0	0.0	0.0		
46	0.0	0.0	0.0	0.0	0.0		
46	0.0	0.0	0.0	0.0	0.0		
46	0.0	0.0	0.0	0.0	0.0		
46	0.0	0.0	0.0	0.0	0.0		
46	0.0	0.0	0.0	0.0	0.0		
46	0.0	0.0	0.0	0.0	0.0		
46	0.0	0.0	0.0	0.0	0.0		
46	0.0	0.0	0.0	0.0			
	NCOL I10	NSTACK I10					
47	2	10					
	element number	COL I10	COL I10	COL I10	COL I10	COL I10	(5I10)
48	1	1	3	5	7	9	
		11	13	15	17	19	
48	2	2	4	6	8	10	
		12	14	16	18	20	

Notes:

1. If the switch ISINK equals zero, Records 44 through 48 are omitted.
2. If the switches IECH01, IECH02, IECH03, IECHO4, and IECH05 equal 1, displays occur. If the switches equal zero, no displays occur.
3. Record 45 is repeated for each prismatic element (NE times), where NE is defined in Record 9.
4. Record 46 is repeated for each node until NN values have been entered, where NN is defined in Record 9.
5. Record 48 is repeated for each element in the element column until NSTACK elements in a column have been input. If a particular column has less than NSTACK elements, zeros are entered until NSTACK elements in a column have been input.
6. Record 48 is repeated for each element column. NSTACK elements for a column are input, then a new record is started and the elements for the next column are input.

5.2.16 Controls of an Irrigated System (*WATER*)

The inputs for control of an irrigated system set switches for the output of inputs to subroutine *WATER*. These inputs are listed below. (See table 18 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
49	1-10	I10	ICONT	Switch for use of subroutine *WATER*.
50	1-10	I10	IOUT1	Switch for display of well inventory (WELL.OUT).
	11-20	I10	IOUT2	Switch for display of monthly pumping (PUMP.OUT).
	21-30	I10	IOUT3	Switch for display of pumping construction (CPUMP.OUT).
	31-40	I10	IOUT4	Switch for display of user inventory (USER.OUT).
	41-50	I10	IOUT5	Switch for display of monthly deliveries (DELIVERY.OUT).
	51-60	I10	IOUT6	Switch for display of delivery construction (CDELIVER.OUT).
51	1-10	I10	IOUT7	Switch for display of crop inventory (CROP.OUT).
	11-20	I10	IOUT8	Switch for display of rooting depth (ROOT.OUT).
	21-30	I10	IOUT9	Switch for display of monthly precipitation (PRECIP.OUT).
	31-40	I10	IOUT10	Switch for display of monthly potential evapotranspiration (PET.OUT).
	41-50	I10	IOUT11	Switch for display of crop and weather construction (CCROP.OUT).
	51-60	I10	IOUT12	Switch for display of recharge factors (HARDPAN.OUT).
52	1-10	I10	IOUT13	Switch for display of pumping destinations (DESTIN.OUT).
	11-20	I10	IOUT14	Switch for display of well exclusions (EXCLUDE.OUT).

Table 18. Example of inputs for output controls for irrigation system data used in subroutine WATER (Records 49 through 52 for MODEL.DAT)

Record	Input records					
	ICONT I10					
49	1					
	IOUT1 I10	IOUT2 I10	IOUT3 I10	IOUT4 I10	IOUT5 I10	IOUT6 I10
50	1	1	1	1	1	1
	IOUT7 I10	IOUT8 I10	IOUT9 I10	IOUT10 I10	IOUT11 I10	IOUT12 I10
51	1	1	1	1	1	1
	IOUT13 I10	IOUT14 I10				
52	1	1				

Notes:

1. If subroutine *WATER* is not used, ICONT in Record 49 is zero, and Records 50 through 52 are omitted.
2. If the switches IECHO1 through IECHO14 equal 1, displays occur. If the switches equal zero, no displays occur.
3. File names in parentheses in the "variable description" column indicate suggested file names only.

5.3 Subroutine *WATER* Input Files

The subroutine *WATER* input files contain the description of the irrigated-agricultural system. The formats of the 15 input files include fields that are not used directly by subroutine *WATER* and are for reference only. Those fields are included to allow storage of more complete data on the irrigated-agricultural system, where that data would be available for other purposes. The fields that are used by subroutine *WATER* are indicated by the occurrence of a variable name.

The file names shown in parentheses in the tables and text are suggested file names only and may be named differently for each application. The file names for subroutine *WATER* used in this report are WELL.DAT, WSTAT.DAT, PUMP.DAT, CPUMP.DAT, USER.DAT, DELIVER.DAT, CDELIVER.DAT, CROP.DAT, ROOT.DAT, PRECIP.DAT, PET.DAT, CCROP.DAT, DESTIN.DAT, RECHARGE.DAT, and EXCLUDE.DAT).

5.3.1 Well-Site Inventory File (WELL.DAT)

The inputs for well-site inventory describe well characteristics. These inputs are listed below. (See table 19 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1	A1	LINE	Record type (=A).
	3-14	A12	WNUM(*i*)	Well number.
	16-35	A20		Local well name (for reference only).
	38-40	A3		Township (for reference only).
	41-43	A3		Range (for reference only).
	44-45	I2		Section (for reference only).
	46	A1		40-acre parcel (for reference only).
	47-48	I2		Sequence number (for reference only).
	49-50	A2		Well type (for reference only).
	53-70	A18		Account number (for reference only).
	72-80	I9		Meter number (for reference only).
2	1	A1	LINE	Record type (=B).
	3-14	A12	WNUM(*i*)	Well number.
	22-30	I9	WNODE(*i,j*)	Pumpage node number.
	32-40	F9.0	WFACT(*i,j*)	Pumpage proportion for node [dimensionless].
3	1	A1	LINE	Record type (=D).
	3-14	A12	WNUM(*i*)	Well number.
	22-30	F9.0	WCAP(*i*)	Well capacity [gal/min].
	32-40	I9		Data estimate code (well capacity) (for reference only.)
	42-50	F9.0		Land-surface elevation [ft] (for reference only).
	52-60	F9.0		Well-casing length [ft] (for reference only).
	62-70	F9.0		Well depth [ft] (for reference only).
	72-80	I9		Data estimate code (well depth) (for reference only).
4	1	A1		Record type (=X) (for reference only).
	3-14	A12		Well number (for reference only).
	23-30	F9.0		Top of perforation [ft] (for reference only).
	33-40	F9.0		Bottom of perforation [ft] (for reference only).
5	1	A1		Record type (=*Y*) (for reference only).
	3-14	A12		Well number (for reference only).
	22-40	F19.0		*X* California coordinate [ft] (for reference only).
	42-60	F19.0		*Y* California coordinate [ft] (for reference only).

Table 19. Example of inputs for well-site inventory file (WELL.DAT)

Record **Input records**

```
         L                            township, range,
         I                     local  section, parcel,
         N                     well   and sequence
         E   WNUM              name   number
         A1   A12              A20    A3

1        A  MIS001             WELL 1 7N/34W-34B1

         L
         I
         N
         E   WNUM                   WNODE      WFACT
         A1   A12                      I9       F9.0

2        B  MIS001                    447      0.137
         B  MIS001                    448      0.119
         B  MIS001                    449      0.001
         B  MIS001                    452      0.266
         B  MIS001                    453      0.230
         B  MIS001                    454      0.001
         B  MIS001                    488      0.096
         B  MIS001                    489      0.083
         B  MIS001                    490      0.000

         L
         I                                      data      land-      well-                data
         N                                  estimate    surface     casing       well  estimate
         E   WNUM                    WCAP       code  elevation     length      depth      code
         A1   A12                    F9.0         I9       F9.0       F9.0       F9.0        I9

3        D  MIS001                 2500.0          0      102.0      192.0      192.0         0

         L
         I
         N    well               top of   bottom of
         E  number          perforation perforation
         A1   A12                  F9.0        F9.0

4        X  MIS001                 97.0       192.0

         L
         I
         N    well                  X California         Y California
         E  number                    coordinate           coordinate
         A1   A12                          F19.0                F19.0

5        Y  MIS001                     1264625.0             428812.5
```

Notes:

1. Record 2 is repeated for each node associated with the well. Record 4 is repeated for each perforation interval in a well. If there are no perforation data available, Record 4 is omitted. Records 1 through 5 are repeated as a group for each well.

2. For each well, WFACT(*i,j*) in Record 2 must sum to 1.

3. In Record 3, data estimate codes are optional and can be used to reference data sources.

The distribution of the pumpage from a well to nodes in the finite-element grid depends on the location of the well and the depth interval of the well screen or perforations. Pumpage for a well that is screened entirely within one element, as shown in figure 26, is distributed proportionally to each of the six nodes that define that element. Two pumpage distributions occur: one distribution is horizontal and one distribution is vertical.

The geometry for the horizontal distribution is shown in figure 26. For the horizontal distribution, the proportion of pumpage assigned to node i is given by the relation

$$p_i = \frac{a_i}{b_i}, \qquad (5.3\text{-}1)$$

where

p_i is the pumpage proportion for node i [dimensionless],
a_i is the perpendicular distance from the side opposite node i to the well [L], and
b_i is the perpendicular distance from the side opposite node i to node i [L].

The proportions for the nodes j and k are given by similar relations. The sum of the proportions for the three nodes, i, j, and k will equal 1.0.

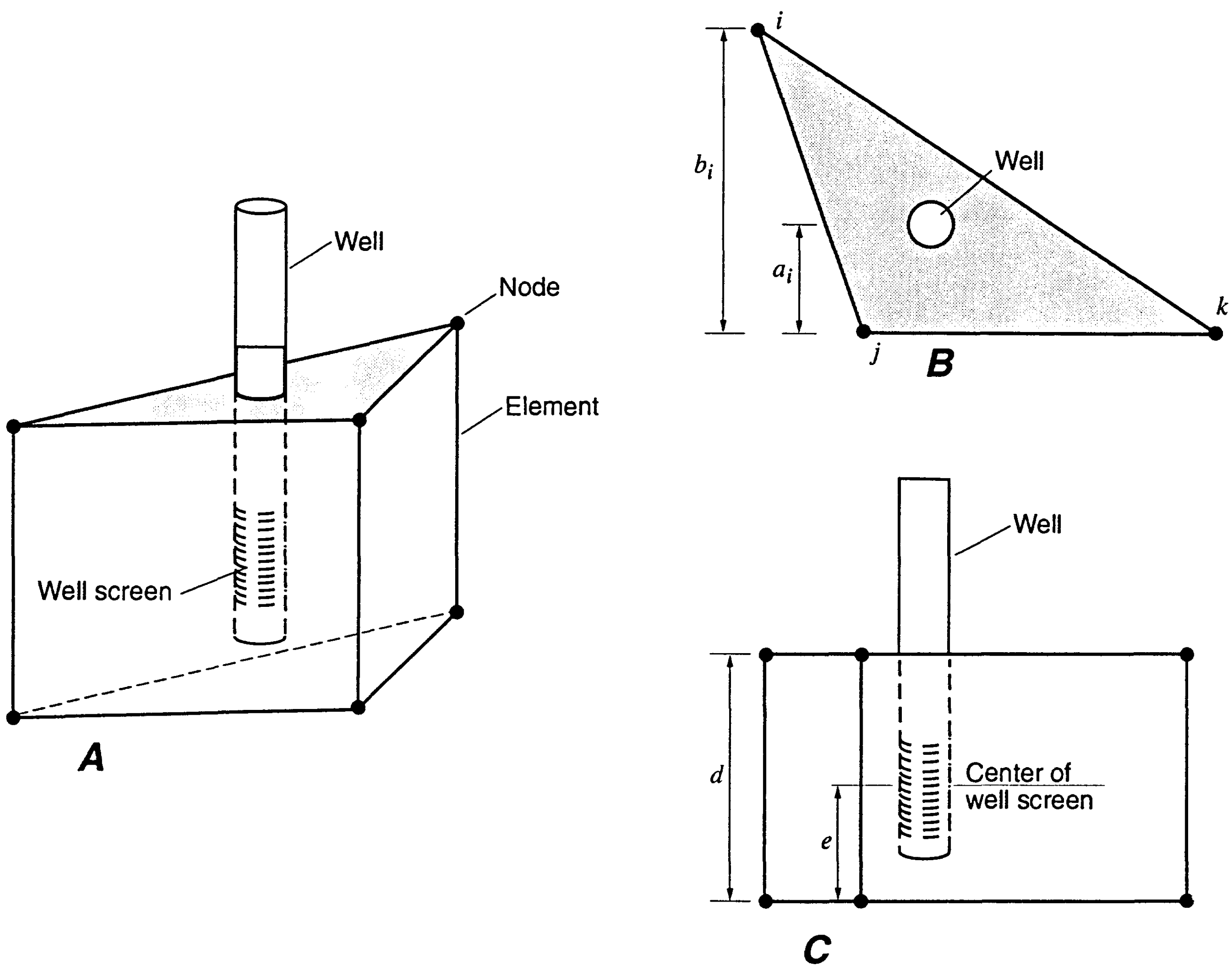


Figure 26. Distribution of pumping from wells to nodes in finite-element grid.

The geometry for the vertical distribution is shown in figure 26. For the vertical distribution, the proportion of pumpage assigned to the top of the element is given by the relation

$$p_T = \frac{e}{d}, \tag{5.3-2}$$

where

p_T	is the pumpage proportion for the top of the element [dimensionless],
e	is the distance from the bottom of the element to the center of the well screen [L], and
d	is the height of the element [L].

The proportion of pumpage assigned to the bottom of the element is given by the relation

$$p_B = l - p_T, \tag{5.3-3}$$

where p_B is the pumpage proportion for the bottom of the element [dimensionless]. The sum of these proportions will equal 1.0.

The horizontal and vertical distribution of pumpage must be combined. Using node i as an example, the combined proportion is given by the relation

$$P_i = p_i p_T, \tag{5.3-4}$$

where P_i is the combined proportion [dimensionless]. The proportions for the other nodes that define the top and bottom of the element are similar.

5.3.2 Well-Status File (WSTAT.DAT)

The inputs for well status specify the pumping period of wells by well-status code and the assignment of user areas to wells. These inputs are as listed below. (See table 20 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1	A1	LINE	Record type (=C).
	3-14	A12	WNUM(*i*)	Well number.
	22-30	I9	WSTAT(*i,j*)	Well-status code.
	34-35	I2	BSM0(*i,j*)	Beginning month for well-status change.
	37-40	I4	BSYR(*i,j*)	Beginning year for well-status change.
	44-45	I2	ESM0(*i,j*)	Ending month for well-status change.
	47-50	I4	ESYR(*i,j*)	Ending year for well-status change.
	53-60	A8	WUSER(*i*)	User Number.

Notes:

1. Record 1 is repeated when the user number for a well changes, the well status changes, or a new beginning and ending period occurs.

Table 20. Example of inputs for well-status file (WSTAT.DAT)

Record **Input records**

```
        L
        I
        N
        E  WNUM                    WSTAT BSMO BSYR ESMO ESYR  WUSER
        A1 A12                        I9   I2   I4   I2   I4  A8

1       C  MIS001                     26   10 1941   09 1994  MISSION
1       C  MIS002                     26   10 1941   09 1994  MISSION
1       C  MIS003                     26   10 1941   09 1994  MISSION
```

The purpose of the well-status file is to assign each well to a group, to identify the construction and abandonment date for each well, and to identify the user area irrigated by each well. The grouping of wells based on the well-status code. The purpose of the well-status code is to assign pumping values to wells as a group. (Well capacity is assigned to each well in the well-inventory file.)

Generally, wells are grouped into ownership or water-use categories. For example, individual group might include municipal wells for a city, private wells within an irrigation district, and district-owned wells within an irrigation district. For example, a well-status code would be assigned to each group using three different codes to identify the municipal, private, and district-owned wells. The well status for a well can change over time, which is indicated by multiple records for the well in the well-status file. For example, a well might initially be privately owned and then converted to a district-owned well.

The well-status codes are arbitrary two-digit integers. The codes are for identification purposes and do not need to follow a particular sequence.

Wells pump to a user area for direct irrigation use or to a canal system for eventual delivery for irrigation use. The user area for direct irrigation use is identified in the well-status file. The user area for a well can change over time, which is indicated by multiple records for the well in the well-status file. A change in user area can occur with a change in the user-area boundary. For example, the boundary of a user area changes when an urban area expands into an area that was previously an agricultural area.

The date of a change in the well-status code or the user area irrigated by the well is specified in the well-status file. For wells with multiple records, the dates should represent a continuous span of time from tl first record to the last record. The construction date for the well is the begin date in the first record. The abandonment date is the date in the last record, unless the well continues to be used through the end of the simulation period. In the latter case, the end date is the end date for the simulation.

5.3.3 Monthly Well-Pumping File (PUMP.DAT)

The inputs for monthly pumpage specify the monthly pumpage for individual wells. These inputs are listed below. (See table 21 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1	A1	LINE	Record type (=F).
	3-14	A12	XWNUM	Well number.
	24-25	I2	PDMO(*i*)	Month of monthly well-pumping record.
	27-30	I4	PDYR(*i*)	Year of monthly well-pumping record.
	32-40	F9.0	QPUMP(*i*)	Pumpage [acre-ft/mo].
	42-50	I9		Data source code (for reference only).

Notes:

1. Record 1 is repeated, listing sequentially the monthly pumpage for a well. Record 1 is then repeated for each well.
2. At least one record must be entered, but the pumpage value may be set to zero.

Table 21. Example of inputs for monthly well-pumping file (PUMP.DAT)

Record	Input records				
	L I N E A1	XWNUM A12	PDMO I2	PDYR I4	QPUMP F9.0
1	F	MIS001	1	1973	26.3
1	F	MIS001	2	1973	40.8
1	F	MIS001	3	1973	23.8
1	F	MIS001	4	1973	0.1
1	F	MIS001	5	1973	53.0
1	F	MIS001	6	1973	221.7
1	F	MIS001	7	1973	192.2
1	F	MIS001	8	1973	188.4
1	F	MIS001	9	1973	214.5
1	F	MIS001	10	1973	163.5
1	F	MIS001	11	1973	98.7
1	F	MIS001	12	1973	4.1
1	F	MIS001	1	1973	22.6
1	F	MIS001	2	1973	7.8
1	F	MIS001	3	1973	147.7
1	F	MIS001	4	1973	198.2
1	F	MIS001	5	1973	227.4
1	F	MIS001	6	1973	242.9

5.3.4 Pumping-Construction File (CPUMP.DAT)

The inputs for pumping construction specify the pumping scenario for a simulation. These inputs are listed below. (See table 22 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1	A1	LINE	Record type (=G).
	4-5	I2	PCMO(*i*)	Month of pumping-construction record.
	7-10	I4	PCYR(*i*)	Year of pumping-construction record.
2	1	A1	LINE	Record type (=H).
	3-10	I8	PSCODE(*i,j*)	Well-status code.
	12-20	I9	PDFILE(*i,j*)	Data-file code.
	22-30	F9.0	RPUMP(*i,j*)	Pumpage [acre-ft/mo].
	34-35	I2	PUMO(*i,j*)	Month of record from monthly well-pumping file to use.
	37-40	I4	PUYR(*i,j*)	Year of record from monthly well-pumping file to use.
	42-50	F9.0	PFACT(*i,j*)	Adjustment factor for pumpage [dimensionless].

Table 22. Example of inputs for pumping-construction file (CPUMP.DAT)

```
Record  Input records

        L
        I
        N
        E  PCMO PCYR
        A1   I2   I4

1       G    08 1973

        L
        I
        N
        E    PSCODE    PDFILE     RPUMP PUMO PUYR       PFACT
                 I8        I9      F9.0   I2   I4        F9.0

2       H        18         0    2000.0   00 0000         1.0
2       H        26         1       0.0   08 1973         1.0
2       H        40         1       0.0   08 1983         0.7
```

Notes:

1. Record 2 is repeated for each active well-status code. Records 1 and 2 are repeated as a group for each month. Pumpage is entered if the data-file code (PDFILE) equals zero. If the data-file code equals 1, pumpage is not entered in CPUMP.DAT, and pumpage for individual wells is entered into the monthly well-pumping file (PUMP.DAT) instead.

The purpose of the pumping-construction file is to specify a pumping scenario for a simulation. For each month in the simulation, the pumpage is specified for each well-status code. The pumpage for a status code can be specified by three alternative approaches. For the first approach, if monthly pumping data for individual wells are not available for a particular month, the total pumpage for all wells within the status-code groups can be specified and the total pumpage is distributed to the individual wells in proportion to the well capacity of individual wells. For the second approach, if monthly pumping data are available for the particular month, that monthly pumping data can be used to specify the pumpage for individual wells. For the third approach, if monthly pumping data are available for a different month, that monthly pumping data can be used to specify the pumpage for the individual wells.

Examples of these three alternative approaches are given in table 22 for August 1973 (PCMO = 08 and PCYR = 1973). For the first approach [status code 18 (PSCODE = 18)], monthly pumping data were not available (PDFILE = 0) so a total for pumpage of 2,000 acre-ft was distributed to the wells (RPUMP = 2000.0). For the second approach [status code 26 (PSCODE = 26)], monthly pumping data were available (PDFILE = 1), thus monthly pumping for the current month was used (PUMO = 08 and PUYR = 1973). For the third approach [status code 40 (PSCODE = 40)], monthly pumping data were not available (PDFILE = 1) for August 1983 (PUMO = 08 and PUYR = 1983). These data were used for August 1973. However, pumpage for August 1983 was multiplied by a factor of 0.7 (PFACT = 0.7) to represent the pumpage for August 1973.

5.3.5 User-Area-Inventory File (USER.DAT)

The inputs for the user-area inventory describe the user area characteristics. These inputs are listed below. (See table 23 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1	A1	LINE	Record type (=I).
	3-10	A8	UNUM(*i*)	User number.
	12-20	I9	UTYPE(*i*)	User-type code.
	22-30	I9		Ditch-tender district number (for reference only).
2	1	A1	LINE	Record type (=Z).
	3-10	A8	UNUM(*i*)	User number.
	12-20	I9	IMO	Month of year for delivery adjustment.
	22-30	F9.0	PORDEL(*i,j*)	Proportion of delivery used for irrigation during specified month of year [dimensionless].
3	1	A1	LINE	Record type (=R).
	3-10	A8	UNUM(*i*)	User number.
	12-20	F9.0	AWC(*i*)	Available water capacity [dimensionless].
	22-30	F9.0	GAMMA(*i*)	Critical soil moisture [dimensionless].
	32-40	F9.0	UCOEF(*i*)	Uniformity coefficient [dimensionless].
	42-50	F9.0	THETA0(*i*)	Initial soil moisture [dimensionless].
	52-60	F9.0	RFACT(*i*)	Precipitation adjustment factor [dimensionless].
	62-70	F9.0	CLOSS(*i*)	Canal-loss coefficient [dimensionless].
	72-80	F9.0	TAILF(*i*)	Tailwater coefficient [dimensionless].
4	1	A1	LINE	Record type (=J).
	3-10	A8	UNUM(*i*)	User number.
	12-20	F9.0	AREA(*i,j*)	Irrigated acreage of user area [acres].
	24-25	I2	BAMO(*i,j*)	Begin month.
	27-30	I4	BAYR(*i,j*)	Begin year.
	34-35	I2	EAMO(*i,j*)	End month.
	37-40	I4	EAYR(*i,j*)	End year.
5	1	A1	LINE	Record type (=K).
	3-10	A8	UNUM(*i*)	User number.
	12-20	I9	UNODE(*i,j,k*)	Recharge node number.
	22-30	F9.0	UFACT(*i,j,k*)	Recharge proportion for node [dimensionless].

Table 23. Example of inputs for user-area inventory file (USER.DAT)

```
Record  Input records

        L
        I
        N
        E  UNUM           UTYPE
        A1 A8                I9

1       I  WESTID01          36

        L
        I
        N
        E  UNUM             IMO     PORDEL
        A1 A8                I9       F9.0

2       Z  WESTID01           1       1.00
2       Z  WESTID01           2       1.00
2       Z  WESTID01           3       1.00
2       Z  WESTID01           4       1.00
2       Z  WESTID01           5       1.00
2       Z  WESTID01           6       1.00
2       Z  WESTID01           7       1.00
2       Z  WESTID01           8       1.00
2       Z  WESTID01           9       1.00
2       Z  WESTID01          10       1.00
2       Z  WESTID01          11       1.00
2       Z  WESTID01          12       1.00

        L
        I
        N
        E  UNUM             AWC      GAMMA      UCOEF   THETA0      RFACT      CLOSS      TAILF
        A1 A8              F9.0       F9.0       F9.0     F9.0       F9.0       F9.0       F9.0

3       R  WESTID01      0.1211       0.50       0.88     0.00     1.1707       0.05       0.10

        L
        I
        N
        E  UNUM            AREA BAMO BAYR EAMO EAYR
        A1 A8              F9.0   I2   I4   I2   I4

4       J  WESTID01      1458.4   10 1961    9 1962

        L
        I
        N
        E  UNUM           UNODE      UFACT
        A1 A8                I9       F9.0

5       K  WESTID01         351     0.0064
5       K  WESTID01         354     0.1072
5       K  WESTID01         358     0.0289
5       K  WESTID01         362     0.0019
5       K  WESTID01         394     0.2279
5       K  WESTID01         397     0.4423
5       K  WESTID01         401     0.0968
```

Table 23. Example of inputs for user-area inventory file (USER.DAT)--Continued

Record	Input records						
5	K	WESTID01	405	0.0016			
5	K	WESTID01	438	0.0551			
5	K	WESTID01	441	0.0310			
5	K	WESTID01	445	0.0008			
	L I N E A1	UNUM A8	AREA F9.0	BAMO I2	BAYR I4	EAMO I2	EAYR I4
4	J	WESTID01	1399.7	10	1962	9	1963
	L I N E A1	UNUM A8	UNODE I9	UFACT F9.0			
5	K	WESTID01	351	0.0064			
5	K	WESTID01	354	0.1072			
5	K	WESTID01	358	0.0289			
5	K	WESTID01	362	0.0019			
5	K	WESTID01	394	0.2279			
5	K	WESTID01	397	0.4423			
5	K	WESTID01	401	0.0968			
5	K	WESTID01	405	0.0016			
5	K	WESTID01	438	0.0551			
5	K	WESTID01	441	0.0310			
5	K	WESTID01	445	0.0008			

Notes:

1. Record 2 is repeated 12 times for each user, once for each month in a calendar year. If PORDEL is equal to 1.0, the total amount of water listed in the pumpage and delivery files assigned to a user area is used in the calculation of recharge for a user area. If PORDEL is less than 1.0, the total amount of water is adjusted by this proportion in the calculation of recharge.

2. Record 4 is repeated for each change in acreage of user area.

3. Record 5 is repeated for each node with each change in area. For a particular user the sum o the recharge proportions (UFACT) in Record 5 must equal 1.

The variables IMO and PORDEL in Record 2 can reduce the total amount of water used for irrigation within a user area on a monthly basis. For example, in an urban area, water supplies generally are used for landscape irrigation and indoor use. The disposal of water used indoors determines the value assigned to PORDEL. If the water used indoors is discharged to septic tanks, the water contributes to ground-water recharge for the area, and PORDEL equals 1.0 for all 12 months of the year. However, if water used indoors is routed to a treatment plant and then discharged to a river, this part of the urban water supply is not applied to the land, and thus, should not be used to calculate ground-water recharge for the area. If wastewater is discharged outside of the user area, the value assigned to PORDEL is less than 1.0, and a different value for PORDEL can be assigned to each month to reflect the usual monthly variation in the proportion of water used indoors and outdoors.

The distribution of the recharge from a user to the nodes in the finite-element grid depends on the intersection of the user area with the elements in the top surface of the grid. The intersection of a user area with an element is shown in figure 27. The recharge within the intersected area is distributed to the three nodes that define the top surface of the element that is intersected. The proportion of recharge from an intersected area to the total recharge from a user is equal to the proportion of irrigated acreage within the intersected area to the total irrigated acreage within the user area.

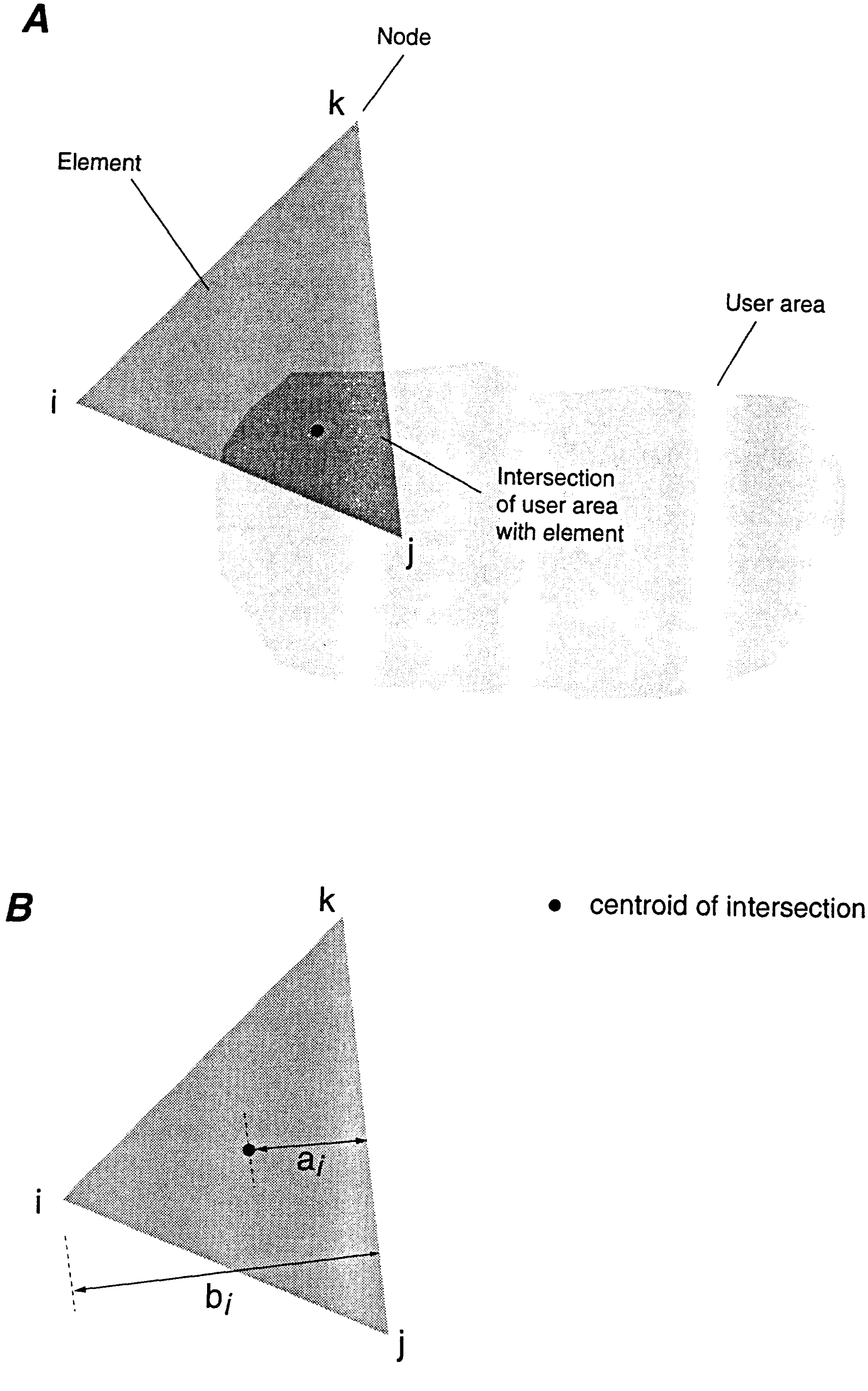


Figure 27. Distribution of recharge from user area to nodes in the finite-element grid.

The proportions of recharge to the nodes are obtained by considering the centroid of the intersected area. The geometry for the distribution of recharge from an intersected area to the node is shown in figure 27B. The proportion of recharge that is assigned to node *i* is given by the relation

$$r_i = \frac{a_i}{b_i}, \tag{5.3-5}$$

where

r_i is the recharge proportion for the node *i* for an intersected area [dimensionless],
a_i is the perpendicular distance from the side opposite node *i* to the centroid [L], and
b_i is the perpendicular distance from the side opposite node *i* to node *i* [L].

The proportions for the nodes *j* and *k* are given by similar relations. The sum of these proportions equals 1.0 for each intersection.

To calculate the recharge proportion (UFACT) for each node assigned recharge for the user area (UNODE), the recharge proportions from each intersected area must be weighted and summed for each node according to the expression

$$R_i = \sum_{I=1}^{n} r_i \frac{A_I}{A}, \tag{5.3-6}$$

where

R_i is the recharge proportion for the node *i* for the user area [dimensionless],
A_I is the irrigated acreage of the intersected area I [L^2], and
A is the irrigated acreage of the user area [L^2].

The sum of the proportions of the nodes assigned to the user area equals 1.0 for each user area in the model.

Missing pages 170 - 169
From original.

The purpose of the delivery-construction file is to specify a canal-delivery scenario for a simulation. For each month in the simulation, the canal delivery is specified for each user-type code. The canal delivery for a user-type code can be specified using three alternative approaches. For the first approach, if monthly delivery data for individual user areas are not available for a particular month, the total delivery for all user areas within a user-type group can be specified, and the total delivery is distributed to individual user areas in proportion to the acreage of individual user areas. For the second approach, if monthly delivery data are available for the particular month, that monthly delivery data can be used to specify the deliveries for individual user areas. For the third approach, if monthly delivery data are available for a different month, that monthly delivery data can be used to specify the deliveries to individual user areas.

Examples of these three approaches are given in table 25 for August 1973 (DCMO = 08 and DCYR = 1973). For the first approach [user-type code 28 (UTCODE = 28)], monthly delivery data were not available (DDFILE = 0), and a total delivery of 2,000 acre-ft was distributed to the user areas (RDEL = 2000.0). For the second approach [user-type code 36 (UTCODE = 36)], monthly delivery data were available (DDFILE = 1), and the monthly deliveries for the current month were used (DUMO = 08 and DUYR = 1973). For the third approach [user-type code 50 (UTCODE = 50)], monthly delivery data were available (DDFILE = 1) for August 1983 (DUMO = 08 and DUYR = 1983), and those data were used for August 1973. However, the deliveries for August 1983 were multiplied by a factor of 0.7 (DFACT = 0.7) to represent the deliveries for August 1973.

5.3.8 Crop-Inventory File (CROP.DAT)

The inputs for crop inventory specify the crops for each user area. These inputs are listed below. (See table 26 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1	A1	LINE	Record type (=P).
	3-10	A8	XUNUM	User number.
	17-20	I4	INYR(*i,j*)	Year of crop inventory record.
	22-30	I9	INCODE(*i,j*)	Crop code.
	32-50	A19		Crop name (for reference only).
	52-60	F9.0	INPOR(*i,j*)	Crop proportion [dimensionless].

Notes:

1. Record 1 is repeated for each user area, year of record, and crop type. If the proportion in a user area for a crop type is zero, omit that record.

Table 26. Example of inputs for crop-inventory file (CROP.DAT)

Record	Input records					
	LINE	XUNUM	INYR	INCODE	crop name	INPOR
	A1	A8	I4	I9	A19	F9.0
1	P	WESTID01	1942	13	LETTUCE	0.649
1	P	WESTID01	1942	10	COLE CROPS	0.248
1	P	WESTID01	1942	21	TRUCK (MISC)	0.103
1	P	WESTID01	1953	13	LETTUCE	0.667
1	P	WESTID01	1953	10	COLE CROPS	0.228
1	P	WESTID01	1953	21	TRUCK (MISC)	0.101
1	P	WESTID01	1966	6	FIELD (MISC)	0.477
1	P	WESTID01	1966	13	LETTUCE	0.275
1	P	WESTID01	1966	10	COLE CROPS	0.248
1	P	WESTID01	1977	05	DRY BEANS	0.921
1	P	WESTID01	1977	20	FLOWERS	0.066
1	P	WESTID01	1977	30	URBAN LANDSCAPE	0.004
1	P	WESTID01	1977	10	COLE CROPS	0.009
1	P	WESTID01	1984	5	DRY BEANS	0.886
1	P	WESTID01	1984	26	CITRUS	0.099
1	P	WESTID01	1984	6	FIELD (MISC)	0.008
1	P	WESTID01	1984	10	COLE CROPS	0.007
1	P	WESTID02	1942	29	PASTURE	1.000
1	P	WESTID02	1953	29	PASTURE	1.000
1	P	WESTID02	1966	29	PASTURE	1.000
1	P	WESTID02	1977	29	PASTURE	1.000
1	P	WESTID02	1984	9	GREEN BEANS	0.966
1	P	WESTID02	1984	29	PASTURE	0.034

5.3.9 Rooting-Depth File (ROOT.DAT)

The inputs for rooting depth specify the rooting depth for individual crops. These inputs are listed below. (See table 27 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1	A1	LINE	Record type (=W).
	3-10	I8	CCODE(*i*)	Crop code.
	12-30	A19		Crop name (for reference only.
	32-40	F9.0	ROOT(*i*)	Rooting depth [inches].

Notes:

1. Record 1 is repeated for each crop.

Table 27. Example of inputs for rooting-depth file (ROOT.DAT)

Record	Input records			
	L I N E A1	CCODE I8	crop name A19	ROOT F9.0
1	W	1	FLAX	48
1	W	2	SUGAR BEETS	66
1	W	3	CORN	36
1	W	4	SUDAN	72
1	W	5	DRY BEANS	42
1	W	6	FIELD (MISC)	42
1	W	7	ARTICHOKES	54
1	W	8	ASPARAGUS	120
1	W	9	GREEN BEANS	36
1	W	10	COLE CROPS	24
1	W	11	SWEET CORN	36
1	W	12	CARROTS	36
1	W	13	CELERY	12
1	W	14	LETTUCE	18
1	W	15	PUMPKINS	72
1	W	16	GARLIC	12
1	W	17	POTATOES	48
1	W	18	SPINACH	24
1	W	19	TOMATOES	48
1	W	20	FLOWERS	42
1	W	21	TRUCK (MISC)	42
1	W	22	STRAWBERRIES	42
1	W	23	BROCCOLI	24
1	W	24	CABBAGE	24
1	W	25	CAULIFLOWER	24
1	W	26	CITRUS	18
1	W	27	DECIDUOUS TREES	72
1	W	28	GRAIN	48
1	W	29	PASTURE	36
1	W	30	URBAN LANDSCAPE	24
1	W	31	NON-IRRIG GRAIN	48
1	W	32	NON-IRRIG PASTURE	120
1	W	33	NATIVE VEGETATION	24

5.3.10 Monthly-Precipitation File (PRECIP.DAT)

The inputs for monthly precipitation specify the monthly precipitation on user areas. These inputs are listed below. (See table 28 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1	A1	LINE	Record type (=U).
	6-9	I4	XYR	Year of precipitation record.
	10-15	F6.0	RAIN(*i*)	January precipitation [inches].
	16-21	F6.0	RAIN(*i*)	February precipitation [inches].
	22-27	F6.0	RAIN(*i*)	March precipitation [inches].
	28-33	F6.0	RAIN(*i*)	April precipitation [inches].
	34-39	F6.0	RAIN(*i*)	May precipitation [inches].
	40-45	F6.0	RAIN(*i*)	June precipitation [inches].
	46-51	F6.0	RAIN(*i*)	July precipitation [inches].
	52-57	F6.0	RAIN(*i*)	August precipitation [inches].
	58-63	F6.0	RAIN(*i*)	September precipitation [inches].
	64-69	F6.0	RAIN(*i*)	October precipitation [inches].
	70-75	F6.0	RAIN(*i*)	November precipitation [inches].
	76-81	F6.0	RAIN(*i*)	December precipitation [inches].

Notes:

1. Record 1 is repeated for each year of record.

Table 28. Example of inputs for monthly precipitation file (PRECIP.DAT)

Record	LINE A1	XYR I4	RAIN jan F6.0	RAIN feb F6.0	RAIN mar F6.0	RAIN apr F6.0	RAIN may F6.0	RAIN jun F6.0	RAIN jul F6.0	RAIN aug F6.0	RAIN sep F6.0	RAIN oct F6.0	RAIN nov F6.0	RAIN dec F6.0
1	U	1975	0.19	3.88	5.82	0.92	0.00	0.00	0.00	0.00	0.00	0.53	0.36	0.16
1	U	1976	0.00	4.68	1.40	1.31	0.03	0.16	0.02	0.57	2.67	0.53	0.80	0.89
1	U	1977	1.87	0.12	1.95	0.01	2.54	0.00	0.00	0.00	0.01	0.00	0.15	3.22
1	U	1978	6.81	8.27	7.92	2.92	0.00	0.00	0.00	0.00	1.43	0.00	1.31	1.14
1	U	1979	4.78	3.08	4.37	0.03	0.09	0.00	0.00	0.00	0.29	0.62	0.50	2.67
1	U	1980	3.69	6.34	1.96	0.65	0.11	0.00	0.02	0.00	0.00	0.00	0.00	1.14
1	U	1981	3.24	2.88	5.98	0.40	0.00	0.00	0.00	0.00	0.00	0.53	0.81	0.89
1	U	1982	2.66	0.77	4.59	2.45	0.00	0.02	0.00	0.00	0.35	1.40	3.11	1.64
1	U	1983	8.43	6.61	6.73	3.55	0.37	0.06	0.00	0.53	0.03	0.52	2.37	3.46
1	U	1984	0.06	0.38	0.52	0.63	0.00	0.00	0.00	0.18	0.07	0.25	2.50	4.63
1	U	1985	0.69	0.85	1.28	0.00	0.00	0.00	0.00	0.00	0.01	0.57	5.06	0.70
1	U	1986	1.93	4.84	5.06	0.27	0.00	0.00	0.00	0.00	0.75	0.00	2.01	0.94
1	U	1987	2.15	1.72	4.00	0.33	0.09	0.04	0.00	0.00	0.00	1.49	0.81	4.63
1	U	1988	1.85	2.28	0.19	2.67	0.12	0.21	0.00	0.00	0.00	0.00	0.77	3.08
1	U	1989	0.46	0.58	0.37	0.20	0.26	0.00	0.00	0.00	0.64	0.28	0.33	0.00
1	U	1990	2.87	1.70	0.44	0.44	0.58	0.01	0.00	0.00	0.05	0.00	0.22	0.82
1	U	1991	1.26	2.17	11.63	0.38	0.00	0.02	0.03	0.04	0.01	0.00	0.00	0.00
1	U	1992	2.69	7.06	2.31	0.01	0.00	0.02	0.06	0.00	0.00	0.49	0.00	3.63
1	U	1993	6.36	5.66	3.16	0.02	0.12	0.11	0.00	0.00	0.00	0.25	0.86	1.75
1	U	1994	1.82	4.47	2.40	0.84	0.74	0.00	0.00	0.00	0.03	0.63	2.38	0.99

5.3.11 Monthly Potential Evapotranspiration File (PET.DAT)

The inputs for monthly potential evapotranspiration (PET) specify the monthly PET for each crop. These inputs are listed below. (See table 29 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1	A1	LINE	Record type (=V).
	3-4	I2	XCODE	Crop code.
	6-9	I4	XYR	Year of potential evapotranspiration.
	10-15	F6.0	PET(*i,j*)	January PET [inches].
	16-21	F6.0	PET(*i,j*)	February PET [inches].
	22-27	F6.0	PET(*i,j*)	March PET [inches].
	28-33	F6.0	PET(*i,j*)	April PET [inches].
	34-39	F6.0	PET(*i,j*)	May PET [inches].
	40-45	F6.0	PET(*i,j*)	June PET [inches].
	46-51	F6.0	PET(*i,j*)	July PET [inches].
	52-57	F6.0	PET(*i,j*)	August PET [inches].
	58-63	F6.0	PET(*i,j*)	September PET [inches].
	64-69	F6.0	PET(*i,j*)	October PET [inches].
	70-75	F6.0	PET(*i,j*)	November PET [inches].
	76-81	F6.0	PET(*i,j*)	December PET [inches].

Notes:

1. Record 1 is repeated for each year of record and for each crop, and is entered by crop and then by year for the crop.

Table 29. Example of inputs for monthly potential evapotranspiration file (PET.DAT)

Record	Input records														
	LINE A1	XCODE I2	XYR I4	PET jan F6.0	PET feb F6.0	PET mar F6.0	PET apr F6.0	PET may F6.0	PET jun F6.0	PET jul F6.0	PET aug F6.0	PET sep F6.0	PET oct F6.0	PET nov F6.0	PET dec F6.0
1	V	1	1971	1.64	2.61	3.37	4.27	2.30	0.28	1.12	0.83	0.76	0.48	0.75	0.67
1	V	1	1972	1.82	1.96	3.82	5.35	2.95	0.29	0.95	0.81	0.59	0.38	0.41	0.54
1	V	1	1973	1.29	1.57	3.55	6.07	3.10	0.30	0.92	0.81	0.69	0.37	0.72	0.54
1	V	1	1974	1.19	2.47	3.05	4.31	3.16	0.33	0.91	0.83	0.64	0.50	0.53	0.70
1	V	1	1975	1.49	2.20	3.78	4.10	3.37	0.31	0.86	0.81	0.56	0.41	0.70	0.74
1	V	1	1976	1.90	3.08	4.93	4.96	3.64	0.37	0.96	0.74	0.56	0.38	0.70	0.97
1	V	1	1977	1.01	2.67	4.57	6.49	2.26	0.31	0.95	0.77	0.55	0.38	0.68	0.44
1	V	1	1978	0.79	1.73	3.26	4.09	3.53	0.31	0.86	0.76	0.65	0.55	0.80	0.89
1	V	1	1979	1.40	1.17	2.52	3.90	3.42	0.28	0.90	0.70	0.58	0.29	0.35	0.63
1	V	1	1980	1.04	1.63	4.63	4.16	2.64	0.25	0.83	0.75	0.60	0.45	0.72	0.69
1	V	2	1971	1.64	2.61	0.32	0.47	6.38	9.40	11.95	9.23	8.29	1.80	0.75	0.67
1	V	2	1972	1.82	1.96	0.36	0.59	8.20	9.60	10.08	9.00	6.47	1.43	0.41	0.54
1	V	2	1973	1.29	1.57	0.33	0.68	8.60	10.00	9.79	9.00	7.57	1.41	0.72	0.54
1	V	2	1974	1.19	2.47	0.29	0.48	8.78	11.07	9.75	9.21	6.96	1.91	0.53	0.70
1	V	2	1975	1.49	2.20	0.35	0.46	9.37	10.34	9.15	8.97	6.08	1.55	0.70	0.74
1	V	2	1976	1.90	3.08	0.46	0.55	10.11	12.20	10.20	8.25	6.06	1.43	0.70	0.97
1	V	2	1977	1.01	2.67	0.43	0.72	6.29	10.21	10.13	8.55	5.95	1.43	0.68	0.44
1	V	2	1978	0.79	1.73	0.31	0.45	9.81	10.29	9.16	8.39	7.04	2.09	0.80	0.89
1	V	2	1979	1.40	1.17	0.24	0.43	9.49	9.31	9.56	7.81	6.30	1.08	0.35	0.63
1	V	2	1980	1.04	1.63	0.43	0.46	7.34	8.45	8.80	8.37	6.56	1.72	0.72	0.69
1	V	3	1971	1.64	2.61	0.32	0.47	0.57	2.91	10.96	7.48	1.44	0.48	0.75	0.67
1	V	3	1972	1.82	1.96	0.36	0.59	0.74	2.98	9.24	7.29	1.12	0.38	0.41	0.54
1	V	3	1973	1.29	1.57	0.33	0.68	0.77	3.10	8.98	7.29	1.31	0.37	0.72	0.54
1	V	3	1974	1.19	2.47	0.29	0.48	0.79	3.43	8.94	7.46	1.21	0.50	0.53	0.70
1	V	3	1975	1.49	2.20	0.35	0.46	0.84	3.21	8.39	7.27	1.05	0.41	0.70	0.74
1	V	3	1976	1.90	3.08	0.46	0.55	0.91	3.78	9.35	6.68	1.05	0.38	0.70	0.97
1	V	3	1977	1.01	2.67	0.43	0.72	0.57	3.17	9.28	6.93	1.03	0.38	0.68	0.44
1	V	3	1978	0.79	1.73	0.31	0.45	0.88	3.19	8.40	6.80	1.22	0.55	0.80	0.89
1	V	3	1979	1.40	1.17	0.24	0.43	0.85	2.89	8.76	6.33	1.09	0.29	0.35	0.63
1	V	3	1980	1.04	1.63	0.43	0.46	0.66	2.62	8.07	6.78	1.14	0.45	0.72	0.69

5.3.12 Crop-Construction and Weather-Construction File (CCROP.DAT)

The inputs for the crop-construction and weather-construction file specify a crop and weather scenario for a simulation. This file selects data from the crop-inventory file, the monthly precipitation file, and the monthly potential evapotranspiration file and assigns these data to each month in the simulation. These inputs are listed below. (See table 30 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1	A1	LINE	Record type (=S).
	4-5	I2	CCMO(*i*)	Month of crop- and weather-construction record.
	7-10	I4	CCYR(*i*)	Year of crop- and weather-construction record.
2	1	A1	LINE	Record type (=T).
	3-10	I8	ETCODE(*i,j*)	User-type code.
	17-20	I4	CUYR(*i,j*)	Year of record from crop inventory file to use for crops.
24-25	I2	EUMO(*i,j*)		Month of record from precipitation and potential evapotranspi- ration files to use for weather.
	27-30	I4	EUYR(*i,j*)	Year of record from precipitation and potential evapotranspi- ration files to use for weather.

Notes:

1. Record 2 is repeated for each user-type code. Records 1 and 2 are repeated as a group for each month.

Table 30. Example of inputs for crop-construction and weather-construction file (CCROP.DAT)

Record	Input records				
	L				
	I				
	N				
	E	CCMO	CCYR		
	A1	I2	I4		
1	S	10	1941		
	L				
	I				
	N				
	E	ETCODE	CUYR	EUMO	EUYR
	A1	I8	I4	I2	I4
2	T	18	1942	10	1941
2	T	25	1942	10	1941
2	T	26	1942	10	1941
2	T	27	1942	10	1941
2	T	28	1942	10	1941
2	T	31	1942	10	1941
2	T	32	1942	10	1941
2	T	33	1942	10	1941
2	T	34	1942	10	1941
2	T	35	1942	10	1941
2	T	36	1942	10	1941
2	T	40	1942	10	1941
2	T	50	1942	10	1941
	L				
	I				
	N				
	E	CCMO	CCYR		
	A1	I2	I4		
1	S	11	1941		
	L				
	I				
	N				
	E	ETCODE	CUYR	EUMO	EUYR
	A1	I8	I4	I2	I4
2	T	18	1942	11	1941
2	T	25	1942	11	1941
2	T	26	1942	11	1941
2	T	27	1942	11	1941
2	T	28	1942	11	1941
2	T	31	1942	11	1941
2	T	32	1942	11	1941
2	T	33	1942	11	1941
2	T	34	1942	11	1941
2	T	35	1942	11	1941
2	T	36	1942	11	1941
2	T	40	1942	11	1941
2	T	50	1942	11	1941

5.3.13 Recharge-Proportions File (HARDPAN.DAT)

The inputs for the recharge-proportion file specify what proportion of deep percolation becomes ground-water recharge. These inputs are listed below. (See table 31 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1-50	5F10.0	RDEEP(*i*)	Proportion of deep percolation that becomes ground-water recharge [dimensionless].

Notes:

1. Record 1 is repeated until a value for each node in the finite-element grid has been entered. The number of nodes is specified by NN in Record 9 of the input file for subroutine *MODEL*.

Table 31. Example of inputs for recharge-proportions file (HARDPAN.DAT)

Record	Input records					Remarks
	RDEEP	RDEEP	RDEEP	RDEEP	RDEEP	
	F10.0	F10.0	F10.0	F10.0	F10.0	(5F10.0)
						The file lists a recharge proportion for each node in the model. If there is no hardpan present, the proportion is equal to 1.00.
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	0.75	1.00	1.00	1.00	0.50	
1	1.00	1.00	1.00	0.50	1.00	
1	1.00	1.00	0.50	1.00	1.00	
1	1.00	0.50	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	0.75	1.00	
1	1.00	1.00	0.50	1.00	1.00	
1	1.00	0.50	1.00	1.00	1.00	
1	0.50	1.00	1.00	1.00	0.50	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	1.00	1.00	
1	1.00	1.00	1.00	0.75	1.00	
1	1.00	1.00	0.50	1.00	1.00	
1	1.00	0.50	1.00	1.00	1.00	
1	0.50	1.00	1.00	1.00	0.50	
1	1.00	1.00	1.00	0.50	1.00	

5.3.14 Pumping-Destination File (DESTIN.DAT)

The inputs for the pumping-destination file specify that pumpage from wells within a group, indicated by the well-status code, is discharged into the canal system, rather than used directly for irrigation. These inputs are listed below. (See table 32 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1-10	I10	NPOUT	Number of groups of wells that do not pump to user.
2	1-10	I10	POUT(*i*)	Identity of well-status code for each group of wells.

Notes:

1. Record 2 is repeated for NPOUT groups of wells.

Table 32. Example of inputs for pumping-destination file (DESTIN.DAT)

Record	Input records
	NPOUT I10
1	3
	POUT I10
2	45
2	46
2	47

5.3.15 Well-Exclusion File (EXCLUDE.DAT)

The inputs for the well-exclusion file specify that pumpage from an indicated well is excluded from the simulation. These inputs are listed below. (See table 33 for a formatted example for these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1-10	I10	NEXC	Number of wells to be excluded.
2	1-12	A12	EWNUM(*i*)	Name of well to be excluded.

Notes:

1. Record 2 is repeated for NEXC well names.

Table 33. Example of inputs for well-exclusion file (EXCLUDE.DAT)

Record	Input records
	NEXC
	I10
1	3
	EWNUM
	A12
2	T15R03S15C01
2	T15R04S02Q02
2	T15R04S32H01

5.4 Subroutine *Search* Input Files

The parameter-identification files contain the description of the parameter-identification problem to be solved. The parameter-index file (MODEL.IND) contains specification for the assignment of aquifer parameters to global parameters within subroutine *SEARCH*, which is described in Section 3.16.2.2. The assignment of parameter indexes for the hydraulic conductivity of the river bed is specified by the variable INDRIV in the river-aquifer interactions file (RIVER). The parameter-search file (SEARCH.DAT) configures the parameter-identification problem that is to be solved. If a simulation does not require parameter identification, the parameter-search file can be omitted. In this case, IFIT in Record 2 of the model input (described in Section 5.2.1) must equal zero, and the parameter-index file in the file specification file (described in Section 5.1) is named "NUL."

5.4.1 Parameter-Index File (MODEL.IND)

The inputs for the parameter-index file describe the association of parameters with the hydraulic properties of elements. These inputs are listed below. (See table 34 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1-10	I10		Element number (for reference only).
	11-20	I10	INDKXX	Parameter index for the hydraulic conductivity in the *x* direction of the element.
	21-30	I10	INDKYY	Parameter index for the hydraulic conductivity in the *y* direction of the element.
	31-40	I10	INDKZZ	Parameter index for the hydraulic conductivity in the *z* direction of the element.
	41-50	I10	INDSS	Parameter index for the specific storage of the element.
	51-60	I10	INDSY	Parameter index for assignment of specific yield to element of the parameter vector.

Notes:

1. Record 1 is repeated for the number of prismatic elements (NE times), where NE is given in Record 9 of the model-input file.
2. More than one aquifer-parameter value can be assigned to a single parameter index, which allows aquifer-parameter values within a three-dimensional block of the ground-water system to be adjusted as a group by the same proportion during the parameter-identification procedure.

Table 34. Example of inputs for parameter-index file (MODEL.IND)

Record	Input records element number I10	INDKXX I10	INDKYY I10	INDKZZ I10	INDSS I10	INDSY I10
1	1	4	4	23	42	61
1	2	4	4	23	42	61
1	3	1	1	20	39	58
1	4	1	1	20	39	58
1	5	1	1	20	39	58
1	6	1	1	20	39	58
1	7	4	4	23	42	61
1	8	4	4	23	42	61
1	9	4	4	23	42	61
1	10	4	4	23	42	61
1	11	4	4	23	42	61
1	12	1	1	20	39	58
1	13	1	1	20	39	58
1	14	1	1	20	39	58
1	15	1	1	20	39	58
1	16	4	4	23	42	61
1	17	4	4	23	42	61
1	18	5	5	24	43	62
1	19	5	5	24	43	62
1	20	5	5	24	43	62
1	21	1	1	20	39	58
1	22	1	1	20	39	58
1	23	1	1	20	39	58
1	24	1	1	20	39	58
1	25	4	4	23	42	61
1	26	4	4	23	42	61
1	27	5	5	24	43	62
1	28	5	5	24	43	62
1	29	5	5	24	43	62
1	30	5	5	24	43	62
1	31	5	5	24	43	62
1	32	1	1	20	39	58
1	33	1	1	20	39	58
1	34	1	1	20	39	58
1	35	1	1	20	39	58
1	36	4	4	23	42	61
1	37	6	6	25	44	63
1	38	5	5	24	43	62
1	39	5	5	24	43	62
1	40	5	5	24	43	62
1	41	5	5	24	43	62
1	42	5	5	24	43	62
1	43	5	5	24	43	62

5.4.2 Parameter-Search File (SEARCH.DAT)

The inputs for the parameter-search file specify the parameter-identification problem to be solved. These inputs are listed below. (See table 35 for a formatted example of these inputs.)

Record	Columns	Format	Variable	Variable Description
1	1-10	I10	NGBL	Number of global parameters.
	11-20	I10	NPAR	Number of active parameters.
	21-30	I10	NITER	Number of iterations in parameter-identification process.
	31-40	F10.0	PFACT	Perturbation factor for calculating the finite-difference approximation for sensitivity matrix.
	41-50	I10	IOLS	Switch for doing an ordinary nonlinear least-squares.
2	1-10	I10		Global parameter number (for reference only).
	11-20	F10.0	PARAM0(*i*)	Parameter value for global parameter [dimensionless].
	21-30	F10.0	IPARAM(*i*)	Active parameter corresponding to global parameter.
3	11-20	F10.0	PARAM(*i*)	Initial value for active parameter [dimensionless].
	21-30	F10.0	UPAR(*i*)	Prior estimate of mean value for parameter [dimensionless].
	31-40	F10.0	VPAR(*i*)	Prior estimate of standard deviation for the estimate of mean value for parameter [dimensionless].
4	1-10	I10	NOBS	Number of water-level observations.
5	1-10	I10	STEP(*i*)	Time step in simulation that corresponds to the measured water level [dimensionless].
	11-20	I10	NODE1(*i*)	First node number.
	21-30	F10.0	WEIGHT1(*i*)	First node weighting factor.
	31-40	I10	NODE2(*i*)	Second node number.
	41-50	F10.0	WEIGHT2(*i*)	Second node weighting factor.
	51-60	I10	NODE3(*i*)	Third node number.
	61-70	F10.0	WEIGHT3(*i*)	Third node weighting factor.
	71-80	F10.0	WLM(*i*)	Water-level measurement [ft].
	81-90	F10.0	WOBS(*i*)	Standard deviation of error in water-level measurement [ft].
	93-104	A12		Well name (for reference only).

Notes:

1. Record 2 is repeated for each of the global parameters (NGBL times).
2. Record 3 is repeated for each active parameter (NPAR times).
3. Record 5 is repeated for NOBS water-level measurements.
4. The number of global parameters (NGBL) must correspond with the number of parameter indexes in the parameter index file. Accordingly, NGBL must equal the total number of aquifer-parameter block assignments made.
5. The number of active parameters (NPAR) is the number of parameters that will be adjusted during parameter-identification process. NPAR must be less than or equal to NGBL.
6. The perturbation factor PFACT is the natural log of the factor by which the aquifer parameters in subroutine *MODEL* are to be perturbed.
7. The switch IOLS equals zero if the MAP least-squares method is to be used. The switch equals 1 if the ordinary nonlinear least squares method is to be used.

8. The active parameter correspondence IPARAM equals zero, if the global parameter is not being adjusted. Otherwise, IPARAM is less than or equal to NPAR. More than one global parameter can be assigned to the same active parameter.

9. The prior estimate of the mean value for the parameter UPAR is entered in terms of the natural log of the mean.

10. The prior estimate of the standard deviation for the estimates of the mean VPAR is entered in terms of the standardization of the natural logs.

11. Each well selected for use in the calibration can be located within a stack of one or more finite elements as described in Section 5.3.1, Well-Site Inventory File. In the horizontal plane, three nodes within the stack of elements will most closely correspond to the screened interval of the well. These three nodes form a horizontal triangle around the well screen. Variables NODE1(*i*), NODE2(*i*), and NODE3(*i*) identify these three nodes for each well. The model first calculates water levels for each node. The weighting factor for each node is given by the relation

$$w_i = \frac{a_i}{b_i},$$

where

w_i is the weighting factor for the node,
a_i is the perpendicular distance from the side opposite the node *i* to the well [L], and
b_i is the perpendicular distance from the side opposite node *i* to node *i* [L].

Table 35. Example of inputs for parameter-search file (SEARCH.DAT)

Record	Input records				
	NGBL I10	NPAR I10	NITER I10	PFACT F10.0	IOLS I10
1	20	3	2	0.05	0
	global parameter number	PARAMO F10.0	IPARAM F10.0		
2	1	0	1		
2	2	0	1		
2	3	0	2		
2	4	0	2		
2	5	0	3		
2	6	0	3		
2	7	0	0		
2	8	0	0		
2	9	0	0		
2	10	0	0		
2	11	0	0		
2	12	0	0		
2	13	0	0		
2	14	0	0		
2	15	0	0		
2	16	0	0		
2	17	0	0		
2	18	0	0		
2	19	0	0		
2	20	0	0		
	active parameter number	PARAM F10.0	UPAR F10.0	VPAR F10.0	
3	1	0	0	0.0063	
3	2	0	0	0.0063	
3	3	0	0	0.0063	
	NOBS I10				
4	1318				

Table 35. Example of inputs for parameter-search file (SEARCH.DAT)--Continued

Record	Input records								
	STEP I10	NODE1 I10	WEIGHT1 F10.0	NODE2 I10	WEIGHT2 F10.0	NODE3 I10	WEIGHT3 F10.0	WLM F10.0	WOBS F10.0	well name A12
5	1	226	0.263	269	0.577	229	0.160	53.0	5.0	T16R04S06C01
5	2	226	0.263	269	0.577	229	0.160	51.2	5.0	T16R04S06C01
5	3	226	0.263	269	0.577	229	0.160	51.5	5.0	T16R04S06C01
5	4	226	0.263	269	0.577	229	0.160	51.8	5.0	T16R04S06C01
5	5	226	0.263	269	0.577	229	0.160	52.1	5.0	T16R04S06C01
5	6	226	0.263	269	0.577	229	0.160	52.3	5.0	T16R04S06C01
5	7	226	0.263	269	0.577	229	0.160	52.0	5.0	T16R04S06C01
5	8	226	0.263	269	0.577	229	0.160	52.0	5.0	T16R04S06C01
5	1	136	0.453	179	0.186	139	0.361	64.6	5.0	T17R04S23C01
5	2	136	0.453	179	0.186	139	0.361	63.5	5.0	T17R04S23C01
5	3	136	0.453	179	0.186	139	0.361	64.2	5.0	T17R04S23C01
5	4	136	0.453	179	0.186	139	0.361	65.6	5.0	T17R04S23C01
5	5	136	0.453	179	0.186	139	0.361	65.3	5.0	T17R04S23C01
5	6	136	0.453	179	0.186	139	0.361	65.8	5.0	T17R04S23C01
5	7	136	0.453	179	0.186	139	0.361	66.1	5.0	T17R04S23C01
5	8	136	0.453	179	0.186	139	0.361	66.2	5.0	T17R04S23C01

5.5 Specifying Array Dimensions for Problem Size

The dimensions of arrays in the *FEMFLOW3D* must be set to accommodate a particular problem size. The dimensions are set by editing two files, each containing a PARAMETER statement. These files are included in appropriate subroutines by INCLUDE statements, where the parameters within the PARAMETER statement are the dimensions of arrays.

The file PARAM.FOR contains the PARAMETER statement that dimensions the size of the finite-element problem to be solved. These arrays are those associated with the finite-element solution, specified-head boundaries, specified-flux boundaries, internal fluxes, stream-aquifer interactions, ground-water evapotranspiration, variable-flux boundaries, fault internal conditions, and land subsidence. However, the PARAMETER statement in file PARAM.FOR does not dimension the size of the irrigated-agricultural problem to be solved.

The parameter value in the PARAMETER statement should be set as follows:

Parameter	Definition
MAXNN	Maximum number of nodes in the finite-element grid. The value of MAXNN equals the number of nodes.
MAXNE	Maximum number of prismatic elements in the finite-element grid. The value of MAXNE equals the number of elements.
MAXNE2	Maximum number of tetrahedral elements in the finite-element grid. The number of tetrahedral elements is never larger than three times the number of prismatic elements. Therefore, the value of MAXNE2 equals three times the value of MAXNE. However, if some of the prismatic elements have zero-height edges, the number of tetrahedral elements will be smaller.
MAXNB	Maximum half-band width. The value of MAXNB is determined by the pattern by which nodes in the finite-element grid are assigned numbers. The half-band width more specifically is determined by the element in the grid that has the maximum difference between the node numbers that define the element. Therefore, the value of MAXNB equals that maximum difference plus 1.
MAXSTP	Maximum number of time steps in the simulation. The value of MAXSTP equals the number of time steps.
MAXFLS	Maximum number of file segments that are concatenated into an input file. The value of MAXFLS equals the maximum number of file segments for an input file. However, the value of MAXFLS equals 1 if no file segments are concatenated.
MAXCOL	Maximum number of element columns. The value of MAXCOL equals the maximum number of element columns specified in the input file for land subsidence. The number of columns is never larger than the number of elements in the top surface of the grid. The value of MAXCOL equals 1 if land subsidence is not simulated.
MAXNUM	Maximum number of elements in a column. The value of MAXNUM equals the maximum number of element layers specified in the input file for land subsidence. The number of elements in a column is never larger than the number of element layers in the grid. The value of MAXNUM equals 1 if land subsidence is not simulated.
MAXCHN	Maximum number of specified-head nodes. The value of MAXCHN equals the number of specified-head nodes. However, the value of MAXCHN equals 1 if specified-head nodes are not simulated.

MAXTAB	Maximum number of entries in specified-head tables. The specified-head tables define hydrographs of heads for the specified-head nodes. The entries in the tables are the points that define the hydrographs. Therefore, the value of MAXTAB equals the number of entries in the table with the largest number entries. However, the value of MAXTAB equals 1 if tables are not used.
MAXSET	Maximum number of flux data sets. The specified-flux data sets define the fluxes at nodes. To represent temporal changes in the spatial distribution of fluxes, multiple data sets can be used. Therefore, the value of MAXSET equals the number of data sets. However, the value of MAXSET equals 1 if no specified-flux nodes are simulated.
MAXRCH	Maximum number of river channels. A river system is represented by a network of tributary river channels. The value of MAXRCH equals the number of separate channels in the network. However, the value of MAXRCH equals 1 if river-aquifer interactions are not simulated.
MAXRIV	Maximum number of node reaches in a river channel. Each river channel is divided into reaches, each associated with a node in the finite-element grid. The value of MAXRIV equals the number of node reaches in the channel having the most node reaches. However, the value of MAXRIV equals 1 if river-aquifer interactions are not simulated.
MAXINF	Maximum number of inflow locations within the river system. These are locations where inflows are specified for each time step. The value of MAXINF equals the number of these inflows. However, the value of MAXINF equals zero if river-aquifer interactions are not simulated.
MAXSEC	Maximum number of tables of stream-channel geometry. The stage-discharge and width-discharge relations for a node reach can be represented by tabulations of the relations. The value of MAXSEC equals the number of these relations. However, the value of MAXSEC equals 1 if river-aquifer relations are not simulated or if tabulated relations are not used.
MAXPTS	Maximum number of points in a table of river-channel geometry. The channel-geometry tables define stage-discharge and width-discharge relations for node reaches. The entries in the table are the points that define the relations. Therefore, the value of MAXPTS equals the number of entries in the table with the largest number of entries. However, the value of MAXPTS equals 1 if river-aquifer interactions are not simulated or if tabulated relations are not used.
MAXETN	Maximum number of evapotranspiration nodes. The value of MAXETN equals the number of evapotranspiration nodes. Value of MAXETN equals 1 if ground-water evapotranspiration is not simulated.
MAXVFB	Maximum number of variable-flux nodes. The value of MAXVFB equals the number of variable-flux nodes. However, the value of MAXVFB equals 1 if variable-flux boundaries are not simulated.
MAXFLN	Maximum number of fault links. The value of MAXFLN equals the number of fault links. However, the value of MAXFLN equals 1 if fault links are not simulated.
MAXPAR	Maximum number of parameter in the parameter-identification problem. The value of MAXPAR equals the number of parameters. However, the value of MAXPAR equals 1 if parameter identification is not done.
MAXWL	Maximum number of measured water levels in the parameter-identification problem. The value of MAXWL equals the number of measured water levels. However, the value of MAXWL equals 1 if parameter identification is not done.

The file PARAM2.FOR contains the PARAMETER statement that dimensions the size of the irrigated-agriculture problem to be solved. If an irrigated-agricultural system is not simulated, the parameter values all equal 1. Otherwise, the parameter values in the PARAMETER statement should be set as follows:

MAXWEL	Maximum number of wells. The value of MAXWEL equals the number of wells.
MAXWST	Maximum number of well-status changes. The status of a well changes if the user area for the well changes or if the group category of the well changes. These changes in well status are reflected by multiple records in the well-status file. The value of MAXWST equals the maximum number of records for an individual well.
MAXWND	Maximum number of well nodes. Each well is associated with a set of nodes. The value of MAXWND equals the number of nodes associated with the well with the largest number of nodes.
MAXWSQ	Maximum number of records in the monthly pumping file. The value of MAXWSQ equals the total monthly pumpage for all wells and all months.
MAXPCE	Maximum number of well-status codes. Well-status codes are established for specified well groups. The value of MAXPCE is the number of different well-status codes.
MAXUSE	Maximum number of user areas. The value of MAXUSE equals the number of user areas.
MAXUST	Maximum number of user-area changes. The acreage within a user area can be changed with time. The value of MAXUST equals the maximum number of acreage change for an individual user area.
MAXUND	Maximum number of user-area nodes. Each user area is associated with a set of nodes, which can change with time. The value of MAXUND equals the number of nodes associated with the user area with the largest number of nodes.
MAXUSQ	Maximum number of monthly canal-delivery values. The value of MAXUSQ equals the total number of monthly delivery values for all user areas and all months.
MAXDCE	Maximum number of user-type codes. User-type codes are established for specified user-area groups. The value of MAXDCE is the number of different user-type codes.
MAXCON	Maximum number of months in construction files. The pumping-construction, delivery-construction, and crop-construction files specify pumpage, canal deliveries, crops, and weather representing a particular simulation scenario. The construction files specify these conditions for each month in the simulation. The value of MAXCON is the number of months in a construction file.
MAXCRP	Maximum number of crops. The value of MAXCRP is the number of crops.
MAXREC	Maximum number of crop inventory records per user.

6.0 REFERENCES CITED

Anderson, M.J., and Woessner, W.W., 1992, Applied groundwater modeling. Simulation of flow and advective transport: California, Academic Press, Inc., 381 p.

Beck, J.V., and Arnold, K.J., 1977, Parameter estimation in engineering and science: New York, John Wiley, 501 p.

Bredehoeft, J.D., and Pinder, G.F., 1970, Digital analysis of the areal flow in multi-aquifer ground-water systems. A quasi three-dimensional model: Water Resources Research, v. 6, no. 3, p. 883-888.

Carslaw, H.S., and Jaeger, J.C., 1959, Conduction of heat in solids: London, Oxford University Press, 510 p.

Darcy, H., 1856, Les fontaines publiques de la villa de Dijon: Victor balmont, Paris.

Durbin, T.J., 1978, Calibration of a mathematical model of the Antelope Valley ground-water basin, California: U.S. Geological Survey Water-Supply Paper 2046, 51 p.

Durbin, T.J., and Berenbrock, C., 1985, Three-dimensional simulation of free-surface aquifers by the finite-element method: U.S. Geological Survey Water-Supply Paper 2270, p. 51-67.

Freeze, R.A., 1971, Three-dimensional transient, saturated-unsaturated flow in a ground-water basin: Water Resources Research, v. 7, no. 2, p. 347-366.

Frind, E.O., and Verge, M.J., 1978, Three-dimensional modeling of ground-water flow systems: Water Resources Research, v. 14, no. 5, p. 844-856.

Helm, D.C., 1975, One-dimensional simulation of aquifer-system compaction near Pixley, California: Water Resources Research, v. 11, no. 3, p. 465-478.

-----1984, Latrobe Valley subsidence predictions: The modeling of time-dependent ground movement due to groundwater withdrawal: State Electricity Commission of Victoria, Australia, Fuel Department and Design Engineering and Environment Department, 2 volumes, variously paged.

Jensen, M.E., ed., 1983, Design and operation of farm irrigation systems: American Society of Agricultural Engineers Monograph 3, 425 p.

Leake, S.A., 1977, Simulation of flow from an aquifer to a partially penetrating trench: Journal of Research, v. 5, no. 5, p. 535-540.

Leake, S.A., and Prudic, D.E., 1988, Documentation of a computer program to simulate aquifer-system compaction using the modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 88-482, 80 p.

Leopold, L.B., Wolman, M.G., and Miller, J.P., 1964, Fluvial processes in geomorphology: California, W.H. Freeman and Company, 522 p.

Lohman, S.W., 1979, Ground-water hydraulics: U.S. Geological Survey Professional Paper 708, 70 p.

Neuman, S.P., 1974, Effect of partial penetration on flow in unconfined aquifers considering delayed gravity response: Water Resources Research, v. 10, no. 3, p. 303-312.

Neuman, S.P., and Witherspoon, P.A., 1971, Analysis of nonsteady flow with a free surface using the finite-element method: Water Resources Research, v. 7, no. 3, p. 611-623.

Pinder, G.F., and Gray, W.G., 1977, Finite-element simulation in surface and subsurface hydrology: New York, Academic Press, 396 p.

Riley, F.S., 1969, Analysis of borehole extensometer data from central California, *in* Tison, L.J., ed., Land subsidence, v. 2: International Association of Scientific Hydrology Publication 89, p. 423-431.

Terzaghi, K., 1925, Erdbaumechanik auf Bodenphysikalischer Grundlage: Wein Leipzig, Deuticke, 399 p.

Theis, C.V., 1935, The relation between the lowering of the piezometric surface and the rate and duration of discharge of a well using groundwater storage: Transactions American Geophysical Union, v. 2, p. 519-524.

Trescott, P.C., 1975, Documentation of finite-difference model for simulation of three-dimensional ground-water flow: U.S. Geological Survey Open-File Report 75-438, 32 p.

Trescott, P.C., and Larson, S.P., 1977, Solution to three-dimensional ground-water flow equations using the strongly implicit procedure: Journal of Hydrology, v. 35, p. 49-60.

Zienkiewicz, O.C., 1977, The finite-element method (3d ed.): New York, McGraw-Hill, 787 p.

—PAGE 195 FOLLOWS—

APPENDIX A

FORTRAN SOURCE CODE

```
C     --------------
      PROGRAM FEMFLOW
C     --------------
C     NOTE: COMPILER SPECIFICATIONS REQUIRED
C     (1) COMPILER MUST ACCEPT "ENTRY" STATEMENTS THAT HAVE A DIFFERENT
C         ARGUMENT LIST THAN THE ARGUMENT LIST FOR THE MAIN SUBROUTINE
C         CALL.
C     (2) INTERNAL VARIABLES WITHIN THE SUBROUTINES MUST BE SAVED.
C     (3) IT IS ASSUMED THAT ALL OF THE ARRAYS ARE INITIALIZED TO ZERO
C         BY THE COMPILER.
C
      INCLUDE 'PARAM.FOR'
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*2 (I-N)
      REAL*8 PARAMX(MAXPAR)
      PARAMX(1)=0.0
      CALL MODEL1(IFIT)
      IF (IFIT.EQ.0) THEN
      ICODE=-99
      IOUT=1
      CALL MODEL3(PARAMX,ICODE,IOUT,IFIT)
      STOP
      ENDIF
      CALL SEARCH(IFIT)
      STOP
      END
      INCLUDE 'SEARCH.FOR'
      INCLUDE 'SOLEQU.FOR'
      INCLUDE 'MODEL.FOR'
      INCLUDE 'EVAP.FOR'
      INCLUDE 'RIVER.FOR'
      INCLUDE 'CHEAD.FOR'
      INCLUDE 'FLUX.FOR'
      INCLUDE 'VFLUX.FOR'
      INCLUDE 'BAND.FOR'
      INCLUDE 'SOLVE.FOR'
      INCLUDE 'BUDGET.FOR'
      INCLUDE 'SHAPE.FOR'
      INCLUDE 'WATER.FOR'
      INCLUDE 'LAND.FOR'
      INCLUDE 'FAULT.FOR'
      INCLUDE 'SINK.FOR'
      INCLUDE 'FMERGE.FOR'
```

```
C     ----------------------
      SUBROUTINE SEARCH(IFIT)
C     ----------------------
C
      INCLUDE 'PARAM.FOR'
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*2 (I-N)
      REAL*8 PARAM0(MAXPAR),PARAMX(MAXPAR),PARAM(MAXPAR),
     1   UPAR(MAXPAR),VPAR(MAXPAR),WLM(MAXWL),WLC(MAXWL),
     2   LWLC(MAXWL),DELP(MAXPAR),WOBS(MAXWL),WEIGHT(MAXWL,3)
      REAL*8 LH(MAXPAR,MAXPAR),RH(MAXPAR),X(MAXWL,MAXPAR)
      INTEGER*2 IPARAM(MAXPAR),STEP(MAXWL),NODE(MAXWL,3)
      CHARACTER*60 FILE1,FILE2,XFILE(MAXFLS)
C
C     OPEN FILES
C
      IR0=10
      IR1=50
      IW1=51
      READ(IR0,9190) FILE1,NFILE
      IF (NFILE.GT.0) THEN
      READ(IR0,9191) (XFILE(I),I=1,NFILE)
      CALL FMERGE(XFILE,FILE1,NFILE)
      ENDIF
      READ(IR0,9191) FILE2
      OPEN (UNIT=IR1,FILE=FILE1,STATUS='OLD')
      OPEN (UNIT=IW1,FILE=FILE2,STATUS='UNKNOWN')
 9190 FORMAT(A60,I10)
 9191 FORMAT(A60)
C
C     READ PARAMETER SEARCH SPECIFICATIONS
C
      WRITE(*,'(1X,''READING IN SEARCH '',A60)') FILE1
      READ(IR1,900) NGBL,NPAR,NITER,PFACT,IOLS
      WRITE(IW1,902) NGBL,NPAR,NITER,PFACT,IOLS
      READ(IR1,904) (PARAM0(IPAR),IPARAM(IPAR),IPAR=1,NGBL)
      WRITE(IW1,906)
      WRITE(IW1,908) (IPAR,PARAM0(IPAR),IPARAM(IPAR),IPAR=1,NGBL)
      READ(IR1,910) (PARAM(IPAR),UPAR(IPAR),VPAR(IPAR),IPAR=1,NPAR)
      WRITE(IW1,912)
      DO 329 IPAR=1,NPAR
      IF (IOLS.NE.1) THEN
      WRITE(IW1,914) IPAR,PARAM(IPAR),UPAR(IPAR),VPAR(IPAR)
      ELSE
      WRITE(IW1,9141) IPAR,PARAM(IPAR),UPAR(IPAR),-999
      ENDIF
 329  CONTINUE
      READ(IR1,916) NOBS
      READ(IR1,920) (STEP(IOBS),(NODE(IOBS,I),WEIGHT(IOBS,I),I=1,3),
     1   WLM(IOBS),WOBS(IOBS),IOBS=1,NOBS)
      WRITE(IW1,918) NOBS
      DO 330 IOBS=1,NOBS
      IF (IOLS.NE.1) THEN
      WRITE(IW1,922) STEP(IOBS),(NODE(IOBS,I),WEIGHT(IOBS,I),I=1,3),
     1   WLM(IOBS),WOBS(IOBS)
      ELSE
      WRITE(IW1,9221) STEP(IOBS),(NODE(IOBS,I),WEIGHT(IOBS,I),I=1,3),
     1   WLM(IOBS),-999
      ENDIF
 330  CONTINUE
```

```
C
 900  FORMAT(3I10,F10.0,I10)
 902  FORMAT(1X,'SEARCH SPECIFICATION'/1X,20('-')/
     1    1X,'NUMBER OF GLOBAL PARAMETERS        ',I10/
     2    1X,'NUMBER OF ACTIVE PARAMETERS        ',I10/
     3    1X,'NUMBER OF ITERATIONS               ',I10/
     4    1X,'PERTURBATION FACTOR                ',1PE10.2/
     5    1X,'SWITCH FOR ORDINARY LEAST SQUARES',I10)
 904  FORMAT(10X,F10.0,I10)
 906  FORMAT(/1X,'GLOBAL PARAMETERS'/1X,17('-')//
     1    1X,4X,'GLOBAL',5X,'VALUE',4X,'ACTIVE'/
     2    1X,1X,'PARAMETER',11X,'PARAMETER'/)
 908  FORMAT(1X,I10,1PE10.2,I10)
 910  FORMAT(10X,3F10.0)
 912  FORMAT(/1X,'ACTIVE PARAMETERS'/1X,17('-')//
     1    1X,1X,'PARAMETER',3X,'INITIAL',2X,'EXPECTED',4X,'UNCER-'/
     2    1X,15X,'VALUE',5X,'VALUE',4X,'TAINTY'/)
 914  FORMAT(1X,I10,1PE10.2,1PE10.2,1PE10.2)
 9141 FORMAT(1X,I10,1PE10.2,1PE10.2,I10)
 916  FORMAT(I10)
 918  FORMAT(/1X,'MEASURED WATER LEVELS'/1X,22('-')/
     1    1X,'NUMBER OF OBSERVATIONS',I10//
     2    1X,6X,'STEP',3(6X,'NODE',4X,'WEIGHT'),5X,'LEVEL',4X,'UNCER-'/
     3    1X,84X,'TAINTY'/)
 920  FORMAT((I10,3(I10,F10.0),2F10.0))
 922  FORMAT(1X,I10,3(I10,F10.3),F10.2,F10.2)
 9221 FORMAT(1X,I10,3(I10,F10.3),F10.2,I10)
C
C     INVERT UNCERTAINTY VECTORS
C
      DO 130 I=1,NOBS
      IF (IOLS.NE.1) THEN
      WOBS(I)=1.0/(WOBS(I)**2)
      ELSE
      WOBS(I)=1.0
      ENDIF
 130  CONTINUE
      DO 132 I=1,NPAR
      IF (IOLS.NE.1) THEN
      VPAR(I)=1.0/(VPAR(I)**2)
      ELSE
      VPAR(I)=0.0
      ENDIF
 132  CONTINUE
C
C     BEGIN ITERATION LOOP
C
      IOUT=0
      ITER=0
 400  ITER=ITER+1
C
C     CONSTRUCT SENSITIVITY MATRIX
C     WATER LEVELS FOR UNPERTURBED PARAMETERS
C
      DO 140 IGBL=1,NGBL
      PARAMX(IGBL)=PARAM0(IGBL)
      II=IPARAM(IGBL)
      IF (II.NE.0) PARAMX(IGBL)=PARAMX(IGBL)+PARAM(II)
```

```
 140  CONTINUE
      IF (ITER.GT.NITER) GO TO 410
      ICODE=1
      CALL MODEL2(PARAMX,STEP,NODE,WEIGHT,LWLC,NOBS,ICODE,IOUT,IFIT)
C
C     SUM OF SQUARED RESIDUALS S0
C
      S0=0.0
      DO 192 J=1,NOBS
      S0=S0+(WLM(J)-LWLC(J))*WOBS(J)*(WLM(J)-LWLC(J))
 192  CONTINUE
      DO 194 J=1,NPAR
      S0=S0+(PARAM(J)-UPAR(J))*VPAR(J)*(PARAM(J)-UPAR(J))
 194  CONTINUE
      IF (ITER.EQ.1) WRITE(IW1,995) S0
C
C     WATER LEVELS FOR PERTURBED PARAMETERS
C
      DO 144 IPAR=1,NPAR
      DO 141 IGBL=1,NGBL
      PARAMX(IGBL)=PARAM0(IGBL)
      II=IPARAM(IGBL)
      IF (II.NE.0) THEN
      IF (II.NE.IPAR) THEN
      PARAMX(IGBL)=PARAMX(IGBL)+PARAM(II)
      ELSE
      PARAMX(IGBL)=PARAMX(IGBL)+PARAM(II)+DLOG(1.0+PFACT)
      ENDIF
      ENDIF
 141  CONTINUE
      ICODE=2
      CALL MODEL2(PARAMX,STEP,NODE,WEIGHT,WLC,NOBS,ICODE,IOUT,IFIT)
C
C     CONSTRUCT MATRIX
C
      DO 142 IOBS=1,NOBS
      X(IOBS,IPAR)=(WLC(IOBS)-LWLC(IOBS))/DLOG(1.0+PFACT)
 142  CONTINUE
 144  CONTINUE
C
C     CONSTRUCT LEFT-HAND SIDE
C
      DO 150 I=1,NPAR
      DO 148 L=1,NPAR
      LH(I,L)=0.0
      DO 145 J=1,NOBS
      LH(I,L)=LH(I,L)+X(J,I)*WOBS(J)*X(J,L)
 145  CONTINUE
 148  CONTINUE
      LH(I,I)=LH(I,I)+VPAR(I)
 150  CONTINUE
C
C     CONSTRUCT RIGH-HAND SIDE
C
      DO 154 I=1,NPAR
      RH(I)=0.0
      DO 151 J=1,NOBS
      RH(I)=RH(I)+X(J,I)*WOBS(J)*(WLM(J)-LWLC(J))
```

```
 151   CONTINUE
       RH(I)=RH(I)+VPAR(I)*(UPAR(I)-PARAM(I))
 154   CONTINUE
C
C      SOLVE FOR PARAMETER CHANGES
C
       CALL SOLEQU(LH,RH,DELP,NPAR)
C
C      ADJUSTMENTS TO STEP LENGTH
C      SUM OF SQUARED RESIDUALS S1
C
       ALPHA=1.0
       ICHG=0
 1601  CONTINUE
       ICHG=ICHG+1
       DO 191 IGBL=1,NGBL
       PARAMX(IGBL)=PARAM0(IGBL)
       II=IPARAM(IGBL)
       IF (II.NE.0) THEN
       PARAMX(IGBL)=PARAMX(IGBL)+(PARAM(II)+ALPHA*DELP(II))
       ENDIF
 191   CONTINUE
       ICODE=3
       CALL MODEL2(PARAMX,STEP,NODE,WEIGHT,WLC,NOBS,ICODE,IOUT,IFIT)
       S1=0.0
       DO 162 J=1,NOBS
       S1=S1+(WLM(J)-WLC(J))*WOBS(J)*(WLM(J)-WLC(J))
 162   CONTINUE
       DO 164 J=1,NPAR
       S1=S1+(PARAM(J)+ALPHA*DELP(J)-UPAR(J))*VPAR(J)*
      1    (PARAM(J)+ALPHA*DELP(J)-UPAR(J))
 164   CONTINUE
       IF (S0.LT.S1) THEN
       ALPHA=ALPHA*0.5
       IF (ALPHA.GE.0.01) GO TO 1601
       WRITE(IW1,996)
       GO TO 410
       ENDIF
       G=0.0
       DO 158 IPAR=1,NPAR
       G=G+DELP(IPAR)*RH(IPAR)
 158   CONTINUE
       A=1.1
       TEST=S0-ALPHA*G*(2.0-1.0/A)
       IF (S1.GE.TEST) THEN
       H=ALPHA*ALPHA*G/(S1-S0+2.0*ALPHA*G)
       ELSE
       H=ALPHA*A
       ENDIF
       DO 168 IPAR=1,NPAR
       PARAM(IPAR)=PARAM(IPAR)+H*DELP(IPAR)
 168   CONTINUE
C
C      OUTPUT RESULTS FOR ITERATION
C
       DO 170 IGBL=1,NGBL
       PARAMX(IGBL)=PARAM0(IGBL)
       II=IPARAM(IGBL)
       IF (II.NE.0) PARAMX(IGBL)=PARAMX(IGBL)+PARAM(II)
```

```
 170   CONTINUE
       ICODE=4
       CALL MODEL2(PARAMX,STEP,NODE,WEIGHT,LWLC,NOBS,ICODE,IOUT,IFIT)
C
C      SUM OF SQUARED RESIDUALS S0
C
       S2=0.0
       DO 1921 J=1,NOBS
       S2=S2+(WLM(J)-LWLC(J))*WOBS(J)*(WLM(J)-LWLC(J))
 1921  CONTINUE
       DO 1941 J=1,NPAR
       S2=S2+(PARAM(J)-UPAR(J))*VPAR(J)*(PARAM(J)-UPAR(J))
 1941  CONTINUE
       WRITE(IW1,950) ITER,ICHG,H,S2
       WRITE(IW1,952) (IPAR,PARAM(IPAR),H*DELP(IPAR),IPAR=1,NPAR)
 950   FORMAT(/1X,'SEARCH RESULTS'/1X,14('-')/
     1    1X,'ITERATION                       ',I10/
     2    1X,'NUMBER OF ADJUSTMENTS           ',I10/
     3    1X,'STEP ADJUSTMENT                 ',1PE10.2/
     4    1X,'SUM OF SQUARED RESIDUALS        ',1PE10.2//
     5    1X,1X,'PARAMETER',5X,'VALUE',4X,'CHANGE'/)
 952   FORMAT(1X,I10,1PE10.2,1PE10.2)
 995   FORMAT(/1X,'SUM OF SQUARED RESIDUALS'/
     1    1X'FOR INITIAL VALUES               ',1PE10.2)
 996   FORMAT(/1X,'CALIBRATION TERMINATED - NEGLIGIBLE'/
     1    1X,'IMPROVEMENT IN REDUCING SUM OF SQUARES')
C
C      END ITERATION LOOP
C
       GO TO 400
 410   CONTINUE
C
C      OUTPUT SEARCH RESULTS
C
       IOUT=1
       ICODE=5
       CALL MODEL2(PARAMX,STEP,NODE,WEIGHT,WLC,NOBS,ICODE,IOUT,IFIT)
       WRITE(IW1,944)
       SEPS=0.0
       SSEPS=0.0
       DO 420 IOBS=1,NOBS
       EPS=WLM(IOBS)-WLC(IOBS)
       SEPS=SEPS+EPS
       SSEPS=SSEPS+EPS**2
       WRITE(IW1,946) STEP(IOBS),(NODE(IOBS,I),WEIGHT(IOBS,I),I=1,3),
     1    WLM(IOBS),WLC(IOBS),EPS
 420   CONTINUE
       SSEPS=SQRT(SSEPS/NOBS-SEPS**2/NOBS**2)
       SEPS=SEPS/NOBS
       WRITE(IW1,948) SEPS,SSEPS
C
 944   FORMAT(/1X,'FINAL WATER LEVELS'/1X,18('-')//
     1    1X,6X,'STEP',3(6X,'NODE',4X,'WEIGHT'),2X,'MEASURED',
     2    2X,'COMPUTED',2X,'RESIDUAL'/)
 946   FORMAT(1X,I10,3(I10,F10.3),3F10.2)
 948   FORMAT(/1X,'MEAN OF RESIDUALS                   ',1PE10.2/
     1  1X,'STANDARD DEVIATION OF RESIDUALS',1PE10.2)
       STOP
       END
```

```
C     --------------------------
      SUBROUTINE SOLEQU(A,B,X,N)
C     --------------------------
C
      INCLUDE 'PARAM.FOR'
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*2 (I-N)
      REAL*8 A(MAXPAR,MAXPAR),B(MAXPAR),X(MAXPAR)
C
C     THIS SUBROUTINE SOLVES A SET OF SYMMETRIC LINEAR EQUATIONS
C
      DO 122 I=1,N
      X(I)=B(I)
 122  CONTINUE
      NM=N-1
      DO 85 I=1,NM
      IP=I+1
      DO 25 J=IP,N
      IF (ABS(A(I,I)).LT.1.E-8) GO TO 200
      A(I,J)=A(I,J)/A(I,I)
 25   CONTINUE
      DO 50 J=IP,N
      DO 45 K=J,N
      IF (A(I,J)) 35,50,35
 35   A(K,J)=A(K,J)-A(K,I)*A(I,J)
      A(J,K)=A(K,J)
 45   CONTINUE
 50   CONTINUE
      X(I)=X(I)/A(I,I)
      DO 75 K=IP,N
      X(K)=X(K)-A(K,I)*X(I)
 75   CONTINUE
 85   CONTINUE
      IF (ABS(A(N,N)).LT.1.E-8) GO TO 202
      X(N)=X(N)/A(N,N)
 100  DO 125 II=1,NM
      I=N-II+1
      DO 125 K=I,N
      X(I-1)=X(I-1)-A(I-1,K)*X(K)
 125  CONTINUE
      GO TO 130
 200  WRITE(*,1)
      WRITE(*,300) I, A(I,I)
 300  FORMAT(//,40X,I5,E20.6)
      GO TO 130
 202  WRITE(*,1)
      WRITE(*,300) N, A(N,N)
 130  RETURN
 1    FORMAT(//,8X,'BAD INVERSE, ZERO ON DIAGONAL')
      END
```

```
C     ----------------------
      SUBROUTINE MODEL1(IFIT)
C     ----------------------
C
      INCLUDE 'PARAM.FOR'
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*2 (I-N)
      CHARACTER*78 TITLE
      REAL*8 X(MAXNN),Y(MAXNN),Z(MAXNN),H0(MAXNN),
     1   KXX0(MAXNE),KYY0(MAXNE),KZZ0(MAXNE),SS0(MAXNE),SY0(MAXNE)
      INTEGER*2 IPRISM(MAXNE,6),ITOP(MAXNE),IN(MAXNE2,4)
      INTEGER*2 INDKXX(MAXNE),INDKYY(MAXNE),INDKZZ(MAXNE),INDSS(MAXNE),
     1   INDSY(MAXNE),ICMO1(MAXSTP),ICMO2(MAXSTP)
      REAL*8 WLC(MAXWL),H(MAXNN),HL(MAXNN),A(MAXNN,MAXNB),
     1   B(MAXNN,MAXNB),F(MAXNN),CCHRH(MAXNN),CCHLH(MAXNN),
     2   CRRH(MAXNN),CRLH(MAXNN),CETRH(MAXNN),CETLH(MAXNN),
     3   HLIT(MAXNN),KXX(MAXNE2),KYY(MAXNE2),KZZ(MAXNE2),
     4   SS(MAXNE2),SY(MAXNE2),QFLUX(MAXNN),QPUMP(MAXSTP,MAXNN),
     5   QRECH(MAXSTP,MAXNN),CVFLH(MAXNN),CVFRH(MAXNN),QVPAST(MAXNN),
     6   DELT(MAXSTP),SUMP(MAXSTP),SUMR(MAXSTP),C(MAXNN,MAXNB),
     7   D(MAXNN,MAXNB),HP(MAXNN),POR2(MAXNE2),WEIGHT(MAXWL,3)
      REAL*8 XE(4),YE(4),ZE(4),AE(4,4),BE(4,4),CE(4,4),DE(4,4)
      REAL*8 LHSM(MAXNN,MAXNB),RHSV(MAXNN)
      REAL*8 QPOS(MAXNN),QNEG(MAXNN)
      INTEGER*2 STEP(MAXWL),NODE(MAXWL,3),ITET3(3,4),ITET2(2,4),
     1   ITET1(4),IEL(MAXNE2),ICOL(MAXNN,MAXNB),NCOL(MAXNN),
     2   IDAG(MAXNN)
      REAL*8 PARAMX(MAXPAR)
      CHARACTER*11 CLOCK
      CHARACTER*60 FILE0,FILE20,FILE1,FILE2,FILE3,FILE4,FILE5,FILE6,
     1   XFILE(MAXFLS)
C
      DATA INDKXX,INDKYY,INDKZZ,INDSS,INDSY/MAXNE*1,MAXNE*1,MAXNE*1,
     1   MAXNE*1,MAXNE*1/
C
C     INPUT UNITS
C        MODEL FILES [FILE.DAT] (UNIT=IR0, FILE=FILE0)
C        MODEL INPUT [MODEL.DAT] (UNIT=IR1, FILE=FILE1)
C        MODEL INDEXES [MODEL.IND] (UNIT=IR2, FILE=FILE5)
C
C     OUTPUT UNITS
C        MODEL RIVER [MODEL.RIV] (UNIT=IW20, FILE=FILE20)
C        MODEL OUTPUT [MODEL.OUT] (UNIT=IW1, FILE=FILE2)
C        MODEL PLOT [MODEL.PLT] (UNIT=IW2, FILE=FILE3)
C        MODEL BUDGET [MODEL.BUD] (UNIT=IW3, FILE=FILE4)
C        MODEL FLUX [MODEL.FLX] (UNIT=IW4, FILE=FILE6)
C
      IR0=10
      IR1=11
      IR2=12
      IW1=13
      IW2=14
      IW3=15
      IW4=16
      IW20=20
C
C     READ NAMES OF MODEL INPUT AND OUTPUT FILES
```

```
C
      WRITE(*,'(1X,''ENTER FILE OF INPUT AND OUTPUT FILES: '')')
      READ(*,'(A60)') FILE0
      OPEN (UNIT=IR0,FILE=FILE0,STATUS='OLD')
C
C     MODEL INPUT FILE
C
      READ(IR0,9190) FILE1,NFILE
      IF (NFILE.GT.0) THEN
      READ(IR0,9191) (XFILE(I),I=1,NFILE)
      CALL FMERGE(XFILE,FILE1,NFILE)
      ENDIF
C
C     MODEL RIVER, OUTPUT, PLOT, BUDGET, AND FLUX FILES
C
      READ(IR0,9191) FILE20
      READ(IR0,9191) FILE2
      READ(IR0,9191) FILE3
      READ(IR0,9191) FILE4
      READ(IR0,9191) FILE6
C
C     MODEL INDEX FILE
C
      READ(IR0,9190) FILE5,NFILE
      IF (NFILE.GT.0) THEN
      READ(IR0,9191) (XFILE(I),I=1,NFILE)
      CALL FMERGE(XFILE,FILE5,NFILE)
      ENDIF
C
C     OPEN MODEL INPUT AND OUTPUT FILES
C
      OPEN (UNIT=IR1,FILE=FILE1,STATUS='OLD')
      OPEN (UNIT=IW20,FILE=FILE20,STATUS='UNKNOWN')
      OPEN (UNIT=IW1,FILE=FILE2,STATUS='UNKNOWN')
      OPEN (UNIT=IW2,FILE=FILE3,STATUS='UNKNOWN')
      OPEN (UNIT=IW3,FILE=FILE4,STATUS='UNKNOWN')
      OPEN (UNIT=IW4,FILE=FILE6,STATUS='UNKNOWN')
      OPEN (UNIT=IR2,FILE=FILE5,STATUS='OLD')
C
 9190 FORMAT(A60,I10)
 9191 FORMAT(A60)
C
C     BEGIN DATA INPUT
C     BASIN NAME
C
      READ(IR1,900) TITLE
      WRITE(IW1,901) TITLE
 900  FORMAT(A78)
 901  FORMAT(1X,A78/1X,78('-')/)
C
C     TIME STEP (DAYS),NUMBER OF TIME STEPS,NUMBER OF RIVER ITERATIONS
C
      READ(IR1,902) MAXKNS,IDELT,NITER,CLOSE,ISKIP,IFIT,ISS,ISOLVE
      WRITE(IW1,903) MAXKNS,IDELT,NITER,CLOSE,ISKIP,IFIT,ISS,ISOLVE
```

```
C
      IF (IDELT.EQ.1) THEN
      READ(IR1,9023) NSTART
      WRITE(IW1,9024) NSTART
      DO 140 I=1,NSTART
      READ(IR1,9022) KNS1,KNS2,DELT0,FDELT
      DO 139 KNS=KNS1,KNS2
      DELT(KNS)=DELT0*(FDELT**(KNS-KNS1))
      WRITE(IW1,9026) KNS,KNS1,KNS2,DELT(KNS),FDELT
      DELT(KNS)=DELT(KNS)*24.0*3600.0
 139  CONTINUE
 140  CONTINUE
      ELSE
      READ(IR1,9037) DELT0
      WRITE(IW1,973) DELT0
      DO 138 KNS=1,MAXKNS
      READ(IR1,974) ICMO1(KNS),ICMO2(KNS)
      DELT(KNS)=(ICMO2(KNS)-ICMO1(KNS)+1)*DELT0
      WRITE(IW1,975) KNS,ICMO1(KNS),ICMO2(KNS),DELT(KNS)
      DELT(KNS)=DELT(KNS)*24.0*3600.0
 138  CONTINUE
      ENDIF
 902  FORMAT(3I10,F10.0,4I10)
 9023 FORMAT(I10)
 9024 FORMAT(/1X,'TIME-STEP CONFIGURATION'/1X,23('-')//
     1   1X,'NUMBER OF RESTARTS  ',I10//
     2   1X,6X,'STEP',6X,'KNS1',6X,'KNS2',6X,'DELT',5X,'FDELT'/)
 9022 FORMAT(2I10,2F10.0)
 9026 FORMAT((1X,3I10,2(1PE10.2)))
 903  FORMAT(1X,'SIMULATION PARAMETERS'/1X,21('-')/
     3   1X,'NUMBER OF TIME STEPS                        ',I10/
     2   1X,'TIME-STEP SCHEME                            ',I10/
     4   1X,'NUMBER OF ITERATIONS                        ',I10/
     5   1X,'CLOSURE CRITERION FOR ITERATIONS            ',1PE10.2/
     6   1X,'SWITCH TO SKIP INTEGRATION UPDATE           ',I10/
     7   1X,'SWITCH TO FIT MODEL TO DATA                 ',I10/
     8   1X,'SWITCH FOR STEADY STATE                     ',I10/
     9   1X,'SWITCH FOR SOLUTION METHOD                  ',I10/)
 9037 FORMAT(F10.0)
 973  FORMAT(/1X,'TIME-STEP CONFIGURATION'/1X,23('-')/
     1   1X,'BASIC TIME STEP      ',1PE10.2//
     2   1X,6X,'STEP',5X,'ICMO1',5X,'ICMO2',6X,'DELT'/)
 974  FORMAT(10X,2I10)
 975  FORMAT(1X,3I10,1PE10.2)
C
C     OUTPUT CONTROLS
C
      READ(IR1,9025) NPRINT,NPLOT,NFLUX
      WRITE(IW1,9036) NPRINT,NPLOT,NFLUX
 9025 FORMAT(3I10)
 9036 FORMAT(/1X,'OUTPUT INTERVALS'/1X,16('-')/
     1   1X,'PRINT INTERVAL                              ',I10/
     2   1X,'PLOT INTERVAL                               ',I10/
     3   1X,'FLUX INTERVAL                               ',I10)
C
C     PARAMETERS FOR ITERATIVE SOLVER
C
      IF (ISOLVE.EQ.2) CALL SOLVE1
```

```
C
C     NUMBER OF NODES AND ELEMENTS
C
      READ(IR1,904) NN,NE
      WRITE(IW1,905) NN,NE
 904  FORMAT (2I10)
 905  FORMAT(/1X,'GRID SIZE'/1X,9('-')/
     1   1X,'NUMBER OF NODES                              ',I10/
     1   1X,'NUMBER OF ELEMENTS                           ',I10)
C
C     NODE COORDINATES (FEET)
C
      WRITE(*,'(1X,''READING COORDINATES IN MODEL'')')
      READ(IR1,906) FACX,FACY,FACZ,IECHO1
      READ(IR1,907) (X(I),Y(I),Z(I),I=1,NN)
      DO 100 I=1,NN
      X(I)=X(I)*FACX
      Y(I)=Y(I)*FACY
      Z(I)=Z(I)*FACZ
 100  CONTINUE
      WRITE(IW1,908) FACX,FACY,FACZ,IECHO1
      IF (IECHO1.NE.0) THEN
      WRITE(IW1,9081)
      WRITE (IW1,909) (I,X(I),Y(I),Z(I),I=1,NN)
      ENDIF
 906  FORMAT(3F10.0,I10)
 907  FORMAT(10X,3E10.0)
 908  FORMAT(/1X,'NODE COORDINATES'/1X,16('-')/
     1   1X,'FACTOR FOR X                                 ',1PE10.2/
     2   1X,'FACTOR FOR Y                                 ',1PE10.2/
     3   1X,'FACTOR FOR Z                                 ',1PE10.2/
     4   1X,'SWITCH FOR ECHO OF COORDINATES               ',I10)
 9081 FORMAT(/1X,6X,'NODE',9X,'X',9X,'Y',9X,'Z'/)
 909  FORMAT((1X,I10,F10.2,F10.2,F10.2))
C
C     ELEMENT INCIDENCES (COUNTER CLOCKWISE TOP THEN BOTTOM)
C
      WRITE(*,'(1X,''READING ELEMENTS IN MODEL'')')
      READ(IR1,9101) IECHO2,IECHO3
      READ(IR1,910) ((IPRISM(L,J),J=1,6),L=1,NE)
      WRITE(IW1,911) IECHO2,IECHO3
      IF (IECHO2.NE.0) THEN
      WRITE(IW1,9111)
      WRITE(IW1,912) (I,(IPRISM(I,J),J=1,6),I=1,NE)
      ENDIF
 910  FORMAT(10X,6I10)
 9101 FORMAT(2I10)
 911  FORMAT(/1X,'TRIANGULAR-PRISM ELEMENT INCIDENCES'/1X,34('-')/
     1   1X,'SWITCH FOR ECHO OF PRISMATIC NODES       ',I10/
     2   1X,'SWITCH FOR ECHO OF TETRAHEDRAL NODES     ',I10)
 9111 FORMAT(/1X,6X,'ELEM',5X,'NODES'/)
 912  FORMAT(1X,7I10)
C
C     DEFINE SUBDIVISION OF BRICK INTO TETRAHEDRONS
```

```
C
      ITET3(1,1)=1
      ITET3(1,2)=2
      ITET3(1,3)=6
      ITET3(1,4)=3
      ITET3(2,1)=5
      ITET3(2,2)=6
      ITET3(2,3)=1
      ITET3(2,4)=2
      ITET3(3,1)=4
      ITET3(3,2)=5
      ITET3(3,3)=6
      ITET3(3,4)=1
      ITET2(1,1)=1
      ITET2(1,2)=5
      ITET2(1,3)=3
      ITET2(1,4)=2
      ITET2(2,1)=4
      ITET2(2,2)=5
      ITET2(2,3)=3
      ITET2(2,4)=1
      ITET1(1)=4
      ITET1(2)=2
      ITET1(3)=3
      ITET1(4)=1
C
C     GENERATE INCIDENCES FOR TETRAHEDRAL ELEMENTS
C         IN EACH TRIANGULAR PRISM
C
      L=0
      DO 1020 LB=1,NE
C
C     DETERMINE NUMBER OF TETRAHEDRONS IN TRIANGULAR PRISM
C
      IF (IPRISM(LB,6).NE.0) GO TO 1031
      IF (IPRISM(LB,6).EQ.0.AND.IPRISM(LB,5).NE.0) GO TO 1041
      IF (IPRISM(LB,6).EQ.0.AND.IPRISM(LB,5).EQ.0) GO TO 1051
C
C     THREE TETRAHEDRONS
C
 1031 CONTINUE
      DO 1033 LT=1,3
      L=L+1
      IEL(L)=LB
      DO 1032 I=1,4
      II=ITET3(LT,I)
      IN(L,I)=IPRISM(LB,II)
 1032 CONTINUE
 1033 CONTINUE
      GO TO 1020
C
C     TWO TETRAHEDRONS
C
 1041 CONTINUE
      DO 1043 LT=1,2
      L=L+1
      IEL(L)=LB
      DO 1042 I=1,4
      II=ITET2(LT,I)
      IN(L,I)=IPRISM(LB,II)
```

```
 1042 CONTINUE
 1043 CONTINUE
      GO TO 1020
C
C     ONE TETRAHEDRON
C
 1051 CONTINUE
      L=L+1
      IEL(L)=LB
      DO 1052 I=1,4
      II=ITET1(I)
      IN(L,I)=IPRISM(LB,II)
 1052 CONTINUE
      GO TO 1020
 1020 CONTINUE
      NE2=L
C
C     DISPLAY NEW ELEMENT INCIDENCES
C
      IF (IECHO3.EQ.1) THEN
      WRITE(IW1,9120)
      WRITE(IW1,9121) (I,(IN(I,J),J=1,4),I=1,NE2)
      ENDIF
 9120 FORMAT(1X,'TETRAHEDRAL ELEMENT INCIDENCES'/1X,30('-')/
     2   1X,6X,'ELEM',6X,'NODES'/)
 9121 FORMAT(1X,5I10)
C
C     FIND HALF-BAND WIDTH
C
      IF (ISOLVE.EQ.1) THEN
      NB=0
      DO 102 L=1,NE2
      IMAX=0
      IMIN=10000
      DO 101 I=1,4
      II=IN(L,I)
      IF (II.GT.IMAX) IMAX=II
      IF (II.LT.IMIN) IMIN=II
 101  CONTINUE
      IF (IMAX-IMIN+1.GT.NB) NB=IMAX-IMIN+1
 102  CONTINUE
      WRITE(IW1,913) NB,NE2
      ENDIF
 913  FORMAT(/1X,'HALF-BAND WIDTH                             ',I10/
     1   1X,'NUMBER OF TETRAHEDRAL ELEMENTS                ',I10)
C
C     CONSTRUCT MATRIX (ICOL)
C
      IF (ISOLVE.EQ.2) THEN
      DO 4005 L=1,NE2
      DO 4004 I=1,4
      II=IN(L,I)
      DO 4003 J=1,4
      JJ=IN(L,J)
      DO 4002 IC=1,MAXNB
      IF (ICOL(II,IC).EQ.JJ) GOTO 4003
      IF (ICOL(II,IC).EQ.0) THEN
      ICOL(II,IC)=JJ
      GOTO 4003
      ENDIF
```

```
 4002 CONTINUE
      WRITE(*,*) 'EXCEED WIDTH'
      STOP
 4003 CONTINUE
 4004 CONTINUE
 4005 CONTINUE
      ENDIF
C
C     LOCATE DIAGONAL AND COUNT NUMBER OF ACTIVE COLUMNS
C
      IF (ISOLVE.EQ.2) THEN
      NB=0
      DO 4012 I=1,NN
      NC=0
      DO 4011 IC=1,MAXNB
      IF (ICOL(I,IC).EQ.I) IDAG(I)=IC
      IF (ICOL(I,IC).NE.0) NC=NC+1
 4011 CONTINUE
      NCOL(I)=NC
      IF (NC.GT.NB) NB=NC
 4012 CONTINUE
      WRITE(IW1,9139) NB,NE2
 9139 FORMAT(/1X,'HALF-BAND WIDTH                              ',I10/
     1   1X,'NUMBER OF TETRAHEDRAL ELEMENTS          ',I10)
      ENDIF
C
C     HYDRAULIC CONDUCTIVITY (FEET PER DAY), SPECIFIC STORAGE,
C        SPECIFIC YIELD (FOR ELEMENTS IN TOP OF GRID ONLY), AND
C        PARAMETER INDEXES
C
      WRITE(*,'(1X,''READING AQUIFER PARAMETERS IN MODEL'')')
      READ(IR1,914) FACKXX,FACKYY,FACKZZ,FACSS,FACSY,IECHO4,IECHO5
      READ(IR1,915) (KXX0(I),KYY0(I),KZZ0(I),SS0(I),SY0(I),
     1    ITOP(I),I=1,NE)
      DO 103 I=1,NE
      KXX0(I)=KXX0(I)*FACKXX
      KYY0(I)=KYY0(I)*FACKYY
      KZZ0(I)=KZZ0(I)*FACKZZ
      SS0(I)=SS0(I)*FACSS
      SY0(I)=SY0(I)*FACSY
 103  CONTINUE
      WRITE(IW1,916) FACKXX,FACKYY,FACKZZ,FACSS,FACSY,IECHO4,IECHO5
      IF (IECHO4.NE.0) THEN
      WRITE(IW1,9161)
      WRITE(IW1,917) (I,KXX0(I),KYY0(I),KZZ0(I),SS0(I),SY0(I),
     1   ITOP(I),I=1,NE)
      ENDIF
      IF (IFIT.EQ.1) THEN
      WRITE(*,'(1X,''READING INDEXES IN MODEL'')')
      READ(IR2,918) (INDKXX(I),INDKYY(I),INDKZZ(I),INDSS(I),
     1   INDSY(I),I=1,NE)
      IF (IECHO5.EQ.1) THEN
      WRITE(IW1,919)
      WRITE(IW1,920) (I,INDKXX(I),INDKYY(I),INDKZZ(I),INDSS(I),
     1   INDSY(I),I=1,NE)
      ENDIF
      ENDIF
      DO 104 I=1,NE
      KXX0(I)=KXX0(I)/(24.0*3600.0)
      KYY0(I)=KYY0(I)/(24.0*3600.0)
      KZZ0(I)=KZZ0(I)/(24.0*3600.0)
```

```
 104  CONTINUE
 914  FORMAT(5F10.0,2I10)
 915  FORMAT(10X,5F10.0,I10)
 918  FORMAT(10X,5I10)
 916  FORMAT(/1X,'AQUIFER CHARACTERISTICS'/1X,23('-')/
     1   1X,'FACTOR FOR KXX                              ',1PE10.2/
     2   1X,'FACTOR FOR KYY                              ',1PE10.2/
     3   1X,'FACTOR FOR KZZ                              ',1PE10.2/
     4   1X,'FACTOR FOR SS                               ',1PE10.2/
     5   1X,'FACTOR FOR SY                               ',1PE10.2/
     6   1X,'SWITCH FOR ECHO OF AQUIFER PARAMETERS   ',I10/
     7   1X,'SWITCH FOR ECHO OF BLOCK INDEXES        ',I10)
 9161 FORMAT(/1X,6X,'ELEM',7X,'KXX',7X,'KYY',7X,'KZZ',8X,'SS',
     1   8X,'SY',7X,'TOP'/)
 917  FORMAT((1X,I10,5(1PE10.2),I10))
 919  FORMAT(1X,'AQUIFER PARAMETER INDEXES'/1X,25('-')/
     1   1X,6X,'ELEM',7X,'KXX',7X,'KYY',7X,'KZZ',8X,'SS',8X,'SY',/)
 920  FORMAT(1X,6I10)
C
C     INITIAL WATER LEVELS (FEET)
C
      WRITE(*,'(1X,''READING INITIAL LEVELS IN MODEL'')')
      READ(IR1,9281) IECHO6
      READ(IR1,928) (H0(I),I=1,NN)
      WRITE(IW1,9291) IECHO6
      IF (IECHO6.NE.0) THEN
      WRITE(IW1,929)
      IDONE=0
      I1=-4
 566  I1=I1+5
      I2=I1+4
      IF (I2.GE.NN) THEN
      I2=NN
      IDONE=1
      ENDIF
      WRITE(IW1,930) I1,(H0(I),I=I1,I2)
      IF (IDONE.EQ.0) GO TO 566
      ENDIF
      IF (NPLOT.NE.0) THEN
      WRITE(IW2,9287) 0
      WRITE(IW2,9286) (H0(I),I=1,NN)
      ENDIF
 9281 FORMAT(I10)
 928  FORMAT(5F10.0)
 9291 FORMAT(/1X,'INITIAL WATER LEVELS'/1X,20('-')/
     1   1X,'SWITCH FOR ECHO OF WATER LEVELS         ',I10)
 929  FORMAT(/1X,6X,'NODE',4X,'LEVELS'/)
 930  FORMAT(1X,I10,5F10.2)
 9287 FORMAT(I10)
 9286 FORMAT(5F10.4)
C
C     CONSTANT-HEAD NODES, PUMPAGE AND RECHARGE, STREAM-AQUIFER
C        INTERACTION, AND EVAPOTRANSPIRATION
```

```
C
      CALL CHEAD1(NN)
      CALL FLUX1(NN)
      CALL RIVER1(NN)
      CALL EVAP1(NN)
      CALL VFLUX1
      CALL FAULT1
      CALL SINK1(NN,NE,NE2,IEL,HP,POR2,IPRISM,X,Y)
      CALL WATER1(NN)
      CLOSE (IR1)
      CLOSE (IR2)
      ICALL=0
      RETURN
C
C     END OF DATA INPUT
C
C     --------------------------------------------------------------
      ENTRY MODEL2(PARAMX,STEP,NODE,WEIGHT,WLC,NWLM,ICODE,IOUT,IFIT)
      ENTRY MODEL3(PARAMX,ICODE,IOUT,IFIT)
C     --------------------------------------------------------------
C
      IPASS=0
      ICALL=ICALL+1
      WRITE(*,9983) ICALL,ICODE
 9983 FORMAT(1X,'****CALL TO MODEL',I5,' WITH CODE',I5)
C
C     CURRENT VALUES FOR HYDRAULIC CONDUCTIVITY, SPECIFIC STORAGE,
C        SPECIFIC YIELD
C
      IF (IOUT.EQ.1) THEN
      OPEN (UNIT=1,FILE='AQUIFER.NEW',STATUS='UNKNOWN')
      DO 9999 L=1,NE
      XKXX=KXX0(L)*DEXP(PARAMX(INDKXX(L)))*86400.0
      XKYY=KYY0(L)*DEXP(PARAMX(INDKYY(L)))*86400.0
      XKZZ=KZZ0(L)*DEXP(PARAMX(INDKZZ(L)))*86400.0
      XSS=SS0(L)*(1.0-POR2(L))*DEXP(PARAMX(INDSS(L)))
      XSY=SY0(L)*DEXP(PARAMX(INDSY(L)))
      WRITE(1,'(I10,5(1PE10.2),I10)') L,XKXX,XKYY,XKZZ,XSS,XSY,
     1  ITOP(L)
 9999 CONTINUE
      CLOSE (1)
      ENDIF
C
      IF (IFIT.EQ.1) THEN
      DO 350 LT=1,NE2
      LP=IEL(LT)
      KXX(LT)=KXX0(LP)*DEXP(PARAMX(INDKXX(LP)))
      KYY(LT)=KYY0(LP)*DEXP(PARAMX(INDKYY(LP)))
            KZZ(LT)=KZZ0(LP)*DEXP(PARAMX(INDKZZ(LP)))
      SS(LT)=SS0(LP)*(1.0-POR2(LT))*DEXP(PARAMX(INDSS(LP)))
 350  CONTINUE
      DO 3501 LP=1,NE
      SY(LP)=SY0(LP)*DEXP(PARAMX(INDSY(LP)))
 3501 CONTINUE
      ELSE
      DO 3502 LT=1,NE2
      LP=IEL(LT)
      KXX(LT)=KXX0(LP)
      KYY(LT)=KYY0(LP)
      KZZ(LT)=KZZ0(LP)
      SS(LT)=SS0(LP)*(1.0-POR2(LT))
```

```
 3502 CONTINUE
      DO 3503 LP=1,NE
      SY(LP)=SY0(LP)
 3503 CONTINUE
      ENDIF
C
C     INITIAL WATER LEVELS
C
      DO 400 I=1,NN
      HL(I)=H0(I)
      H(I)=H0(I)
 400  CONTINUE
C
C     BEGIN TIME-STEP LOOP
C
      STIME=0.
      KNS=0
 401  KNS=KNS+1
      IF (KNS.GT.MAXKNS) RETURN
      FSS=1.0
      IF (KNS.EQ.1 .AND. ISS.EQ.1) THEN
      FSS=0.0
      STIME=-DELT(1)/(3600.0*24.0)
      ENDIF
      DELTX=DELT(KNS)
      STIME=STIME+(DELTX/(3600.0*24.0))
      CALL TIME(CLOCK)
      WRITE(*,9982) ICALL,KNS,CLOCK
 9982 FORMAT(1X,'    BEGINNING TIME STEP ',2I5,5X,A11)
C
C     INTERCHANGE WATER LEVELS
C
      DO 402 I=1,NN
      HL(I)=H(I)
 402  CONTINUE
C
C     COMPUTE VALUES FOR SOURCE/SINK VECTOR (Q) AND
C        COMPUTE CONSTANT-HEAD COEFFICIENTS (CCHRH) AND (CCHLH)
C
      IOUTX=0
      CALL FLUX2(QFLUX,SUMF,KNS,IOUTX,NN)
      CALL VFLUX2(H0,H,QVPAST,CVFLH,CVFRH,DELTX,SUMVF,NN,KNS,ISS,IOUTX)
      IF (ICALL.EQ.1) THEN
      IOUTX=1
      CALL WATER2(KNS,MAXKNS,ICMO1(KNS),ICMO2(KNS),ISS,QPUMP,QRECH,
     1   SUMP(KNS),SUMR(KNS),IOUTX,NN)
      IOUTX=0
      ENDIF
C
C     BEGIN EVAPOTRANSPIRATION, AND RIVER ITERATIONS
C
      DO 4171 I=1,NN
      HLIT(I)=HL(I)
 4171 CONTINUE
      DO 421 ITER=1,NITER
C
```

```
C     INITIALIZE MATRICES AND LOAD FAULT COEFFICIENTS
C
      IF (ISOLVE.EQ.1) THEN
      IF (ISKIP.EQ.1.AND.IPASS.EQ.1) GO TO 4140
      DO 406 I=1,NN
      DO 405 J=1,NB
      A(I,J)=0.0
      B(I,J)=0.0
      C(I,J)=0.0
      D(I,J)=0.0
  405 CONTINUE
  406 CONTINUE
      CALL FAULT2(A,ICOL,NCOL,ISOLVE)
      ENDIF
C
      IF (ISOLVE.EQ.2) THEN
      DO 606 I=1,NN
      NC=NCOL(I)
      DO 605 J=1,NC
      A(I,J)=0.0
      B(I,J)=0.0
  605 CONTINUE
  606 CONTINUE
      ISKIP=0
      IOUTX=0
      CALL FAULT2(A,ICOL,NCOL,ISOLVE)
      ENDIF
C
C     BEGIN LOOP OVER ELEMENTS
C
      DO 412 L=1,NE2
C
C     ELEMENT STIFFNESS MATRIX (AE) AND DYNAMIC MATRIX (BE)
C
      XH=0.0
      XHP=0.0
      DO 407 I=1,4
      II=IN(L,I)
      XE(I)=X(II)
      YE(I)=Y(II)
      ZE(I)=Z(II)
      XH=XH+H(II)*0.25
      XHP=XHP+HP(II)*0.25
  407 CONTINUE
      CALL SHAPE1(KXX(L),KYY(L),KZZ(L),SS(L),XE,YE,ZE,AE,BE,L)
      IHP=0
      IF (XH.LT.XHP) IHP=1
      CALL SINK2(XE,YE,ZE,CE,DE,L,IHP,POR2)
C
C     GLOBAL STIFFNESS MATRIX (A) AND DYNAMIC MATRIX (B)
C
      IF (ISOLVE.EQ.1) THEN
      DO 411 I=1,4
      II=IN(L,I)
      DO 410 J=1,4
      JJ=IN(L,J)-II+1
      IF (JJ.LT.1) GO TO 410
      A(II,JJ)=A(II,JJ)+AE(I,J)
      B(II,JJ)=B(II,JJ)+BE(I,J)
      C(II,JJ)=C(II,JJ)+CE(I,J)
      D(II,JJ)=D(II,JJ)+DE(I,J)
```

```
 410   CONTINUE
 411   CONTINUE
       ENDIF
C
       IF (ISOLVE.GT.1) THEN
       DO 611 I=1,4
       II=IN(L,I)
       DO 610 J=1,4
       JJ=IN(L,J)
       NC=NCOL(II)
       DO 6071 IC=1,NC
       IF (ICOL(II,IC).EQ.JJ) THEN
       A(II,IC)=A(II,IC)+AE(I,J)
       B(II,IC)=B(II,IC)+BE(I,J)
       C(II,IC)=C(II,IC)+CE(I,J)
       D(II,IC)=D(II,IC)+DE(I,J)
       GOTO 610
       ENDIF
 6071  CONTINUE
       WRITE(*,*) 'DID NOT FILL'
       STOP
 610   CONTINUE
 611   CONTINUE
       ENDIF
 412   CONTINUE
C
C      ADD FREE-SURFACE BOUNDARY CONDITION TO DYNAMIC| MATRIX (B)
C
       DO 520 L=1,NE
       IF (ITOP(L).EQ.0) GO TO 520
C
C      ELEMENT MATRIX (BE)
C
       DO 507 I=1,3
       II=IPRISM(L,I)
       XE(I)=X(II)
       YE(I)=Y(II)
 507   CONTINUE
       CALL SHAPE2(SY(L),XE,YE,BE,L)
C
C      GLOBAL MATRIX (B)
C
       IF (ISOLVE.EQ.1) THEN
       DO 511 I=1,3
       II=IPRISM(L,I)
       DO 510 J=1,3
       JJ=IPRISM(L,J)-II+1
       IF (JJ.LT.1) GO TO 510
       B(II,JJ)=B(II,JJ)+BE(I,J)
 510   CONTINUE
 511   CONTINUE
       ENDIF
```

```
C
      IF (ISOLVE.EQ.2) THEN
      DO 5115 I=1,3
      II=IPRISM(L,I)
      DO 5105 J=1,3
      JJ=IPRISM(L,J)
      NC=NCOL(II)
      DO 5075 IC=1,NC
      IF (ICOL(II,IC).EQ.JJ) THEN
      B(II,IC)=B(II,IC)+BE(I,J)
      GOTO 5105
      ENDIF
 5075 CONTINUE
 5105 CONTINUE
 5115 CONTINUE
      ENDIF
 520  CONTINUE
C
C     END LOOP OVER ELEMENTS
C     ADJUST MATRIX (B) FOR TIME-STEP LENGTH
C
      DO 414 I=1,NN
      DO 413 J=1,NB
      B(I,J)=B(I,J)*FSS/DELTX
      C(I,J)=C(I,J)*FSS/DELTX
      D(I,J)=D(I,J)*FSS/DELTX
 413  CONTINUE
 414  CONTINUE
C
C     COMPUTE RIVER AND EVAPOTRANSPIRATION COEFFICIENTS
C
      IOUTX=0
      CALL RIVER2(PARAMX,CRRH,CRLH,H,SUMRV,KNS,NN)
      CALL EVAP2(CETRH,CETLH,H,SUMET,KNS,IOUTX,NN)
      CALL CHEAD2(CCHRH,CCHLH,H,STIME,SUMCH,IOUTX,NN)
C
C     CONSTRUCT LEFT-HAND-SIDE MATRIX (G)
C
      IF (ISOLVE.EQ.1) THEN
      DO 419 I=1,NN
      DO 418 J=1,NB
      LHSM(I,J)=A(I,J)+B(I,J)+C(I,J)
 418  CONTINUE
      LHSM(I,1)=LHSM(I,1)+CCHLH(I)+CRLH(I)+CETLH(I)+CVFLH(I)
 419  CONTINUE
      ENDIF
C
      IF (ISOLVE.EQ.2) THEN
      DO 619 I=1,NN
      NC=NCOL(I)
      DO 618 J=1,NC
      LHSM(I,J)=A(I,J)+B(I,J)
 618  CONTINUE
      ID=IDAG(I)
      LHSM(I,ID)=LHSM(I,ID)+CCHLH(I)+CRLH(I)+CETLH(I)+CVFLH(I)
 619  CONTINUE
      ENDIF
 4140 CONTINUE
```

```
C
C     MULTIPLICATION OF GLOBAL DYNAMIC MATRIX (B) AND
C        LAST-WATER-LEVEL VECTOR (HL) TO PRODUCE VECTOR (C)
C
      IF (ISOLVE.EQ.1) THEN
      DO 417 I=1,NN
      F(I)=B(I,1)*HL(I)+C(I,1)*HP(I)+D(I,1)*(HL(I)-HP(I))
      DO 416 J=2,NB
      II=I+J-1
      IF (II.GT.NN) GO TO 415
      F(I)=F(I)+B(I,J)*HL(II)+C(I,J)*HP(II)+D(I,J)*(HL(II)-HP(II))
  415 II=I-J+1
      IF (II.LT.1) GO TO 416
      F(I)=F(I)+B(II,J)*HL(II)+C(II,J)*HP(II)+D(II,J)*(HL(II)-HP(II))
  416 CONTINUE
  417 CONTINUE
      ENDIF
C
      IF (ISOLVE.GT.1) THEN
      DO 617 I=1,NN
      SUM=0.0
      NC=NCOL(I)
      DO 616 IC=1,NC
      JJ=ICOL(I,IC)
      SUM=SUM+B(I,IC)*HL(JJ)+C(I,IC)*HP(JJ)+D(I,IC)*(HL(JJ)-HP(JJ))
  616 CONTINUE
      F(I)=SUM
  617 CONTINUE
      ENDIF
C
C     CONSTRUCT RIGHT-HAND-SIDE VECTOR (F)
C
      IF (ISOLVE.EQ.1) THEN
      DO 420 I=1,NN
      RHSV(I)=F(I)+QFLUX(I)+QPUMP(KNS,I)+QRECH(KNS,I)+QVPAST(I)+
     1   CCHRH(I)+CRRH(I)+CETRH(I)+CVFRH(I)
  420 CONTINUE
      ENDIF
C
      IF (ISOLVE.EQ.2) THEN
      DO 4208 I=1,NN
      RHSV(I)=F(I)+QFLUX(I)+QPUMP(KNS,I)+QRECH(KNS,I)+QVPAST(I)+
     1   CCHRH(I)+CRRH(I)+CETRH(I)+CVFRH(I)
 4208 CONTINUE
      ENDIF
C
C     COMPUTE NEW WATER-LEVEL VECTOR (H)
C
      IF (ISOLVE.EQ.1) THEN
      IF (IPASS.EQ.0 .OR. ISKIP.EQ.0) THEN
      CALL BAND1(LHSM,RHSV,H,NN,NB)
      ENDIF
      IF (ISS.EQ.0 .OR. KNS.EQ.2) IPASS=1
      CALL BAND2(LHSM,RHSV,H,NN,NB)
      ENDIF
C
      IF (ISOLVE.EQ.2) THEN
      CALL SOLVE2(LHSM,RHSV,H,NN,ICOL,IDAG,NCOL)
      ENDIF
```

```
C
C     CHECK FOR CLOSURE
C
      DELMAX=0.0
      DO 4201 I=1,NN
      DELH=ABS(H(I)-HLIT(I))
      IF (DELH.GT.DELMAX) DELMAX=DELH
      HLIT(I)=H(I)
 4201 CONTINUE
      IF (DELMAX.LE.CLOSE) GO TO 4211
 421  CONTINUE
      IF (ITER.GT.NITER) ITER=ITER-1
 4211 CONTINUE
C
C     END RIVER, AND EVAPOTRANSPIRATION ITERATIONS
C     ASSIGN COMPUTED INITIAL STEADY-STATE HEADS TO
C     INITIAL-HEAD VECTOR (H0)
C
      IF(ISS.EQ.1.AND.KNS.EQ.1) THEN
      DO 4225 I=1,NN
      H0(I)=H(I)
 4225 CONTINUE
      ENDIF
C
C     UPDATE COMPUTED-WATER-LEVEL VECTOR (WLC)
C
      IF (IFIT.EQ.1) THEN
      DO 422 IOBS=1,NWLM
      IF (STEP(IOBS).EQ.KNS) THEN
      WLC(IOBS)=0.0
      DO 4227 I=1,3
      WLC(IOBS)=WLC(IOBS)+H(NODE(IOBS,I))*WEIGHT(IOBS,I)
 4227 CONTINUE
      ENDIF
 422  CONTINUE
      ENDIF
      IF (IOUT.EQ.0) GO TO 401
      WRITE(IW1,980) KNS,STIME,ITER,DELMAX
      IF (NPRINT.NE.0) THEN
      IF (MOD(KNS-ISS,NPRINT).EQ.0) THEN
C
C     DISPLAY WATER LEVELS
C
      WRITE(IW1,981)
      IDONE=0
      I1=-4
 567  I1=I1+5
      I2=I1+4
      IF (I2.GE.NN) THEN
      I2=NN
      IDONE=1
      ENDIF
      WRITE(IW1,982) I1,(H(I),I=I1,I2)
      IF (IDONE.EQ.0) GO TO 567
      ENDIF
      ENDIF
      IF (NPLOT.NE.0) THEN
      IF (MOD(KNS-ISS,NPLOT).EQ.0) THEN
      WRITE(IW2,9288) KNS
      WRITE(IW2,9289) (H(I),I=1,NN)
      ENDIF
      ENDIF
```

```
 980  FORMAT(/1X,'TIME STEP',I10/1X,19('-')/
     1    1X,'ELAPSED TIME                                   ',1PE10.2/
     2    1X,'NUMBER OF ITERATIONS                           ',I10/
     3    1X,'WATER-LEVEL CHANGE ON LAST ITERATION           ',0PF10.3)
 981  FORMAT(/1X,'COMPUTED WATER LEVELS'/1X,21('-')/
     1    1X,6X,'NODE',4X,'LEVELS'/)
 982  FORMAT(1X,I10,5F10.2)
 9288 FORMAT(I10)
 9289 FORMAT(5F10.4)
 9298 FORMAT(5(1PE10.3))
C
C     ASSEMBLE AND DISPLAY OUTPUTS
C
C     WATER BUDGET
C
      IOUTX=0
      IF (NPRINT.NE.0) THEN
      IF (MOD(KNS-ISS,NPRINT).EQ.0) IOUTX=1
      ENDIF
C
C     ASSEMBLE FLUX OUTPUT
C
      DO 4187 I=1,NN
      QPOS(I)=0.0
      QNEG(I)=0.0
      IF (QPUMP(KNS,I).GE.0.0) QPOS(I)=QPOS(I)+QPUMP(KNS,I)
      IF (QPUMP(KNS,I).LT.0.0) QNEG(I)=QNEG(I)+QPUMP(KNS,I)
      IF (QRECH(KNS,I).GE.0.0) QPOS(I)=QPOS(I)+QRECH(KNS,I)
      IF (QRECH(KNS,I).LT.0.0) QNEG(I)=QNEG(I)+QRECH(KNS,I)
 4187 CONTINUE
C
C     CALL VFLUX, FAULT AND FLUX

      CALL VFLUX3(H0,H,QPOS,QNEG,QVPAST,CVFLH,CVFRH,DELTX,SUMVF,
     1    NN,KNS,ISS,IOUTX)
      CALL FAULT3(H,SUMFL,NN,IOUTX)
      CALL FLUX3(QFLUX,QPOS,QNEG,SUMF,KNS,IOUTX,NN)
C
C     CALL RIVER
C
      IOUTZ=1
      CALL RIVER3(CRRH,CRLH,H,QPOS,QNEG,SUMRV,DELTX,KNS,MAXKNS,
     1       IOUTZ,NN,IW20)
C
C     CALL EVAP, CHEAD, AND SINK
C
      CALL EVAP3(CETRH,CETLH,H,QPOS,QNEG,SUMET,KNS,IOUTX,NN)
      CALL CHEAD3(CCHRH,CCHLH,H,QPOS,QNEG,SUMCH,IOUTX,NN)
      CALL SINK3(X,Y,Z,XE,YE,ZE,IN,NN,NE,NE2,IEL,POR2,H,HL,HP,
     1    SUMQE,SUMQV,FSS,DELTX,IOUTX)
C
C     CALL BUDGET
C
      CALL BUDGET(X,Y,Z,IN,IPRISM,ITOP,H,HL,SS,SY,DELTX,NE2,NE,SUMF,
     1    SUMP(KNS),SUMR(KNS),SUMRV,SUMET,SUMCH,SUMVF,SUMFL,
     2    SUMQE,SUMQV,IOUT,FSS,KNS,STIME)
C
```

```
C     DISPLAY FLUX OUTPUT
C
      IF (NFLUX.NE.0) THEN
      IF (MOD(KNS-ISS,NFLUX).EQ.0) THEN
      WRITE(IW4,9288) KNS
      WRITE(IW4,9298) (QPOS(I),I=1,NN)
      WRITE(IW4,9298) (QNEG(I),I=1,NN)
      ENDIF
      ENDIF
C
C     END TIME-STEP LOOP
C
      GO TO 401
      END
```

```
C     --------------------
      SUBROUTINE EVAP1(NN)
C     --------------------
C
      INCLUDE 'PARAM.FOR'
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*2 (I-N)
      REAL*8 ETAREA(MAXETN),LAND(MAXETN),H0ET(MAXETN),QET(MAXETN),
     1   CETRH(MAXNN),CETLH(MAXNN),H(MAXNN),FACSET(MAXSTP),
     2   DELH0(MAXETN),ETMAX(MAXETN)
      REAL*8 QPOS(MAXNN),QNEG(MAXNN)
      INTEGER*2 ETNODE(MAXETN)
      IR1=11
      IW1=13
C
C     NUMBER OF ETNODES
C
      WRITE(*,'(1X,''READING IN EVAP'')')
      READ(IR1,900) NETN,MAXKNS,IECHO1,IECHO2
      WRITE(IW1,901) NETN,MAXKNS,IECHO1,IECHO2
      IF (NETN.EQ.0) RETURN
 900  FORMAT(4I10)
 901  FORMAT(/1X,'EVAPOTRANSPIRATION'/1X,18('-')/
     1   1X,'NUMBER OF NODES                             ',I10/
     2   1X,'NUMBER OF TIME STEPS                        ',I10/
     3   1X,'SWITCH FOR ECHO OF INPUT DATA               ',I10/
     4   1X,'SWITCH FOR ECHO OF FLUXES                   ',I10)
C
C     ET NODES, AREA (ACRES), LAND-SURFACE ALTITUDE (FEET)
C
      READ(IR1,902) (ETNODE(I),ETAREA(I),LAND(I),DELH0(I),
     1   ETMAX(I),I=1,NETN)
      IF (IECHO1.EQ.1) THEN
      WRITE(IW1,903)
      WRITE(IW1,904) (ETNODE(I),ETAREA(I),LAND(I),DELH0(I),
     1   ETMAX(I),I=1,NETN)
      ENDIF
 902  FORMAT(I10,4F10.0)
 903  FORMAT(/1X,6X,'NODE',6X,'AREA',6X,'LAND',5X,'DELH0',
     1   5X,'ETMAX'/)
 904  FORMAT(1X,I10,0PF10.2,0PF10.2,0PF10.2,1PE10.2)
C
C     TIME-SET MULTIPLIERS
C
      READ(IR1,910) (FACSET(I),I=1,MAXKNS)
      IF (IECHO1.EQ.1) THEN
      WRITE(IW1,911)
      WRITE(IW1,912) (I,FACSET(I),I=1,MAXKNS)
      ENDIF
 910  FORMAT(10X,F10.0)
 911  FORMAT(/1X,'TIME-STEP MULTIPLIERS'/1X,21('-')/
     1   1X,4X,'STEP',4X,'FACTOR'/)
 912  FORMAT(1X,I10,1PE10.2)
C
C     ADJUST LAND ALTITUDES TO REPRESENT ZERO-ET ALTITUDE
C
      DO 101 I=1,NETN
      ETAREA(I)=ETAREA(I)*43560.0
      H0ET(I)=LAND(I)-DELH0(I)
      ETMAX(I)=ETMAX(I)/(365.0*24.0*3600.0)
```

```
 101  CONTINUE
      RETURN
C
C     END OF DATA INPUT
C
C     ------------------------------------------
      ENTRY EVAP2(CETRH,CETLH,H,SUMET,KNS,IOUT,NN)
C     ------------------------------------------
C
C     INITIALIZE VECTORS (CETRH) AND (CETLH)
C
      DO 100 I=1,NN
      CETRH(I)=0.0
      CETLH(I)=0.0
 100  CONTINUE
      IF (NETN.EQ.0) RETURN
C
C     COMPUTE LEFT-HAND AND RIGHT-HAND COEFFICIENTS
C
      DO 110 I=1,NETN
      J=ETNODE(I)
      IF (H(J).LE.H0ET(I)) THEN
      CETRH(J)=0.0
      CETLH(J)=0.0
      ELSE
      COEF=ETAREA(I)*ETMAX(I)*FACSET(KNS)/DELH0(I)
      CETRH(J)=COEF*H0ET(I)
      CETLH(J)=COEF
      ENDIF
 110  CONTINUE
      RETURN
C
C     -------------------------------------------------------
      ENTRY EVAP3(CETRH,CETLH,H,QPOS,QNEG,SUMET,KNS,IOUT,NN)
C     -------------------------------------------------------
C
C     COMPUTE ET FLUXES AND DISPLAY RESULTS
C
      SUMET=0.0
      IF (NETN.EQ.0) RETURN
      DO 120 I=1,NETN
      J=ETNODE(I)
      QET(I)=CETRH(J)-CETLH(J)*H(J)
      IF (QET(I).GE.0.0) QPOS(J)=QPOS(J)+QET(I)
      IF (QET(I).LT.0.0) QNEG(J)=QNEG(J)+QET(I)
      SUMET=SUMET+QET(I)
 120  CONTINUE
      IF (IECHO2.EQ.1 .AND. IOUT.EQ.1) THEN
      WRITE(IW1,950) SUMET
      WRITE(IW1,951) (ETNODE(I),LAND(I)-H(ETNODE(I)),QET(I),I=1,NETN)
      ENDIF
 950  FORMAT(/1X,'EVAPOTRANSPIRATION'/1X,18('-')/
     1   1X,'CUMULATIVE RATE',1PE10.2/
     2   1X,6X,'NODE',5X,'DEPTH',6X,'RATE'/)
 951  FORMAT(1X,I10,0PF10.2,1PE10.2)
      RETURN
      END
```

```
C     --------------------
      SUBROUTINE RIVER1(NN)
C     --------------------
C
      INCLUDE 'PARAM.FOR'
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*2 (I-N)
      REAL*8 PARAMX(MAXPAR),KBED0(MAXRCH,MAXRIV)
      INTEGER*2 INDRIV(MAXRCH,MAXRIV)
      REAL*8 HBED(MAXRCH,MAXRIV),LENGTH(MAXRCH,MAXRIV),
     1   KBED(MAXRCH,MAXRIV),BBED(MAXRCH,MAXRIV),QRIV(MAXRCH,MAXRIV),
     2   FACA(MAXRCH,MAXRIV)
      REAL*8 CRRH(MAXNN),CRLH(MAXNN),H(MAXNN),KFACT
      REAL*8 QIN(MAXINF,MAXSTP)
      REAL*8 XSUMQ1(MAXRCH,MAXRIV),XSUMQ2(MAXRCH,MAXRIV),
     1   XSUMQ3(MAXRCH,MAXRIV),XSUMQ4(MAXRCH,MAXRIV),
     2   XSUMQ5(MAXRCH,MAXRIV)
      INTEGER*2 RNODE(MAXRCH,MAXRIV),SET(MAXRCH,MAXRIV),NRN(MAXRCH),
     1   JOIN(MAXRCH,MAXRIV),ICOEF(MAXRCH,MAXRIV),ITABLE(MAXRCH,MAXRIV)
      REAL*8 QTAB(MAXSEC,MAXPTS),WTAB(MAXSEC,MAXPTS),DTAB(MAXSEC,MAXPTS)
      REAL*8 QPOS(MAXNN),QNEG(MAXNN)
      INTEGER*2 NPT(MAXSEC)
      IR1=11
      IW1=13
C
C     NUMBER OF RIVER NODES, TRIBUTARY NODES, AND TIME STEPS
C
      READ(IR1,900) NRCH,NINF,NSTEP,IECHO1,IECHO2,IECHO3
      IF (NRCH.EQ.0) RETURN
C
C     CHANNEL GEOMETRY
C
      READ(IR1,901) AD,BD,AW,BW
      WRITE(IW1,920) NRCH,NINF,NSTEP,IECHO1,IECHO2,IECHO3
      IF (IECHO1.EQ.1) WRITE(IW1,902) AD,BD,AW,BW
 900  FORMAT(6I10)
 901  FORMAT(4F10.0)
 920  FORMAT(/1X,'STREAM-AQUIFER INTERACTION'/1X,26('-')//
     1   1X,'NUMBER OF CHANNEL REACHES       ',I10/
     2   1X,'NUMBER OF INFLOWS               ',I10/
     3   1X,'NUMBER OF TIME STEPS            ',I10/
     5   1X,'ECHO CHANNEL SPECIFICATIONS     ',I10/
     6   1X,'ECHO CHANNEL INFLOWS            ',I10/
     7   1X,'ECHO CHANNEL FLOWS              ',I10)
 902  FORMAT(/1X,'DEPTH COEFFICIENT                  ',1PE10.2/
     2   1X,'DEPTH EXPONENT                  ',1PE10.2/
     3   1X,'WIDTH COEFFICIENT               ',1PE10.2/
     4   1X,'WIDTH EXPONENT                  ',1PE10.2)
C
C     RIVER NODES, ALTITUDE OF RIVER BED, REACH LENGTH,
C        CHANNEL-BED LEAKANCE, INFLOW IDENTIFIER
C
      DO 100 IRCH=1,NRCH
      READ(IR1,922) NRN(IRCH),KFACT
      READ(IR1,903) (RNODE(IRCH,I),HBED(IRCH,I),LENGTH(IRCH,I),
     1   KBED0(IRCH,I),BBED(IRCH,I),JOIN(IRCH,I),SET(IRCH,I),
     2   ITABLE(IRCH,I),INDRIV(IRCH,I),I=1,NRN(IRCH))
```

```
 100   CONTINUE
       DO 1001 IRCH=1,NRCH
       IF (IECHO1.EQ.1) THEN
       WRITE(IW1,904) IRCH,KFACT
       WRITE(IW1,905) (RNODE(IRCH,I),HBED(IRCH,I),LENGTH(IRCH,I),
     1   KBED0(IRCH,I),BBED(IRCH,I),JOIN(IRCH,I),SET(IRCH,I),
     2   ITABLE(IRCH,I),INDRIV(IRCH,I),I=1,NRN(IRCH))
       ENDIF
       DO 1002 I=1,NRN(IRCH)
       KBED0(IRCH,I)=KBED0(IRCH,I)*KFACT/(24.0*3600.0)
 1002  CONTINUE
 1001  CONTINUE
 903   FORMAT(I10,4F10.0,4I10)
 904   FORMAT(/1X,'RIVER REACH          ',I10/1X,30('-')//
     1   1X,'FACTOR FOR BED PERMEABILITY',1PE10.2//
     2   1X,6X,'NODE',2X,'ALTITUDE',4X,'LENGTH',6X,'PERM',
     3   1X,'THICKNESS',3X,'JOINING',7X,'SET',5X,'TABLE'
     4   5X,'INDEX'/)
 905   FORMAT(1X,I10,2F10.2,1PE10.2,0PF10.2,4I10)
 922   FORMAT(I10,F10.0)
C
C      CHANNEL GEOMETRY TABLES
C
       READ(IR1,940) NTABLE
       DO 132 ITAB=1,NTABLE
       READ(IR1,942) NPT(ITAB)
       READ(IR1,943) (QTAB(ITAB,IPT),WTAB(ITAB,IPT),
     1   DTAB(ITAB,IPT),IPT=1,NPT(ITAB))
       IF (IECHO1.EQ.1) THEN
       WRITE(IW1,941) ITAB
       WRITE(IW1,944) (IPT,QTAB(ITAB,IPT),WTAB(ITAB,IPT),
     1   DTAB(ITAB,IPT),IPT=1,NPT(ITAB))
       ENDIF
 132   CONTINUE
 940   FORMAT(I10)
 941   FORMAT(/1X,'CHANNEL GEOMETRY TABLE',I8/1X,30('-')/
     1   1X,5X,'POINT',6X,'FLOW',5X,'WIDTH',5X,'DEPTH'/)
 942   FORMAT(I10)
 943   FORMAT(3F10.0)
 944   FORMAT(I10,3F10.1)
C
C      RIVER AND TRIBUTARY INFLOWS
C
       SUM=0.0
       DO 102 I=1,NINF
       READ(IR1,9071) RFACT
       READ(IR1,907) (QIN(I,J),J=1,NSTEP)
       DO 153 J=1,NSTEP
       QIN(I,J)=QIN(I,J)*RFACT
       SUM=SUM+QIN(I,J)
 153   CONTINUE
       SUM=SUM/NSTEP
       IF (IECHO2.EQ.1) THEN
       WRITE(IW1,908) I
       WRITE(IW1,909) (QIN(I,J),J=1,NSTEP)
       WRITE(IW1,9099) SUM
       ENDIF
```

```
  102   CONTINUE
  9071 FORMAT(F10.0)
  907   FORMAT(5F10.0)
  908   FORMAT(/1X,'INFLOW SET',I10/1X,20('-')/)
  909   FORMAT((1X,5F10.2))
  9099 FORMAT(/1X,'AVERAGE FLOW',F8.2)
        RETURN
C
C       END OF DATA INPUT
C
C       ------------------------------------------------
        ENTRY RIVER2(PARAMX,CRRH,CRLH,H,SUMR,ISTEP,NN)
C       ------------------------------------------------
C
C       ADJUST BED PERMEABILITY
C
        DO 3001 IRCH=1,NRCH
        DO 3002 I=1,NRN(IRCH)
        KBED(IRCH,I)=KBED0(IRCH,I)*DEXP(PARAMX(INDRIV(IRCH,I)))
 3002   CONTINUE
 3001   CONTINUE

C       INITIALIZE VECTORS (CRRH) AND (CRLH)
C
        DO 160 I=1,NN
        CRRH(I)=0.0
        CRLH(I)=0.0
  160   CONTINUE
        SUMR=0.0
        IF (NRCH.EQ.0) RETURN
C
C       ROUTE WATER IN RIVER AND COMPUTE RIGHT-HAND AND LEFT-HAND
C          COEFFICIENTS
C          Q1   INFLOW FROM UPSTREAM
C          Q2   INFLOW FROM TRIBUTARY
C          Q3   INFLOW FROM NON-TRIBUTARY
C          Q4   OUTFLOW THROUGH RIVER BED
C          Q5   OUTFLOW TO DOWNSTREAM
C
        DO 265 IRCH=1,NRCH
        DO 263 I=1,NRN(IRCH)
C
C       SET VALUES FOR KNOWN INFLOWS
C
        Q1=0.0
        IF (I.NE.1) Q1=QRIV(IRCH,I-1)
        Q2=0.0
        IF (JOIN(IRCH,I).NE.0) THEN
        JJ=JOIN(IRCH,I)
        II=NRN(JJ)
        Q2=QRIV(JJ,II)
        ENDIF
        Q3=0.0
        IF (SET(IRCH,I).NE.0) THEN
        JJ=SET(IRCH,I)
        Q3=QIN(JJ,ISTEP)
        ENDIF
        QSUM=Q1+Q2+Q3
        IF (QSUM.LT.0.0) QSUM=0.0
```

```
C
C     CHANNEL GEOMETRY FROM EXPONENTIAL RELATIONS
C
      IF (ITABLE(IRCH,I).EQ.0) THEN
      WIDTH=AW*QSUM**BW
      DEPTH=AD*QSUM**BD
      GOTO 149
      ENDIF
C
C     CHANNEL GEOMETRY FROM TABLE
C
      IF (ITABLE(IRCH,I).NE.0) THEN
      ITAB=ITABLE(IRCH,I)
C
      IF (QSUM.LE.QTAB(ITAB,1)) THEN
      WIDTH=WTAB(ITAB,1)
      DEPTH=DTAB(ITAB,1)
      GOTO 149
      ENDIF
C
      IF (QSUM.GE.QTAB(ITAB,NPT(ITAB))) THEN
      WIDTH=WTAB(ITAB,NPT(ITAB))
      DEPTH=DTAB(ITAB,NPT(ITAB))
      GOTO 149
      ENDIF
C
      DO 146 IPT=2,NPT(ITAB)
      IF (QSUM.GE.QTAB(ITAB,IPT-1) .AND. QSUM.LE.QTAB(ITAB,IPT)) THEN
      WIDTH=WTAB(ITAB,IPT-1)+(WTAB(ITAB,IPT)-WTAB(ITAB,IPT-1))*
     1   (QSUM-QTAB(ITAB,IPT-1))/(QTAB(ITAB,IPT)-QTAB(ITAB,IPT-1))
      DEPTH=DTAB(ITAB,IPT-1)+(DTAB(ITAB,IPT)-DTAB(ITAB,IPT-1))*
     1   (QSUM-QTAB(ITAB,IPT-1))/(QTAB(ITAB,IPT)-QTAB(ITAB,IPT-1))
      GOTO 149
      ENDIF
 146  CONTINUE
C
      ENDIF
 149  CONTINUE
      HEAD=HBED(IRCH,I)+DEPTH
      AREA0=WIDTH*LENGTH(IRCH,I)
C
C     COMPUTE LEAKAGE RATE AND MAXIMUM LEAKAGE RATE
C
      JJ=RNODE(IRCH,I)
      Q4=AREA0*KBED(IRCH,I)*(HEAD-H(JJ))/BBED(IRCH,I)
      QMAX=AREA0*KBED(IRCH,I)*(BBED(IRCH,I)+DEPTH)/BBED(IRCH,I)
C
C     CHECK FOR LEAKAGE RATE GREATER THAN INFLOW OR GREATER
C        THAN MAXIMUM RATE FOR UNIT GRADIENT
C
      IF (Q4.LT.0.0) THEN
      FACA(IRCH,I)=1.0
      AREA=AREA0*FACA(IRCH,I)
      Q4=Q4
      ICOEF(IRCH,I)=1
      GO TO 262
      ENDIF
```

```
C
      IF (QSUM.EQ.0.0) THEN
      FACA(IRCH,I)=0.0
      AREA=AREA0*FACA(IRCH,I)
      Q4=0.0
      ICOEF(IRCH,I)=2
      GO TO 262
      ENDIF
C
      IF (Q4.LE.DMIN1(QSUM,QMAX)) THEN
      FACA(IRCH,I)=1.0
      AREA=AREA0*FACA(IRCH,I)
      Q4=Q4
      ICOEF(IRCH,I)=3
      GO TO 262
      ENDIF
C
      IF (QSUM.LE.DMIN1(Q4,QMAX)) THEN
      FACA(IRCH,I)=QSUM/Q4
      AREA=AREA0*FACA(IRCH,I)
      Q4=QSUM
      ICOEF(IRCH,I)=4
      GO TO 262
      ENDIF
C
      IF (QMAX.LE.DMIN1(Q4,QSUM)) THEN
      FACA(IRCH,I)=1.0
      AREA=AREA0*FACA(IRCH,I)
      Q4=QMAX
      ICOEF(IRCH,I)=5
      GO TO 262
      ENDIF
C
 262  CONTINUE
      QRIV(IRCH,I)=QSUM-Q4
C
C     COMPUTE RIVER COEFFICIENTS
C
      JJ=RNODE(IRCH,I)
      IF (ICOEF(IRCH,I).LE.4) THEN
      CRLH(JJ)=AREA*KBED(IRCH,I)/BBED(IRCH,I)
      CRRH(JJ)=AREA*KBED(IRCH,I)*HEAD/BBED(IRCH,I)
      ENDIF
      IF (ICOEF(IRCH,I).EQ.5) THEN
      CRLH(JJ)=0.0
      CRRH(JJ)=Q4
      ENDIF
 263  CONTINUE
 265  CONTINUE
      RETURN
C
C     --------------------------------------------------
      ENTRY RIVER3(CRRH,CRLH,H,QPOS,QNEG,SUMR,DELT,ISTEP,
     1   NSTEPX,IOUT,NN,IW20)
C     --------------------------------------------------
```

```
C
      SUMR=0.0
      IF (NRCH.EQ.0) RETURN
      IF (ISTEP.EQ.1) THEN
      SRESID=0.0
      STIME=0.0
      DO 335 IRCH=1,NRCH
      DO 334 I=1,NRN(IRCH)
      XSUMQ1(IRCH,I)=0.0
      XSUMQ2(IRCH,I)=0.0
      XSUMQ3(IRCH,I)=0.0
      XSUMQ4(IRCH,I)=0.0
      XSUMQ5(IRCH,I)=0.0
  334 CONTINUE
  335 CONTINUE
      ENDIF
C
C     DISPLAY RIVER BUDGET COMPONENTS
C
      SUMQ3=0.0
      SUMQ4=0.0
      DO 167 IRCH=1,NRCH
      IF (IOUT.EQ.1 .AND. IECHO3.EQ.1) THEN
        WRITE(IW20,929) ISTEP
        WRITE(IW20,930) IRCH
      ENDIF
      DO 166 I=1,NRN(IRCH)
      Q1=0.0
      IF (I.NE.1) Q1=QRIV(IRCH,I-1)
      Q2=0.0
      IF (JOIN(IRCH,I).NE.0) THEN
      JJ=JOIN(IRCH,I)
      II=NRN(JJ)
      Q2=QRIV(JJ,II)
      ENDIF
      Q3=0.0
      IF (SET(IRCH,I).NE.0) THEN
      JJ=SET(IRCH,I)
      Q3=QIN(JJ,ISTEP)
      QSUM=Q1+Q2+Q3
      IF (QSUM.LT.0.0) Q3=-Q1-Q2
      ENDIF
      JJ=RNODE(IRCH,I)
      IF (ICOEF(IRCH,I).LE.4) Q4=CRRH(JJ)-CRLH(JJ)*H(JJ)
      IF (ICOEF(IRCH,I).EQ.5) Q4=CRRH(JJ)
      Q5=QRIV(IRCH,I)
      IF (Q4.GE.0.0) QPOS(JJ)=QPOS(JJ)+Q4
      IF (Q4.LE.0.0) QNEG(JJ)=QNEG(JJ)+Q4
      IF (IOUT.EQ.1 .AND. IECHO3.EQ.1) WRITE(IW20,931) RNODE(IRCH,I),
     1   Q1,Q2,Q3,Q4,Q5,ICOEF(IRCH,I)
C
C     CUMULATIVE RIVER BUDGET
C
      SUMQ3=SUMQ3+Q3
      SUMQ4=SUMQ4+Q4
      XSUMQ1(IRCH,I)=XSUMQ1(IRCH,I)+Q1
      XSUMQ2(IRCH,I)=XSUMQ2(IRCH,I)+Q2
      XSUMQ3(IRCH,I)=XSUMQ3(IRCH,I)+Q3
      XSUMQ4(IRCH,I)=XSUMQ4(IRCH,I)+Q4
      XSUMQ5(IRCH,I)=XSUMQ5(IRCH,I)+Q5
```

```
 166  CONTINUE
 167  CONTINUE
      SUMR=SUMQ4
C
C     DISPLAY DETAILED AVERAGE BUDGET
C
      IF (ISTEP.EQ.NSTEPX .AND. IOUT.EQ.1) THEN
      DO 267 IRCH=1,NRCH
      WRITE(IW20,9301) IRCH
      DO 266 I=1,NRN(IRCH)
      Q1=XSUMQ1(IRCH,I)/FLOAT(NSTEPX)
      Q2=XSUMQ2(IRCH,I)/FLOAT(NSTEPX)
      Q3=XSUMQ3(IRCH,I)/FLOAT(NSTEPX)
      Q4=XSUMQ4(IRCH,I)/FLOAT(NSTEPX)
      Q5=XSUMQ5(IRCH,I)/FLOAT(NSTEPX)
      WRITE(IW20,9311) RNODE(IRCH,I),Q1,Q2,Q3,Q4,Q5
 266  CONTINUE
 267  CONTINUE
      ENDIF
C
C     DISPLAY SUMMARY CUMULATIVE BUDGET
C
      Q5=QRIV(NRCH,NRN(NRCH))
      RESID=SUMQ3-SUMQ4-QRIV(NRCH,NRN(NRCH))
      SRESID=SRESID+RESID*DELT
      STIME=STIME+DELT
      IF (IOUT.EQ.1) WRITE(IW20,932) SUMQ3,SUMQ4,QRIV(NRCH,NRN(NRCH)),
     1   RESID,SRESID/STIME
 929  FORMAT(/1X,'TIME STEP ',I10)
 930  FORMAT(/1X,'SURFACE-WATER BUDGET COMPONENTS REACH    ',I10/
     1   1X,50('-')//
     2   1X,6X,'NODE',2X,'UPSTREAM',1X,'TRIBUTARY',4X,'INFLOW',
     3   2X,'RECHARGE',3X,'OUTFLOW',6X,'CODE'/)
 931  FORMAT(1X,I10,5F10.2,I10)
 9301  FORMAT(/1X,'AVERAGE SURFACE-WATER BUDGET COMPONENTS REACH     ',
     1   I10/1X,60('-')//
     2   1X,6X,'NODE',2X,'UPSTREAM',1X,'TRIBUTARY',4X,'INFLOW',
     3   2X,'RECHARGE',3X,'OUTFLOW'/)
 9311 FORMAT(1X,I10,5F10.2)
 932  FORMAT(/1X,'SURFACE-WATER BUDGET'/1X,20('-')//
     1   1X,'INFLOWS                        ',1PE10.2/
     2   1X,'OUTFLOW THROUGH RIVER BED      ',1PE10.2/
     3   1X,'OUTFLOW FROM LAST REACH        ',1PE10.2/
     4   1X,'RESIDUAL                       ',1PE10.2/
     5   1X,'CUMULATIVE RESIDUAL            ',1PE10.2)
 981  FORMAT(1X,I10,4F10.2)
      RETURN
      END
```

```
C     ---------------------
      SUBROUTINE CHEAD1(NN)
C     ---------------------
C
      INCLUDE 'PARAM.FOR'
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*2 (I-N)
      REAL*8 CHEAD(MAXCHN),CCHRH(MAXNN),CCHLH(MAXNN),H(MAXNN),
     1   QCH(MAXCHN),K(MAXCHN),B(MAXCHN),
     2   AREA(MAXCHN),CHEAD0(MAXCHN)
      REAL*8 TTAB(MAXCHN,MAXTAB),CHTAB(MAXCHN,MAXTAB),XCHTAB(MAXCHN)
      REAL*8 QPOS(MAXNN),QNEG(MAXNN)
      INTEGER*2 CHNODE(MAXCHN),TYPE(MAXCHN),TABLE(MAXCHN),NPT(MAXCHN)
      IR1=11
      IW1=13
C
C     BEGIN DATA INPUT
C     CONSTANT-HEAD NODES AND WATER-LEVEL VALUES
C
      WRITE(*,'(1X,''READING IN CHEAD'')')
      READ(IR1,900) NCHN,IECHO1,IECHO2
      WRITE(IW1,901) NCHN,IECHO1,IECHO2
      IF (NCHN.EQ.0) RETURN
      READ(IR1,902) (CHNODE(I),CHEAD0(I),K(I),B(I),AREA(I),
     1   TYPE(I),TABLE(I),I=1,NCHN)
      IF (IECHO1.EQ.1) THEN
      WRITE(IW1,903)
      WRITE(IW1,904) (CHNODE(I),CHEAD0(I),K(I),B(I),AREA(I),
     1   TYPE(I),TABLE(I),I=1,NCHN)
      ENDIF
      DO 120 I=1,NCHN
      K(I)=K(I)/(24.0*3600.0)
  120 CONTINUE
  900 FORMAT(3I10)
  901 FORMAT(/1X,'CONSTANT-HEAD NODES'/1X,19('-')/
     1   1X,'NUMBER OF NODES                            ',I10/
     2   1X,'SWITCH FOR ECHO OF INPUT HEADS             ',I10/
     3   1X,'SWITCH FOR ECHO OF STEP FLUXES             ',I10)
  902 FORMAT(I10,4F10.0,2I10)
  903 FORMAT(/1X,6X,'NODE',6X,'HEAD',9X,'K',9X,'B',6X,'AREA',
     1   6X,'TYPE',5X,'TABLE'/)
  904 FORMAT(1X,I10,0PF10.2,1PE10.2,1PE10.2,1PE10.2,2I10)
C
C     CONSTANT-HEAD TABLES
C
      READ(IR1,951) NTABLE
      IF (NTABLE.NE.0) THEN
      DO 351 ITABLE=1,NTABLE
      READ(IR1,952) NPT(ITABLE)
      READ(IR1,953) (TTAB(ITABLE,IPT),CHTAB(ITABLE,IPT),IPT=1,
     1   NPT(ITABLE))
      WRITE(IW1,954) ITABLE
      WRITE(IW1,955) (IPT,TTAB(ITABLE,IPT),CHTAB(ITABLE,IPT),IPT=1,
     1   NPT(ITABLE))
  351 CONTINUE
      ENDIF
  951 FORMAT(I10)
  952 FORMAT(I10)
  953 FORMAT(2F10.0)
  954 FORMAT(/1X,'CONSTANT-HEAD TABLE',I10/1X,29('-')//
     1   1X,5X,'POINT',6X,'TIME',6X,'HEAD'/)
```

```
 955  FORMAT((1X,I10,2F10.2))
      RETURN
C
C     END OF DATA INPUT
C
C     -----------------------------------------------
      ENTRY CHEAD2(CCHRH,CCHLH,H,STIME,SUMCH,IOUT,NN)
C     -----------------------------------------------
C
C     INITIALIZE VECTORS (CCHRH) AND (CCHLH)
C
      DO 100 I=1,NN
      CCHRH(I)=0.0
      CCHLH(I)=0.0
 100  CONTINUE
      SUMCH=0.0
      IF (NCHN.EQ.0) RETURN
C
C     CONSTANT HEADS FROM TABLES
C
      IF (NTABLE.NE.0) THEN
      DO 361 ITABLE=1,NTABLE
      IF (STIME.LE.TTAB(ITABLE,1)) THEN
      XCHTAB(ITABLE)=CHTAB(ITABLE,1)
      GOTO 361
      ENDIF
      IF (STIME.GE.TTAB(ITABLE,NPT(ITABLE))) THEN
      XCHTAB(ITABLE)=CHTAB(ITABLE,NPT(ITABLE))
      GOTO 361
      ENDIF
      DO 360 IPT=2,NPT(ITABLE)
      IF (STIME.LE.TTAB(ITABLE,IPT)) THEN
      CH1=CHTAB(ITABLE,IPT-1)
      CH2=CHTAB(ITABLE,IPT)
      T1=TTAB(ITABLE,IPT-1)
      T2=TTAB(ITABLE,IPT)
      XCHTAB(ITABLE)=CH1+(CH2-CH1)*(STIME-T1)/(T2-T1)
      GOTO 361
      ENDIF
 360  CONTINUE
 361  CONTINUE
      ENDIF
C
C     COMPUTE LEFT-HAND AND RIGHT-HAND COEFFICIENTS
C
      DO 101 I=1,NCHN
C
      J=CHNODE(I)
      ITABLE=TABLE(I)
      IF (ITABLE.NE.0) CHEAD(I)=CHEAD0(I)+XCHTAB(ITABLE)
      IF (ITABLE.EQ.0) CHEAD(I)=CHEAD0(I)
      DIFF=CHEAD(I)-H(J)
C
      IF (TYPE(I).EQ.0) THEN
      CCHLH(J)=AREA(I)*K(I)/B(I)
      CCHRH(J)=AREA(I)*K(I)*CHEAD(I)/B(I)
      GOTO 101
      ENDIF
```

```
C
      IF (TYPE(I).EQ.1) THEN
      IF (DIFF.LE.B(I)) THEN
      CCHLH(J)=AREA(I)*K(I)/B(I)
      CCHRH(J)=AREA(I)*K(I)*CHEAD(I)/B(I)
      GOTO 101
      ELSE
      CCHLH(J)=0.0
      CCHRH(J)=AREA(I)*K(I)
      GOTO 101
      ENDIF
      ENDIF
C
      IF (TYPE(I).EQ.2) THEN
      IF (DIFF.LE.0.0) THEN
      CCHLH(J)=AREA(I)*K(I)/B(I)
      CCHRH(J)=AREA(I)*K(I)*CHEAD(I)/B(I)
      GOTO 101
      ELSE
      CCHLH(J)=0.0
      CCHRH(J)=0.0
      GOTO 101
      ENDIF
      ENDIF
C
 101  CONTINUE
      RETURN
C     -----------------------------------------------------
      ENTRY CHEAD3(CCHRH,CCHLH,H,QPOS,QNEG,SUMCH,IOUT,NN)
C     -----------------------------------------------------
C
C     COMPUTE CONSTANT-HEAD FLUXES AND DISPLAY RESULTS
C
      SUMCH=0.0
      IF (NCHN.EQ.0) RETURN
      DO 150 I=1,NCHN
      J=CHNODE(I)
      QCH(I)=CCHRH(J)-CCHLH(J)*H(J)
      IF (QCH(I).GE.0.0) QPOS(J)=QPOS(J)+QCH(I)
      IF (QCH(I).LT.0.0) QNEG(J)=QNEG(J)+QCH(I)
      SUMCH=SUMCH+QCH(I)
 150  CONTINUE
      IF (IECHO2.EQ.1 .AND. IOUT.EQ.1) THEN
      WRITE(IW1,905) SUMCH
      WRITE(IW1,906) (CHNODE(I),CHEAD(I),H(CHNODE(I)),
     1   QCH(I),I=1,NCHN)
      ENDIF
 905  FORMAT(/1X,'CONSTANT-HEAD NODES'/1X,19('-')/
     1   1X,'CUMULATIVE RATE',1PE10.2/
     2   1X,6X,'NODE',5X,'CHEAD',9X,'H',6X,'RATE'/)
 906  FORMAT(1X,I10,0PF10.2,0PF10.2,1PE10.2)
      RETURN
      END
```

```
C     --------------------
      SUBROUTINE FLUX1(NN)
C     --------------------
C
      INCLUDE 'PARAM.FOR'
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*2 (I-N)
      REAL*8 QFLUX(MAXNN),QSET(MAXNN,MAXSET),FACSET(MAXSTP,MAXSET)
      REAL*8 QPOS(MAXNN),QNEG(MAXNN)
      INTEGER*2 QNODE(MAXNN)
      IR1=11
      IW1=13
C
C     NUMBER OF PUMPING PERIODS AND FACTOR FOR PUMPING
C
      WRITE(*,'(1X,''READING IN FLUX'')')
      READ(IR1,900) NQSET,MAXKNS,FACQ,IECHO1,IECHO2
      WRITE(IW1,901) NQSET,MAXKNS,FACQ,IECHO1,IECHO2
      IF (NQSET.EQ.0) RETURN
 900  FORMAT(2I10,F10.0,2I10)
 901  FORMAT(/1X,'SOURCE AND SINK FLUXES'/1X,22('-')/
     1   1X,'NUMBER OF FLUX DATA SETS                ',I10/
     2   1X,'NUMBER OF TIME STEPS                    ',I10/
     3   1X,'FACTOR FOR NODAL FLUXES                 ',1PE10.2/
     4   1X,'SWITCH FOR ECHO OF INPUT FLUXES         ',I10/
     5   1X,'SWITCH FOR ECHO OF STEP FLUXES          ',I10)
C
C     PUMPAGE DATA SETS
C
      DO 103 J=1,NQSET
      READ(IR1,902) NQ
      DO 110 I=1,NN
      QSET(I,J)=0.0
 110  CONTINUE
      READ(IR1,903) (QNODE(I),QSET(QNODE(I),J),I=1,NQ)
      DO 102 I=1,NN
      QSET(I,J)=QSET(I,J)*FACQ
 102  CONTINUE
      IF (IECHO1.EQ.1) THEN
      WRITE(IW1,904) J
      WRITE(IW1,905) (QNODE(I),QSET(QNODE(I),J),I=1,NQ)
      ENDIF
 103  CONTINUE
C
 902  FORMAT(I10)
 903  FORMAT(I10,F10.0)
 904  FORMAT(/1X,'FLUX SET',I10/1X,18('-')/
     1   1X,6X,'NODE',5X,'QFLUX'/)
 905  FORMAT(1X,I10,1PE10.2)
C
C     TIME-STEP INDICATORS AND MULTIPLIERS
C
      DO 203 I=1,MAXKNS
      READ(IR1,906) (FACSET(I,J),J=1,NQSET)
 203  CONTINUE
      IF (IECHO1.EQ.1) THEN
      WRITE(IW1,907)
      IF (NQSET.LE.5) THEN
      DO 204 I=1,MAXKNS
      WRITE(IW1,908) I,(FACSET(I,J),J=1,NQSET)
 204  CONTINUE
      ELSE
```

```
      DO 205 I=1,MAXKNS
      WRITE(IW1,908) I,(FACSET(I,J),J=1,5)
      WRITE(IW1,9087) (FACSET(I,J),J=6,NQSET)
 205  CONTINUE
      ENDIF
      ENDIF
 906  FORMAT(10X,5F10.0)
 907  FORMAT(/1X,'TIME-STEP INDICATORS AND MULTIPLIERS'/1X,37('-')/
     1   1X,6X,'STEP',3X,'FACTORS'/)
 908  FORMAT((1X,I10,5(1PE10.2)))
 9087 FORMAT((1X,10X,5(1PE10.2)))
      RETURN
C
C     END DATA INPUT
C
C     ----------------------------------------
      ENTRY FLUX2(QFLUX,SUMF,KNS,IOUT,NN)
C     ----------------------------------------
C
C     INITIALIZE VECTOR (QFLUX)
C
      DO 105 I=1,NN
      QFLUX(I)=0.0
 105  CONTINUE
      IF (NQSET.EQ.0) RETURN
C
C     COMPUTE PUMPAGE FOR TIME STEP
C
      DO 104 I=1,NN
      DO 1041 J=1,NQSET
      QFLUX(I)=QFLUX(I)+QSET(I,J)*FACSET(KNS,J)
 1041 CONTINUE
 104  CONTINUE
      RETURN
C
C     ---------------------------------------------------
      ENTRY FLUX3(QFLUX,QPOS,QNEG,SUMF,KNS,IOUT,NN)
C     ---------------------------------------------------
C
      SUMF=0.0
      IF (NQSET.EQ.0) RETURN
      DO 106 I=1,NN
      SUMF=SUMF+QFLUX(I)
      IF (QFLUX(I).GE.0.0) QPOS(I)=QPOS(I)+QFLUX(I)
      IF (QFLUX(I).LT.0.0) QNEG(I)=QNEG(I)+QFLUX(I)
 106  CONTINUE
C
C     DISPLAY FLUXES FOR STEP
C
      IF (IECHO2.EQ.1 .AND. IOUT.EQ.1) THEN
      WRITE(IW1,909) SUMF
      DO 107 I=1,NN
      IF (QFLUX(I).NE.0.0) WRITE(IW1,910) I,QFLUX(I)
 107  CONTINUE
 909  FORMAT(/1X,'SOURCE AND SINK FLUXES'/1X,22('-')/
     1   1X,'TOTAL FLUXES        ',1PE10.2/
     2   1X,6X,'NODE',5X,'QFLUX'/)
 910  FORMAT(1X,I10,1PE10.2)
      ENDIF
      RETURN
      END
```

```
C     -----------------
      SUBROUTINE VFLUX1
C     -----------------
C
      INCLUDE 'PARAM.FOR'
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*2 (I-N)
      INTEGER*2 VFNODE(MAXVFB)
      REAL*8 K(MAXVFB),SS(MAXVFB),HEIGHT(MAXVFB),WIDTH(MAXVFB),
     1   QVF(MAXVFB)
      REAL*8 H(MAXNN),H0(MAXNN),QVFLUX(MAXNN),QVPAST(MAXNN),
     1   CVFRH(MAXNN),CVFLH(MAXNN)
      REAL*8 HSAVE(MAXVFB,MAXSTP),TIME(MAXSTP)
      REAL*8 QPOS(MAXNN),QNEG(MAXNN)
      IR1=11
      IW1=13
C
C     DATA INPUT
C
      WRITE(*,'(1X,''READING IN VFLUX'')')
      READ(IR1,900) NVFB,IECHO1,IECHO2
      WRITE(IW1,901) NVFB,IECHO1,IECHO2
      IF (NVFB.EQ.0)  RETURN
      IF (IECHO1.NE.0) WRITE(IW1,906)
      DO 100 IBX=1,NVFB
      READ(IR1,902) VFNODE(IBX),K(IBX),SS(IBX),WIDTH(IBX),
     1   HEIGHT(IBX)
      IF (IECHO1.NE.0) THEN
      WRITE(IW1,903) VFNODE(IBX),K(IBX),SS(IBX),WIDTH(IBX),
     1   HEIGHT(IBX)
      ENDIF
      K(IBX)=K(IBX)/(24.0*3600.0)
 100  CONTINUE
 900  FORMAT(3I10)
 901  FORMAT(/1X,'VARIABLE FLUX BOUNDARY NODES'/1X,28('-')/
     1   1X,'NUMBER OF VARIABLE FLUX NODES           ',I10/
     2   1X,'SWITCH FOR ECHO OF PARAMETERS           ',I10/
     3   1X,'SWITCH FOR ECHO OF FLUXES               ',I10)
 906  FORMAT(/1X,6X,'NODE',9X,'K',8X,'SS',5X,'WIDTH',4X,'HEIGHT'/)
 902  FORMAT(I10,4F10.0)
 903  FORMAT(1X,I10,0PF10.2,1PE10.2,0PF10.2,0PF10.2)
      RETURN
C
C     ------------------------------------------------------
      ENTRY VFLUX2 (H0,H,QVPAST,CVFLH,CVFRH,DELT,SVFLUX,
     1   NN,KNS,ISS,IOUT)
C     ------------------------------------------------------
C
C     INITIALIZE ARRAYS
C
      DO 200 I=1,NN
      QVFLUX(I)=0.0
      QVPAST(I)=0.0
      CVFLH(I)=0.0
      CVFRH(I)=0.0
 200  CONTINUE
      IF (NVFB.EQ.0) RETURN
      IF(KNS.EQ.1.AND.ISS.EQ.1) RETURN
```

```
C
C     FILL TIME VECTOR
C
      IF (KNS-ISS.EQ.1) TIME(KNS)=DELT
      IF (KNS-ISS.GT.1) TIME(KNS)=TIME(KNS-1)+DELT
C
C     LOOP OVER BOUNDARY NODES
C
      DO 220 IBX=1,NVFB
      II=VFNODE(IBX)
      PI=3.14159
      FACTOR=K(IBX)*HEIGHT(IBX)*WIDTH(IBX)/
     1   SQRT(PI*K(IBX)/SS(IBX))
C
C     EFFECTS OF PAST STEPS
C
      SUMQ=0.0
      NIS=KNS-1
      IF (NIS.LT.1) GO TO 215
      DO 210 IS=1,NIS
      DELH=HSAVE(IBX,IS)-H0(II)
      T1=TIME(KNS-1)
      T2=TIME(KNS)
      IF (IS.GT.1) THEN
      T1=TIME(KNS-1)-TIME(IS-1)
      T2=TIME(KNS)-TIME(IS-1)
      ENDIF
C      T1=DABS(T2-DELT)
      SUMQ=SUMQ-2.0*FACTOR*DELH*(SQRT(T2)-SQRT(T1))/DELT
      T1=TIME(KNS-1)-TIME(IS)
      T2=TIME(KNS)-TIME(IS)
C      T1=DABS(T2-DELT)
      SUMQ=SUMQ+2.0*FACTOR*DELH*(SQRT(T2)-SQRT(T1))/DELT
 210  CONTINUE
      QVPAST(II)=SUMQ
C
C     EFFECTS OF PRESENT STEP
C
 215  CONTINUE
      T1=0.0
      T2=DELT
      CVFRH(II)=2.0*FACTOR*(SQRT(T2)-SQRT(T1))*H0(II)/DELT
      CVFLH(II)=2.0*FACTOR*(SQRT(T2)-SQRT(T1))/DELT
 220  CONTINUE
      RETURN
C
C     -----------------------------------------------
      ENTRY VFLUX3 (H0,H,QPOS,QNEG,QVPAST,CVFLH,CVFRH,
     1   DELT,SVFLUX,NN,KNS,ISS,IOUT)
C     -----------------------------------------------
C
      SVFLUX=0.0
      IF (NVFB.EQ.0) RETURN
      IF (KNS.EQ.1.AND.ISS.EQ.1) RETURN
      DO 300 IBX=1,NVFB
      II=VFNODE(IBX)
      HSAVE(IBX,KNS)=H(II)
      QVF(IBX)=CVFRH(II)-CVFLH(II)*H(II)
      SVFLUX=SVFLUX+QVF(IBX)+QVPAST(II)
      QVFLUX(II)=QVPAST(II)+QVF(IBX)
      IF (QVFLUX(II).GE.0.0) QPOS(II)=QPOS(II)+QVFLUX(II)
      IF (QVFLUX(II).LT.0.0) QNEG(II)=QNEG(II)+QVFLUX(II)
```

```
 300  CONTINUE
C
C     PRINT VARIABLE FLUXES
C
      IF (IECHO2.EQ.1 .AND. IOUT.EQ.1) WRITE(IW1,904) SVFLUX
      DO 301 IBX=1,NVFB
      II=VFNODE(IBX)
      IF (IECHO2.EQ.1 .AND. IOUT.EQ.1)
     1   WRITE(IW1,905) II,QVF(IBX),QVPAST(II),QVFLUX(II)
 301  CONTINUE
 904  FORMAT(/1X,'VARIABLE-FLUX NODES'/1X,19('-')/
     1   1X,'CUMULATIVE RATE      ',1PE10.2//
     2   1X,6X,'NODE',3X,'PRESENT',6X,'PAST',5X,'TOTAL'/)
 905  FORMAT(1X,I10,3(1PE10.2))
      RETURN
      END
```

```
C     -----------------------------
      SUBROUTINE BAND1(G,F,H,NN,NB)
C     -----------------------------
C
      INCLUDE 'PARAM.FOR'
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*2 (I-N)
      REAL*8 G(MAXNN,MAXNB),F(MAXNN)
      REAL*8 H(MAXNN)
C
C     UPPER TRIANGULARIZE MATRIX OF COEFFICIENTS (G)
C
      DO 221 I=1,NN
      IP=NN-I+1
      IF (NB.LT.IP) IP=NB
      DO 220 J=1,IP
      IQ=NB-J
      IF ((I-1).LT.IQ) IQ=I-1
      SUM=G(I,J)
      IF (IQ.LT.1) GO TO 450
      DO 440 K=1,IQ
      II=I-K
      JZ=J+K
      SUM=SUM-G(II,K+1)*G(II,JZ)
 440  CONTINUE
 450  IF (J.NE.1) GO TO 230
      IF (SUM.LE.0.0) THEN
      GO TO 260
      ENDIF
      TEMP=1.0/DSQRT(SUM)
      G(I,J)=TEMP
      GO TO 220
 230  G(I,J)=SUM*TEMP
 220  CONTINUE
 221  CONTINUE
C
      RETURN
 260  WRITE(*,910) I
      STOP
 910  FORMAT(1X,'UPPER TRIANGULARIZATION FAILS AT ROW',I4)
C
C     ------------------------
      ENTRY BAND2(G,F,H,NN,NB)
C     ------------------------
C
C     BACK SUBSTITUTION OF VECTOR (F) INTO UPPER TRIANGULARIZED MATRIX
C
      DO 320 I=1,NN
      J=I-NB+1
      IF ((I+1).LE.NB) J=1
      SUM=F(I)
      K1=I-1
      IF (J.GT.K1) GO TO 340
      DO 330 K=J,K1
      II=I-K+1
      SUM=SUM-G(K,II)*H(K)
 330  CONTINUE
 340  H(I)=SUM*G(I,1)
 320  CONTINUE
```

```
C
      DO 540 I1=1,NN
      I=NN-I1+1
      J=I+NB-1
      IF (J.GT.NN) J=NN
      SUM=H(I)
      K2=I+1
      IF (K2.GT.J) GO TO 250
      DO 550 K=K2,J
      KK=K-I+1
      SUM=SUM-G(I,KK)*H(K)
  550 CONTINUE
  250 H(I)=SUM*G(I,1)
  540 CONTINUE
C
      RETURN
      END
```

```
C     -----------------
      SUBROUTINE SOLVE1
C     -----------------
C
      INCLUDE 'PARAM.FOR'
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*2 (I-N)
      REAL*8 G(MAXNN,MAXNB),F(MAXNN),H(MAXNN)
      INTEGER*2 ICOL(MAXNN,MAXNB),IDAG(MAXNN),NCOL(MAXNN)
      IR1=11
      IW1=13
C
      READ(IR1,902) NITER,OMEGA,CLOSE
      WRITE(IW1,901) NITER,OMEGA,CLOSE
 901  FORMAT(/1X,'MATRIX SOLUTION PARAMETERS'/1X,26('-')/
     2   1X,'MAXIMUM ITERATIONS                  ',I10/
     3   1X,'RELAXATION FACTOR                   ',F10.2/
     4   1X,'CLOSURE CRITERION                   ',1PE10.2)
 902  FORMAT(I10,2F10.0)
      RETURN
C
C     -------------------------------------
      ENTRY SOLVE2(G,F,H,NN,ICOL,IDAG,NCOL)
C     -------------------------------------
C
C     BEGIN ITERATIONS
C
      DO 500 ITER=1,NITER
C
C     LOOP OVER ROWS
C
      DIFMAX=0.0
      DO 400 I=1,NN
      SUM=0.0
      ID=IDAG(I)
      NC=NCOL(I)
      DO 300 IC=1,NC
      IF (IC.EQ.ID) GOTO 300
      JJ=ICOL(I,IC)
      SUM=SUM+G(I,IC)*H(JJ)
 300  CONTINUE
      IF (DABS(SUM).LE.1.0D-20) SUM=0.0
      HLAST=H(I)
      H(I)=H(I)+OMEGA*(-SUM+F(I))/G(I,ID)-OMEGA*H(I)
      DIFF=DABS(H(I)-HLAST)
      DIFMAX=DMAX1(DIFF,DIFMAX)
 400  CONTINUE
      IF (ITER.EQ.1) WRITE(*,920) ITER,DIFMAX
      IF (ITER.NE.1) WRITE(*,921) ITER,DIFMAX
C
C     CHECK FOR CONVERGENCE
```

```
C
      IF(DIFMAX.LE.CLOSE) GOTO 600
 500  CONTINUE
C
      ITER=ITER-1
 600  WRITE(IW1,900) ITER,DIFMAX
 900  FORMAT(/1X,'NUMBER OF ITERATIONS          ',I10/
     1   1X,'MAXIMUM DIFFERENCE           ',1PE10.2)
 920  FORMAT(' ','ITERATION ',I4,1PE10.2)
 921  FORMAT('+','ITERATION ',I4,1PE10.2)
      RETURN
      END
```

```
C     ------------------------------------------------------------
      SUBROUTINE BUDGET(X,Y,Z,IN,IPRISM,ITOP,H,HL,SS,SY,
     1   DELT,NE2,NE,SUMF,SUMP,SUMR,SUMRV,SUMET,SUMCH,SUMVF,SUMFL,
     2   SUMQE,SUMQV,IOUT,FSS,KNS,STIME)
C     ------------------------------------------------------------
C
      INCLUDE 'PARAM.FOR'
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*2 (I-N)
      REAL*8 X(MAXNN),Y(MAXNN),Z(MAXNN),SS(MAXNE2),SY(MAXNE2),
     1   H(MAXNN),HL(MAXNN)
      REAL*8 XE(4),YE(4),ZE(4),DELHE(4)
      INTEGER*2 IN(MAXNE2,4),IPRISM(MAXNE,6),ITOP(MAXNE)
      IW1=13
      IW3=15
C
C     STORAGE CHANGE DUE TO ELASTIC RESPONSE
C
      DELS=0.0
      DO 102 L=1,NE2
C
C     RE-ASSIGN NODAL COORDINATES
C
      DO 100 I=1,4
      II=IN(L,I)
      XE(I)=X(II)
      YE(I)=Y(II)
      ZE(I)=Z(II)
      DELHE(I)=H(II)-HL(II)
  100 CONTINUE
      X1=XE(1)
      X2=XE(2)
      X3=XE(3)
      X4=XE(4)
      Y1=YE(1)
      Y2=YE(2)
      Y3=YE(3)
      Y4=YE(4)
      Z1=ZE(1)
      Z2=ZE(2)
      Z3=ZE(3)
      Z4=ZE(4)
C
C     VOLUME OF ELEMENT
C
      A1=X2*(Y3*Z4-Y4*Z3)-X3*(Y2*Z4-Y4*Z2)+X4*(Y2*Z3-Y3*Z2)
      B1=-Y3*Z4+Y4*Z3+Y2*Z4-Y4*Z2-Y2*Z3+Y3*Z2
      C1=X3*Z4-X4*Z3-X2*Z4+X4*Z2+X2*Z3-X3*Z2
      D1=-X3*Y4+X4*Y3+X2*Y4-X4*Y2-X2*Y3+X3*Y2
      VOL=(A1+B1*X1+C1*Y1+D1*Z1)/6.0
C
C     STORAGE CHANGE IN ELEMENT
C
      SDELH=0.0
      DO 400 I=1,4
      SDELH=SDELH+DELHE(I)
  400 CONTINUE
      WATER=SS(L)*VOL*SDELH*FSS/4.0
      DELS=DELS+WATER
  102 CONTINUE
C
C     STORAGE CHANGE DUE TO WATER-TABLE RESPONSE
```

```
C
      DO 202 L=1,NE
      IF (ITOP(L).EQ.0) GO TO 202
      DO 200 I=1,3
      II=IPRISM(L,I)
      XE(I)=X(II)
      YE(I)=Y(II)
      DELHE(I)=H(II)-HL(II)
 200  CONTINUE
C
C     RE-ASSIGN NODAL COORDINATES
C
      X1=XE(1)
      X2=XE(2)
      X3=XE(3)
      Y1=YE(1)
      Y2=YE(2)
      Y3=YE(3)
C
C      AREA OF ELEMENT
C
      AREA=(X2*Y3-X3*Y2-X1*Y3+X3*Y1+X1*Y2-X2*Y1)/2.0
C
C     STORAGE CHANGE IN ELEMENT
C
      SDELH=0.0
      DO 500 I=1,3
      SDELH=SDELH+DELHE(I)
 500  CONTINUE
      WATER=SY(L)*AREA*SDELH*FSS/3.0
      DELS=DELS+WATER
 202  CONTINUE
      DELS=DELS/DELT
C
C     WATER-BUDGET RESIDUAL (RESIDUAL-OUT POSITIVE)
C
      RESID=SUMF+SUMP+SUMR+SUMRV+SUMET+SUMCH+SUMVF+SUMFL-
     1   SUMQE-SUMQV-DELS
C
C     DISPLAY WATER BUDGET
C
      WRITE(IW3,901) KNS,STIME,SUMF,SUMP,SUMR,SUMCH,SUMET,SUMRV,SUMVF,
     1   SUMFL,SUMQE,SUMQV,DELS,RESID
      IF (IOUT.EQ.0) RETURN
      WRITE(IW1,900) SUMF,SUMP,SUMR,SUMCH,SUMET,SUMRV,SUMVF,
     1   SUMFL,SUMQE,SUMQV,DELS,RESID
 900  FORMAT(/1X,'GROUND-WATER BUDGET'/1X,19('-')/
     1   1X,'SOURCE-SINK NODES                        ',1PE10.2/
     2   1X,'PUMPING                                  ',1PE10.2/
     3   1X,'RECHARGE                                 ',1PE10.2/
     2   1X,'CONSTANT-HEAD NODES                      ',1PE10.2/
     3   1X,'EVAPOTRANSPIRATION                       ',1PE10.2/
     4   1X,'STREAM-AQUIFER INTERACTION               ',1PE10.2/
     5   1X,'VARIABLE-FLUX BOUNDARIES                 ',1PE10.2/
     6   1X,'FAULT-NODE PAIRS                         ',1PE10.2/
     7   1X,'ELASTIC STORAGE CHANGE IN FINE BEDS      ',1PE10.2/
     8   1X,'INELASTIC STORAGE CHANGE IN FINE BEDS    ',1PE10.2/
     9   1X,'ELASTIC STORAGE CHANGE IN COARSE BEDS    ',1PE10.2/
     1   1X,'RESIDUAL                                 ',1PE10.2)
 901  FORMAT(I10,13(1PE10.2))
      RETURN
      END
```

```
C     ---------------------------------------------------
      SUBROUTINE SHAPE1(KXX,KYY,KZZ,SS,XE,YE,ZE,AE,BE,L)
C     ---------------------------------------------------
C
      INCLUDE 'PARAM.FOR'
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*2 (I-N)
      REAL*8 KXX,KYY,KZZ,XE(4),YE(4),ZE(4),AE(4,4),BE(4,4)
      REAL*8 B(4),C(4),D(4)
C
C     VOLUME INTEGRATIONS
C     RE-ASSIGN NODAL COORDINATES
C
      X1=XE(1)
      X2=XE(2)
      X3=XE(3)
      X4=XE(4)
      Y1=YE(1)
      Y2=YE(2)
      Y3=YE(3)
      Y4=YE(4)
      Z1=ZE(1)
      Z2=ZE(2)
      Z3=ZE(3)
      Z4=ZE(4)
C
C     COFACTORS OF THE VOLUME DETERMINATE
C
      A1=X2*(Y3*Z4-Y4*Z3)-X3*(Y2*Z4-Y4*Z2)+X4*(Y2*Z3-Y3*Z2)
      B(1)=-Y3*Z4+Y4*Z3+Y2*Z4-Y4*Z2-Y2*Z3+Y3*Z2
      B(2)=+Y3*Z4-Y4*Z3-Y1*Z4+Y4*Z1+Y1*Z3-Y3*Z1
      B(3)=-Y2*Z4+Y4*Z2+Y1*Z4-Y4*Z1-Y1*Z2+Y2*Z1
      B(4)=+Y2*Z3-Y3*Z2-Y1*Z3+Y3*Z1+Y1*Z2-Y2*Z1
      C(1)=X3*Z4-X4*Z3-X2*Z4+X4*Z2+X2*Z3-X3*Z2
      C(2)=-X3*Z4+X4*Z3+X1*Z4-X4*Z1-X1*Z3+X3*Z1
      C(3)=X2*Z4-X4*Z2-X1*Z4+X4*Z1+X1*Z2-X2*Z1
      C(4)=-X2*Z3+X3*Z2+X1*Z3-X3*Z1-X1*Z2+X2*Z1
      D(1)=-X3*Y4+X4*Y3+X2*Y4-X4*Y2-X2*Y3+X3*Y2
      D(2)=+X3*Y4-X4*Y3-X1*Y4+X4*Y1+X1*Y3-X3*Y1
      D(3)=-X2*Y4+X4*Y2+X1*Y4-X4*Y1-X1*Y2+X2*Y1
      D(4)=+X2*Y3-X3*Y2-X1*Y3+X3*Y1+X1*Y2-X2*Y1
      VOL=(A1+B(1)*X1+C(1)*Y1+D(1)*Z1)/6.0
      IF (VOL.LE.0.0) WRITE(*,'(1X,''NEGATIVE VOLUME FOR'',I10)') L
C
C     ELEMENT STIFFNESS MATRIX (AE) AND DYNAMIC MATRIX (BE)
C

      DO 201 I=1,4
      DO 200 J=1,4
      AE(I,J)=(KXX*B(I)*B(J)+KYY*C(I)*C(J)+KZZ*D(I)*D(J))/(36.0*VOL)
      BE(I,J)=SS*VOL/20.0
      IF (I.EQ.J) BE(I,J)=BE(I,J)*2.0
 200  CONTINUE
 201  CONTINUE
      RETURN
C
C     ---------------------------
      ENTRY SHAPE2(SY,XE,YE,BE,L)
C     ---------------------------
```

```
C
C     AREA INTEGRATIONS
C     RE-ASSIGN NODAL COORDINATES
C
      X1=XE(1)
      X2=XE(2)
      X3=XE(3)
      Y1=YE(1)
      Y2=YE(2)
      Y3=YE(3)
C
C     AREA OF ELEMENT
C
      AREA=(X2*Y3-X3*Y2-X1*Y3+X3*Y1+X1*Y2-X2*Y1)/2.0
      IF (AREA.LE.0.0) WRITE(*,'(1X,''NEGATIVE VOLUME FOR'',I10)') L
C
C     ELEMENT DYNAMIC MATRIX (BE)
C
      DO 506 I=1,3
      DO 505 J=1,3
      BE(I,J)=SY*AREA/12.0
      IF (I.EQ.J) BE(I,J)=BE(I,J)*2.0
 505  CONTINUE
 506  CONTINUE
      RETURN
      END
```

```
C     ---------------------
      SUBROUTINE WATER1(NN)
C     ---------------------
C
      INCLUDE 'PARAM.FOR'
      INCLUDE 'PARAM2.FOR'
      IMPLICIT INTEGER*2 (I-N)
      REAL*8 PNODE(MAXSTP,MAXNN),RNODE(MAXSTP,MAXNN),SUMP,SUMR
      REAL*4 PUSER(MAXUSE),DUSER(MAXUSE),ETUSER(MAXUSE),RUSER(MAXUSE)
      INTEGER*2 WSTAT(MAXWEL,MAXWST),WNODE(MAXWEL,MAXWND),
     1   BSMO(MAXWEL,MAXWST),BSYR(MAXWEL,MAXWST),ESMO(MAXWEL,MAXWST),
     2   ESYR(MAXWEL,MAXWST),NSTAT(MAXWEL),
     3   NWNODE(MAXWEL)
      INTEGER*2 XWSTAT,XBSMO,XBSYR,XESMO,XESYR
      CHARACTER*60 FILE2,FILE3,FILE4,FILE13,FILE14,FILE15,
     1   FILE16,XFILE(MAXFLS)
      CHARACTER*252 LINE,BUFFER
      CHARACTER*12 WNUM(MAXWEL),XWNUM,WUSER(MAXWEL,MAXWST),XWUSER,
     1   EWNUM(MAXWEL)
      REAL WFACT(MAXWEL,MAXWND),WCAP(MAXWEL)
      INTEGER*2 XPDMO,XPDYR,PDMO(MAXWSQ),PDYR(MAXWSQ),
     1   NQPUMP(MAXWEL),POUT(MAXWEL)
      REAL QPUMP(MAXWSQ)
      INTEGER*2 UTYPE(MAXUSE),BAMO(MAXUSE,MAXUST),
     1   BAYR(MAXUSE,MAXUST),EAMO(MAXUSE,MAXUST),EAYR(MAXUSE,MAXUST),
     2   UNODE(MAXUSE,MAXUST,MAXUND),NUNODE(MAXUSE,MAXUST),
     3   NAREA(MAXUSE)
      CHARACTER*60 FILE5,FILE6,FILE7,FILE8,FILE9,FILE10,FILE11,
     1   FILE12,FILE17,FILE18,FILE19,FILE20,FILE21,FILE22,FILE23,
     2   FILE24,FILE25,FILE26,FILE27,FILE28,FILE29,FILE30,FILE31
      CHARACTER*8 UNUM(MAXUSE),XUNUM
      REAL AREA(MAXUSE,MAXUST),UFACT(MAXUSE,MAXUST,MAXUND),
     1   RFACT(MAXUSE),RDEEP(MAXNN),TAILF(MAXUSE),THETA0(MAXUSE)
      INTEGER*2 XDDMO,XDDYR,DDMO(MAXUSQ),DDYR(MAXUSQ),
     1   NQDEL(MAXUSE)
      REAL QDEL(MAXUSQ),RDEL(MAXCON,MAXDCE)
      INTEGER*2 PCMO(MAXCON),PCYR(MAXCON),PSCODE(MAXCON,MAXPCE),
     1   PDFILE(MAXCON,MAXPCE),PUMO(MAXCON,MAXPCE),PUYR(MAXCON,MAXPCE),
     2   NPUMP(MAXCON)
      REAL PFACT(MAXCON,MAXPCE)
      REAL RPUMP(MAXCON,MAXPCE),AWC(MAXUSE),
     1   GAMMA(MAXUSE),PORDEL(MAXUSE,12),UCOEF(MAXUSE),
     2   DFACT(MAXCON,MAXDCE)
      INTEGER*2 WGET(MAXWEL),DGET(MAXUSE),AGET(MAXUSE),SGET(MAXWEL)
      INTEGER*2 DCMO(MAXCON),DCYR(MAXCON),NDEL(MAXCON),
     1   DDFILE(MAXCON,MAXDCE),DUMO(MAXCON,MAXDCE),
     2   DUYR(MAXCON,MAXDCE),UTCODE(MAXCON,MAXDCE),
     3   ETCODE(MAXCON,MAXDCE)
      INTEGER*2 XYR,XCODE
      INTEGER*2 INYR(MAXUSE,MAXREC),INCODE(MAXUSE,MAXREC),
     1   NREC(MAXUSE),CCODE(MAXCRP),NPET(MAXCRP),CCMO(MAXCON),
     2   CCYR(MAXCON),CUYR(MAXCON,MAXDCE),EUMO(MAXCON,MAXDCE),
     3   EUYR(MAXCON,MAXDCE),NCREC(MAXCON)
      REAL INPOR(MAXUSE,MAXREC),ROOT(MAXCRP),RAIN(MAXCON),XRAIN(12),
     1   XPET(12),PET(MAXCRP,MAXCON),CLOSS(MAXUSE),CPOR(MAXUSE,MAXCRP),
     2   DEPTH(MAXUSE)
      INTEGER*4 WSTART(MAXWEL),USTART(MAXUSE),ISEQ,ISEQ1,ISEQ2,
     1   I1,I2,J1,J2
      INTEGER*2 PINDX(MAXPCE),DINDX(MAXUST),RINDX(MAXUST)
      REAL P(MAXCON,MAXPCE),PU(MAXCON,MAXPCE),DU(MAXCON,MAXUST),
     1   RU(MAXCON,MAXUST),EU(MAXCON,MAXUST)
C
```

```
C     INPUT AND OUTPUT FILES
C        WATER.DAT
C        WELL.DAT
C        WSTAT.DAT
C        PUMP.DAT
C        CPUMP.DAT
C        USER.DAT
C        DELIVER.DAT
C        CDELIVER.DAT
C        CROP.DAT
C        ROOT.DAT
C        PRECIP.DAT
C        PET.DAT
C        CCROP.DAT
C        HARDPAN.DAT
C        DESTIN.DAT
C        EXCLUDE.DAT
C        WELL.OUT
C        PUMP.OUT
C        CPUMP.OUT
C        USER.OUT
C        DELIVER.OUT
C        CDELIVER.OUT
C        CROP.OUT
C        ROOT.OUT
C        PRECIP.OUT
C        PET.OUT
C        CCROP.OUT
C        HARDPAN.OUT
C        DESTIN.OUT
C        EXCLUDE.OUT
C
      IR0=10
      IR1=11
      IR2=21
      IR3=22
      IR4=23
      IR5=24
      IR6=25
      IR7=26
      IR8=27
      IR9=28
      IR10=29
      IR11=30
      IR12=31
      IR13=32
      IR14=46
      IR15=48
      IR16=49
      IW1=33
      IW2=34
      IW3=35
      IW4=36
      IW5=37
      IW6=38
      IW7=39
      IW8=40
      IW9=41
      IW10=42
      IW11=43
      IW12=44
      IW13=45
```

```
      IW14=47
      IW15=13
      IW16=50
      WRITE(*,'(1X,''READING IN WATER'')')
      READ(IR1,920) ICONT
      WRITE(IW15,9202) ICONT
 9202 FORMAT(/1X,'WATER PARAMETERS'/1X,16('-')/
     1   1X,'SWITCH FOR WATER                                  ',I10)
      IF (ICONT.EQ.0) RETURN
      WRITE(*,'(1X,''  CONCATENATING FILES'')')
C
C     INPUT WELL INVENTORY
C
      READ(IR0,9190) FILE2,NFILE
      IF (NFILE.GT.0) THEN
      READ(IR0,9191) (XFILE(I),I=1,NFILE)
      CALL FMERGE(XFILE,FILE2,NFILE)
      ENDIF
C
C     INPUT WELL STATUS
C
      READ(IR0,9190) FILE29,NFILE
      IF (NFILE.GT.0) THEN
      READ(IR0,9191) (XFILE(I),I=1,NFILE)
      CALL FMERGE(XFILE,FILE29,NFILE)
      ENDIF
C
C     INPUT MONTHLY PUMPING
C
      READ(IR0,9190) FILE3,NFILE
      IF (NFILE.GT.0) THEN
      READ(IR0,9191) (XFILE(I),I=1,NFILE)
      CALL FMERGE(XFILE,FILE3,NFILE)
      ENDIF
C
C     INPUT PUMPING CONSTRUCTION
C
      READ(IR0,9190) FILE4,NFILE
      IF (NFILE.GT.0) THEN
      READ(IR0,9191) (XFILE(I),I=1,NFILE)
      CALL FMERGE(XFILE,FILE4,NFILE)
      ENDIF
C
C     INPUT USER INVENTORY
C
      READ(IR0,9190) FILE5,NFILE
      IF (NFILE.GT.0) THEN
      READ(IR0,9191) (XFILE(I),I=1,NFILE)
      CALL FMERGE(XFILE,FILE5,NFILE)
      ENDIF
C
C     INPUT MONTHLY DELIVERIES
C
      READ(IR0,9190) FILE6,NFILE
      IF (NFILE.GT.0) THEN
      READ(IR0,9191) (XFILE(I),I=1,NFILE)
      CALL FMERGE(XFILE,FILE6,NFILE)
      ENDIF
C
```

```
C      INPUT DELIVERY CONSTRUCTION
C
       READ(IR0,9190) FILE7,NFILE
       IF (NFILE.GT.0) THEN
       READ(IR0,9191) (XFILE(I),I=1,NFILE)
       CALL FMERGE(XFILE,FILE7,NFILE)
       ENDIF
C
C      INPUT CROP INVENTORY
C
       READ(IR0,9190) FILE8,NFILE
       IF (NFILE.GT.0) THEN
       READ(IR0,9191) (XFILE(I),I=1,NFILE)
       CALL FMERGE(XFILE,FILE8,NFILE)
       ENDIF
C
C      INPUT ROOTING DEPTHS
C
       READ(IR0,9190) FILE9,NFILE
       IF (NFILE.GT.0) THEN
       READ(IR0,9191) (XFILE(I),I=1,NFILE)
       CALL FMERGE(XFILE,FILE9,NFILE)
       ENDIF
C
C      INPUT MONTHLY PRECIPITATION
C
       READ(IR0,9190) FILE10,NFILE
       IF (NFILE.GT.0) THEN
       READ(IR0,9191) (XFILE(I),I=1,NFILE)
       CALL FMERGE(XFILE,FILE10,NFILE)
       ENDIF
C
C      INPUT MONTHLY PET
C
       READ(IR0,9190) FILE11,NFILE
       IF (NFILE.GT.0) THEN
       READ(IR0,9191) (XFILE(I),I=1,NFILE)
       CALL FMERGE(XFILE,FILE11,NFILE)
       ENDIF
C
C      INPUT CROP AND WEATHER CONSTRUCTION
C
       READ(IR0,9190) FILE12,NFILE
       IF (NFILE.GT.0) THEN
       READ(IR0,9191) (XFILE(I),I=1,NFILE)
       CALL FMERGE(XFILE,FILE12,NFILE)
       ENDIF
C
C      INPUT NODAL HARDPAN
C
       READ(IR0,9190) FILE25,NFILE
       IF (NFILE.GT.0) THEN
       READ(IR0,9191) (XFILE(I),I=1,NFILE)
       CALL FMERGE(XFILE,FILE25,NFILE)
       ENDIF
C
C      INPUT PUMPING DESTINATION
```

```
C
      READ(IR0,9190) FILE27,NFILE
      IF (NFILE.GT.0) THEN
      READ(IR0,9191) (XFILE(I),I=1,NFILE)
      CALL FMERGE(XFILE,FILE27,NFILE)
      ENDIF
C
C     INPUT WELL EXCLUSIONS
C
      READ(IR0,9190) FILE30,NFILE
      IF (NFILE.GT.0) THEN
      READ(IR0,9191) (XFILE(I),I=1,NFILE)
      CALL FMERGE(XFILE,FILE30,NFILE)
      ENDIF
C
C     OUTPUT FILES
C
      READ(IR0,9191) FILE13
      READ(IR0,9191) FILE14
      READ(IR0,9191) FILE15
      READ(IR0,9191) FILE16
      READ(IR0,9191) FILE17
      READ(IR0,9191) FILE18
      READ(IR0,9191) FILE19
      READ(IR0,9191) FILE20
      READ(IR0,9191) FILE21
      READ(IR0,9191) FILE22
      READ(IR0,9191) FILE23
      READ(IR0,9191) FILE24
      READ(IR0,9191) FILE26
      READ(IR0,9191) FILE28
      READ(IR0,9191) FILE31
 9190 FORMAT(A60,I10)
 9191 FORMAT(A60)
C
C     READ FILES AND ECHO SWITCHES FOR WELL INVENTORY FILE,
C        MONTHLY PUMPING FILE, AND PUMPING CONSTRUCTION FILE
C     PROGRAM CONTROL FILE
C
      OPEN (UNIT=IW1,FILE=FILE13,STATUS='UNKNOWN')
      READ(IR1,921) IOUT1,IOUT2,IOUT3,IOUT4,IOUT5
      READ(IR1,921) IOUT6,IOUT7,IOUT8,IOUT9,IOUT10
      READ(IR1,921) IOUT11,IOUT12,IOUT13,IOUT14
      WRITE(IW1,922) ICONT,IOUT1,IOUT2,IOUT3,IOUT4,IOUT5,IOUT6,
     1   IOUT7,IOUT8,IOUT9,IOUT10,IOUT11,IOUT12,IOUT13,IOUT14
 920  FORMAT(I10)
 921  FORMAT(6I10)
 922  FORMAT(1X,'PROGRAM CONTROLS'/1X,16('-')/
     1   1X,'SWITCH FOR CONTROL OF SUBROUTINE                    ',I10/
     2   1X,'SWITCH FOR OUTPUT OF WELL INVENTORY                 ',I10/
     3   1X,'SWITCH FOR OUTPUT OF MONTHLY PUMPING                ',I10/
     4   1X,'SWITCH FOR OUTPUT OF PUMPING CONSTRUCTION           ',I10/
     5   1X,'SWITCH FOR OUTPUT OF USER INVENTORY                 ',I10/
     6   1X,'SWITCH FOR OUTPUT OF MONTHLY DELIVERIES             ',I10/
     7   1X,'SWITCH FOR OUTPUT OF DELIVERY CONSTRUCTION          ',I10/
     8   1X,'SWITCH FOR OUTPUT OF CROP INVENTORY                 ',I10/
     9   1X,'SWITCH FOR OUTPUT OF ROOTING DEPTH                  ',I10/
     1   1X,'SWITCH FOR OUTPUT OF MONTHLY PRECIPITATION          ',I10/
     2   1X,'SWITCH FOR OUTPUT OF MONTHLY EVAPOTRANSPIRATION     ',I10/
     3   1X,'SWITCH FOR OUTPUT OF CROP AND WEATHER CONSTRUCTION',I10/
     4   1X,'SWITCH FOR OUTPUT OF RECHARGE FACTORS               ',I10/
     5   1X,'SWITCH FOR OUTPUT OF PUMPING DESTINATIONS           ',I10/
```

```
     6   1X,'SWITCH FOR OUTPUT OF WELL EXCLUSIONS                  ',I10//
     7   1X,'INPUT ERRORS'/1X,12('-')/)
C
C     INPUT OF WELL INVENTORY FILE
C
      NPINDX=0
      NDINDX=0
      NRINDX=0
      IWELL=0
      WRITE(*,'(1X,''  WELL INVENTORY ''/5X,A60)') FILE2
      OPEN (UNIT=IR2,FILE=FILE2,STATUS='OLD')
 100  READ(IR2,924,END=102) LINE
C
      IF (LINE(1:1).EQ.'*') GO TO 100
      IF (LINE(1:1).EQ.'X') GO TO 100
      IF (LINE(1:1).EQ.'Y') GO TO 100
C
      IF (LINE(1:1).EQ.'A') THEN
      IWELL=IWELL+1
      WRITE(BUFFER,925) LINE
      READ(BUFFER,926) WNUM(IWELL)
      INODE=0
      ISTAT=0
      GO TO 100
      ENDIF
C
      IF (LINE(1:1).EQ.'B') THEN
      INODE=INODE+1
      WRITE(BUFFER,925) LINE
      READ(BUFFER,927) WNODE(IWELL,INODE),WFACT(IWELL,INODE)
      NWNODE(IWELL)=INODE
      GO TO 100
      ENDIF
C
      IF (LINE(1:1).EQ.'D') THEN
      WRITE(BUFFER,925) LINE
      READ(BUFFER,946) WCAP(IWELL)
      GO TO 100
      ENDIF
C
      WRITE(IW1,929) WNUM(IWELL)
      GO TO 100
 102  CONTINUE
      NWELL=IWELL
      CLOSE (IR2)
C
C     INPUT WELL STATUS
C
      WRITE(*,'(1X,''  WELL STATUS ''/5X,A60)') FILE29
      OPEN (UNIT=IR15,FILE=FILE29,STATUS='OLD')
      DO 1133 IWELL=1,NWELL
      NSTAT(IWELL)=0
 1133 CONTINUE
 1100 READ(IR15,924,END=1102) LINE
C     IF (LINE(1:1).EQ.'*') GOTO 1100
C
      IF (LINE(1:1).EQ.'C') THEN
      WRITE(BUFFER,925) LINE
      READ(BUFFER,928) XWNUM,XWSTAT,XBSMO,XBSYR,XESMO,XESYR,XWUSER
      DO 1143 IWELL=1,NWELL
      IF (WNUM(IWELL).EQ.XWNUM) GOTO 1144
```

```
 1143 CONTINUE
      WRITE(IW1,9299) XWNUM
      GOTO 1100
 1144 CONTINUE
      ISTAT=NSTAT(IWELL)+1
      NSTAT(IWELL)=ISTAT
      WSTAT(IWELL,ISTAT)=XWSTAT
      BSMO(IWELL,ISTAT)=XBSMO
      BSYR(IWELL,ISTAT)=XBSYR
      ESMO(IWELL,ISTAT)=XESMO
      ESYR(IWELL,ISTAT)=XESYR
      WUSER(IWELL,ISTAT)=XWUSER
      GO TO 1100
      ENDIF
      GOTO 1100
 1102 CONTINUE
      CLOSE (IR15)
C
 924  FORMAT(A252)
 925  FORMAT(A252)
 926  FORMAT(2X,A12)
 927  FORMAT(21X,I9,1X,F9.0)
 928  FORMAT(2X,A12,7X,I9,4I5,2X,A8)
 929  FORMAT(1X,'ERROR IN RECORD TYPE FOR WELL INVENTORY AFTER WELL '
     1   ,A12)
 9299 FORMAT(1X,'WELL NOT IN INVENTORY FOR WELL STATUS ',A12)
 946  FORMAT(21X,F9.0)
C
C     ADJUST NODE FACTORS TO UNITY
C
      DO 5254 IWELL=1,NWELL
      INODEX=NWNODE(IWELL)
      SWFACT=0.0
      DO 5252 INODE=1,INODEX
      SWFACT=SWFACT+WFACT(IWELL,INODE)
 5252 CONTINUE
      DO 5529 INODE=1,INODEX
      WFACT(IWELL,INODE)=WFACT(IWELL,INODE)/SWFACT
 5529 CONTINUE
 5254 CONTINUE
C
C     OUTPUT OF WELL INVENTORY FILE
C
      IF (IOUT1.NE.1) GO TO 255
      OPEN (UNIT=IW2,FILE=FILE14,STATUS='UNKNOWN')
      WRITE(IW2,990)
      DO 254 IWELL=1,NWELL
      WRITE(IW2,976)
      WRITE(IW2,977) IWELL,WNUM(IWELL),WCAP(IWELL)
      WRITE(IW2,972)
      INODEX=NWNODE(IWELL)
      SWFACT=0.0
      DO 252 INODE=1,INODEX
      SWFACT=SWFACT+WFACT(IWELL,INODE)
      WRITE(IW2,973) INODE,WNODE(IWELL,INODE),WFACT(IWELL,INODE)
 252  CONTINUE
      DO 2529 INODE=1,INODEX
      WFACT(IWELL,INODE)=WFACT(IWELL,INODE)/SWFACT
```

```
 2529 CONTINUE
      WRITE(IW2,974)
      ISTATX=NSTAT(IWELL)
      IF (ISTATX.EQ.0) WRITE(IW1,9748) WNUM(IWELL)
      DO 253 ISTAT=1,ISTATX
      WRITE(IW2,975) ISTAT,WSTAT(IWELL,ISTAT),BSMO(IWELL,ISTAT),
     1   BSYR(IWELL,ISTAT),ESMO(IWELL,ISTAT),ESYR(IWELL,ISTAT),
     2   WUSER(IWELL,ISTAT)
 253  CONTINUE
 254  CONTINUE
      CLOSE (IW2)
 255  CONTINUE
C
 990  FORMAT(1X,'WELL INVENTORY FILE'/1X,19('-'))
 976  FORMAT(/1X,5X,'IWELL',2X,'WNUM',11X,'WCAP'/
     1   1X,5X,5('-'),2X,4('-'),11X,4('-'))
 977  FORMAT(1X,I10,2X,A12,F10.2)
 972  FORMAT(/1X,5X,'INODE',5X,'WNODE',5X,'WFACT'/
     1   1X,5X,5('-'),6X,4('-'),5X,5('-'))
 973  FORMAT(1X,2I10,F10.4)
 974  FORMAT(/1X,5X,'ISTAT',5X,'WSTAT',6X,'BSMO',6X,'BSYR',
     1   6X,'ESMO',6X,'ESYR',2X,'WUSER'/
     1   1X,5X,5('-'),5X,5('-'),2(6X,4('-'),6X,4('-')),2X,5('-'))
 9748 FORMAT(1X,'NO STATUS FOR WELL ',A8)
 975  FORMAT(1X,6I10,2X,A8)
C
C     PUMPING DESTINATION FILE
C
      WRITE(*,'(1X,''  PUMPING DESTINATION ''/5X,A60)') FILE27
      OPEN (UNIT=IR14,FILE=FILE27,STATUS='OLD')
      IF (IOUT13.EQ.1) OPEN (UNIT=IW14,FILE=FILE28,STATUS='UNKNOWN')
      READ(IR14,9387) NPOUT
      IF (NPOUT.NE.0) THEN
      READ(IR14,9388) (POUT(IPOUT),IPOUT=1,NPOUT)
      IF (IOUT13.EQ.1) THEN
      WRITE(IW14,9389)
      WRITE(IW14,9390) (POUT(IPOUT),IPOUT=1,NPOUT)
      ENDIF
      ENDIF
      CLOSE (IR14)
      CLOSE (IW14)
 9387 FORMAT(I10)
 9388 FORMAT(I10)
 9389 FORMAT(1X,'PUMPING DESTINATION TO CANALS'/1X,29('-')//
     1   1X,4X,'PSCODE'/)
 9390 FORMAT(1X,I10)
C
C     PUMPING EXCLUSION FILE
C
      WRITE(*,'(1X,''  PUMPING EXCLUSION ''/5X,A60)') FILE30
      OPEN (UNIT=IR16,FILE=FILE30,STATUS='OLD')
      IF (IOUT14.EQ.1) OPEN (UNIT=IW16,FILE=FILE31,STATUS='UNKNOWN')
      READ(IR16,9487) NEXC
      IF (NEXC.NE.0) THEN
      READ(IR16,9488) (EWNUM(IEXC),IEXC=1,NEXC)
      IF (IOUT14.EQ.1) THEN
      WRITE(IW16,9489)
      WRITE(IW16,9490) (EWNUM(IEXC),IEXC=1,NEXC)
      ENDIF
      ENDIF
```

```
      CLOSE (IR16)
      CLOSE (IW16)
 9487 FORMAT(I10)
 9488 FORMAT(A12)
 9489 FORMAT(1X,'PUMPING EXCLUSIONS'/1X,14('-')//
     1   1X,2X,'EWNUM'/)
 9490 FORMAT(2X,A12)
C
C     INPUT OF MONTHLY PUMPING
C
      ISEQ=0
      DO 112 IWELL=1,NWELL
      NQPUMP(IWELL)=0
      WSTART(IWELL)=0
 112  CONTINUE
      WRITE(*,'(1X,''  MONTHLY PUMPING ''/5X,A60)') FILE3
      OPEN (UNIT=IR3,FILE=FILE3,STATUS='OLD')
 104  READ(IR3,936,END=110) LINE
C
      IF (LINE(1:1).EQ.'*') GO TO 104
C
      IF (LINE(1:1).EQ.'F') THEN
      WRITE(BUFFER,937) LINE
      READ(BUFFER,930) XWNUM,XPDMO,XPDYR,XQPUMP
      DO 105 I=1,NWELL
      IF (XWNUM.EQ.WNUM(I)) GO TO 106
 105  CONTINUE
      WRITE(IW1,931) XWNUM
      GO TO 104
 106  CONTINUE
      IWELL=I
      ISEQ=ISEQ+1
      IF (WSTART(IWELL).EQ.0) WSTART(IWELL)=ISEQ
      NQPUMP(IWELL)=NQPUMP(IWELL)+1
      PDMO(ISEQ)=XPDMO
      PDYR(ISEQ)=XPDYR
      QPUMP(ISEQ)=XQPUMP
      ENDIF
      GO TO 104
C
 110  CONTINUE
      CLOSE (IR3)
C
 936  FORMAT(A252)
 937  FORMAT(A252)
 930  FORMAT(2X,A12,6X,2I5,F10.0)
 931  FORMAT(1X,'WELL ',A12,' NOT IN INVENTORY FILE')
C
C     OUTPUT OF MONTHLY PUMPING
C
      IF (IOUT2.NE.1) GO TO 245
      OPEN (UNIT=IW3,FILE=FILE15,STATUS='UNKNOWN')
      WRITE(IW3,991)
      DO 244 IWELL=1,NWELL
      IF (NQPUMP(IWELL).EQ.0) GO TO 244
      WRITE(IW3,978)
      WRITE(IW3,979) IWELL,WNUM(IWELL)
      WRITE(IW3,968)
      ISEQ1=WSTART(IWELL)
      ISEQ2=ISEQ1+NQPUMP(IWELL)-1
      DO 243 ISEQ=ISEQ1,ISEQ2
      WRITE(IW3,967) ISEQ,PDMO(ISEQ),PDYR(ISEQ),QPUMP(ISEQ)
```

```
 243   CONTINUE
 244   CONTINUE
       CLOSE (IW3)
 245   CONTINUE
C
 991   FORMAT(1X,'MONTHLY PUMPING FILE'/1X,20('-'))
 978   FORMAT(/1X,5X,'IWELL',6X,'WNUM'/
     1    1X,5X,5('-'),6X,4('-'))
 979   FORMAT(1X,I10,6X,A12)
 968   FORMAT(/1X,6X,'ISEQ',6X,'PDMO',6X,'PDYR',5X,'QPUMP'/
     1    1X,6X,4('-'),6X,4('-'),6X,4('-'),5X,5('-'))
 967   FORMAT(1X,3I10,F10.2)
C
C      INPUT OF PUMPING CONSTRUCTION
C
       ICMO=0
       WRITE(*,'(1X,''  PUMPING CONSTRUCTION ''/5X,A60)') FILE4
       OPEN (UNIT=IR4,FILE=FILE4,STATUS='OLD')
 400   READ(IR4,916,END=410) LINE
C
       IF (LINE(1:1).EQ.'*') GO TO 400
C
       IF (LINE(1:1).EQ.'G') THEN
       IPUMP=0
       ICMO=ICMO+1
       WRITE(BUFFER,917) LINE
       READ(BUFFER,918) PCMO(ICMO),PCYR(ICMO)
       GO TO 400
       ENDIF
C
       IF (LINE(1:1).EQ.'H') THEN
       IPUMP=IPUMP+1
       WRITE(BUFFER,917) LINE
       READ(BUFFER,919) PSCODE(ICMO,IPUMP),PDFILE(ICMO,IPUMP),
     1    RPUMP(ICMO,IPUMP),PUMO(ICMO,IPUMP),PUYR(ICMO,IPUMP),
     2    PFACT(ICMO,IPUMP)
       NPUMP(ICMO)=IPUMP
       GO TO 400
       ENDIF
C
       WRITE(IW1,932) PCMO(ICMO),PCYR(ICMO)
       GO TO 400
 410   CONTINUE
       NCMO=ICMO
       CLOSE (IR4)
C
 916   FORMAT(A252)
 917   FORMAT(A252)
 918   FORMAT(1X,I4,I5)
 919   FORMAT(2X,I8,1X,I9,1X,F9.0,I5,I5,1X,F9.0)
 932   FORMAT(1X,'ERROR IN RECORD TYPE FOR PUMPING',
     1    ' CONSTRUCTION AFTER',I2,1X,I4)
C
C      OUTPUT OF PUMPING CONSTRUCTION
```

```
C
      IF (IOUT3.NE.1) GO TO 422
      OPEN (UNIT=IW4,FILE=FILE16,STATUS='UNKNOWN')
      WRITE(IW4,992)
      DO 420 ICMO=1,NCMO
      WRITE(IW4,933) ICMO,PCMO(ICMO),PCYR(ICMO)
      WRITE(IW4,934)
      IPUMPX=NPUMP(ICMO)
      DO 416 IPUMP=1,IPUMPX
      WRITE(IW4,935) IPUMP,PSCODE(ICMO,IPUMP),
     1    PDFILE(ICMO,IPUMP),RPUMP(ICMO,IPUMP),
     2    PUMO(ICMO,IPUMP),PUYR(ICMO,IPUMP),PFACT(ICMO,IPUMP)
 416      CONTINUE
 420  CONTINUE
      CLOSE (IW4)
 422  CONTINUE
C
 992  FORMAT(1X,'PUMPING CONSTRUCTION FILE'/1X,25('-'))
 933  FORMAT(/1X,6X,'ICMO',6X,'PCMO',6X,'PCYR'/
     1    1X,5X,5('-'),6X,4('-'),6X,4('-')/
     2    1X,3I10)
 934  FORMAT(/1X,5X,'IPUMP',4X,'PSCODE',4X,'PDFILE',5X,'RPUMP',
     1    6X,'PUMO',6X,'PUYR',5X,'PFACT'/
     2    1X,5X,5('-'),4X,6('-'),4X,6('-'),5X,5('-'),
     3    6X,4('-'),6X,4('-'),5X,5('-'))
 935  FORMAT(1X,3I10,F10.2,2I10,F10.2)
C
C     READ FILES AND ECHO SWITCHES FOR USER INVENTORY FILE,
C        MONTHLY DELIVERY FILE, AND DELIVERY CONSTRUCTION FILE
C
C
C     INPUT OF HARDPAN FACTORS FOR RECHRAGE FROM USER AREAS
C
      WRITE(*,'(1X,''  HARDPAN ''/5X,A60)') FILE25
      OPEN (UNIT=IR13,FILE=FILE25,STATUS='OLD')
      IF (IOUT12.EQ.1) OPEN (UNIT=IW13,FILE=FILE26,STATUS='UNKNOWN')
      READ(IR13,945) (RDEEP(I),I=1,NN)
      IF (IOUT12.EQ.1) THEN
      WRITE(IW13,9451)
      WRITE(IW13,9452) (RDEEP(I),I=1,NN)
      ENDIF
      CLOSE (IR13)
      CLOSE (IW13)
 945  FORMAT(5F10.0)
 9451 FORMAT(1X,'HARDPAN FACTORS'/1X,15('-')/)
 9452 FORMAT(1X,5F10.3)
C
C     INPUT OF USER INVENTORY
C
      DO 198 IUSER=1,MAXUSE
      DO 197 IMO=1,12
      PORDEL(IUSER,IMO)=1.0
 197  CONTINUE
 198  CONTINUE
      IUSER=0
      WRITE(*,'(1X,''  USER INVENTORY ''/5X,A60)') FILE5
      OPEN (UNIT=IR5,FILE=FILE5,STATUS='OLD')
```

```
200   READ(IR5,960,END=202) LINE
C
       IF (LINE(1:1).EQ.'*') GO TO 200
C
       IF (LINE(1:1).EQ.'I') THEN
       IUSER=IUSER+1
       IAREA=0
       WRITE(BUFFER,961) LINE
       READ(BUFFER,962) UNUM(IUSER),UTYPE(IUSER)
       GO TO 200
       ENDIF
C
       IF (LINE(1:1).EQ.'Z') THEN
       WRITE(BUFFER,961) LINE
       READ(BUFFER,9629) IMO,PORDEL(IUSER,IMO)
       GO TO 200
       ENDIF
C
       IF (LINE(1:1).EQ.'R') THEN
       WRITE(BUFFER,961) LINE
       READ(BUFFER,995) AWC(IUSER),GAMMA(IUSER),UCOEF(IUSER),
     1   THETA0(IUSER),RFACT(IUSER),CLOSS(IUSER),TAILF(IUSER)
       GO TO 200
       ENDIF
C
       IF (LINE(1:1).EQ.'J') THEN
       IAREA=IAREA+1
       WRITE(BUFFER,961) LINE
       READ(BUFFER,963) AREA(IUSER,IAREA),BAMO(IUSER,IAREA),
     1   BAYR(IUSER,IAREA),EAMO(IUSER,IAREA),EAYR(IUSER,IAREA)
       NAREA(IUSER)=IAREA
       INODE=0
       GO TO 200
       ENDIF
C
       IF (LINE(1:1).EQ.'K') THEN
       INODE=INODE+1
       WRITE(BUFFER,961) LINE
       READ(BUFFER,964) UNODE(IUSER,IAREA,INODE),
     1   UFACT(IUSER,IAREA,INODE)
       NUNODE(IUSER,IAREA)=INODE
       GO TO 200
       ENDIF
C
       WRITE(IW1,965) UNUM(IUSER)
       GO TO 200
 202   CONTINUE
       NUSER=IUSER
       CLOSE (IR5)
C
       DO 2027 IWELL=1,NWELL
       NSTATX=NSTAT(IWELL)
       DO 2028 ISTAT=1,NSTATX
       I1=12*BSYR(IWELL,ISTAT)+BSMO(IWELL,ISTAT)
       I2=12*ESYR(IWELL,ISTAT)+ESYR(IWELL,ISTAT)
       IF (I2.EQ.0) I2=100000
       DO 2029 IUSER=1,NUSER
       NAREAX=NAREA(IUSER)
       J1=12*BAYR(IUSER,1)+BAMO(IUSER,1)
       J2=12*EAYR(IUSER,NAREAX)+EAMO(IUSER,NAREAX)
       IF (J2.EQ.0) J2=100000
       IF ((J1.GE.I1 .AND. J1.LE.I2) .OR. (J2.GE.I1 .AND. J2.LE.I2)
```

```
     1   .OR. (J1.LE.I1 .AND. J2.GE.I2)) THEN
      IF (WUSER(IWELL,ISTAT).EQ.'        ') GO TO 2028
      IF (WUSER(IWELL,ISTAT).EQ.UNUM(IUSER)) GO TO 2028
      ENDIF
 2029 CONTINUE
      WRITE(IW1,9969) WUSER(IWELL,ISTAT),WNUM(IWELL)
 2028 CONTINUE
 2027 CONTINUE
C
 9969 FORMAT(1X,'USER ',A8,' FOR WELL ',A8,' NOT IN USER INVENTORY')
 960  FORMAT(A252)
 961  FORMAT(A252)
 962  FORMAT(2X,A8,1X,I9)
 9629 FORMAT(11X,I9,1X,F9.0)
 963  FORMAT(11X,F9.0,I5,I5,I5,I5)
 964  FORMAT(11X,I9,1X,F9.0)
 995  FORMAT(11X,F9.0,1X,F9.0,1X,F9.0,1X,F9.0,1X,F9.0,1X,F9.0,1X,F9.0)
 965  FORMAT(1X,'ERROR IN RECORD TYPE AFTER USER',A8)
C
C     ADJUST NODE FACTORS TO UNITY
C
      DO 5313 IUSER=1,NUSER
      IAREAX=NAREA(IUSER)
      DO 5312 IAREA=1,IAREAX
      INODEX=NUNODE(IUSER,IAREA)
      SUFACT=0.0
      DO 5310 INODE=1,INODEX
      SUFACT=SUFACT+UFACT(IUSER,IAREA,INODE)
 5310 CONTINUE
      DO 5109 INODE=1,INODEX
      UFACT(IUSER,IAREA,INODE)=UFACT(IUSER,IAREA,INODE)/SUFACT
 5109 CONTINUE
 5312 CONTINUE
 5313 CONTINUE
C
C     OUTPUT OF USER INVENTORY
C
      IF (IOUT4.NE.1) GO TO 314
      OPEN (UNIT=IW5,FILE=FILE17,STATUS='UNKNOWN')
      WRITE(IW5,993)
      DO 313 IUSER=1,NUSER
      WRITE(IW5,910)
      WRITE(IW5,911) IUSER,UNUM(IUSER)
      WRITE(IW5,9217)
      WRITE(IW5,9218) (IMO,PORDEL(IUSER,IMO),IMO=1,12)
      WRITE(IW5,9110) UTYPE(IUSER),AWC(IUSER),GAMMA(IUSER),
     1   UCOEF(IUSER),THETA0(IUSER),CLOSS(IUSER),TAILF(IUSER),
     2   RFACT(IUSER)
      IAREAX=NAREA(IUSER)
      DO 312 IAREA=1,IAREAX
      WRITE(IW5,912)
      WRITE(IW5,913) IAREA,AREA(IUSER,IAREA),BAMO(IUSER,IAREA),
     1   BAYR(IUSER,IAREA),EAMO(IUSER,IAREA),EAYR(IUSER,IAREA)
      WRITE(IW5,914)
      INODEX=NUNODE(IUSER,IAREA)
      SUFACT=0.0
      DO 310 INODE=1,INODEX
      SUFACT=SUFACT+UFACT(IUSER,IAREA,INODE)
      WRITE(IW5,915) INODE,UNODE(IUSER,IAREA,INODE),
     1   UFACT(IUSER,IAREA,INODE)
```

```
 310  CONTINUE
      DO 3109 INODE=1,INODEX
      UFACT(IUSER,IAREA,INODE)=UFACT(IUSER,IAREA,INODE)/SUFACT
 3109 CONTINUE
 312  CONTINUE
 313  CONTINUE
      CLOSE (IW5)
 314  CONTINUE
 993  FORMAT(1X,'USER INVENTORY FILE'/1X,19('-'))
 910  FORMAT(/1X,5X,'IUSER',6X,'UNUM'/
     1   1X,5X,5('-'),6X,4('-'))
 911  FORMAT(1X,I10,6X,A8)
 9217 FORMAT(/1X,'DELIVERY PROPORTION'/1X,19('-')//
     1   1X,7X,'IMO',4X,'PORDEL')
 9218 FORMAT(1X,I10,F10.2)
 9110 FORMAT(/1X,5X,'UTYPE',7X,'AWC',5X,'GAMMA',
     1   5X,'UCOEF',4X,'THETA0',5X,'CLOSS',5X,'TAILF'/
     2   1X,5X,5('-'),7X,3('-'),5X,5('-'),5X,5('-'),
     3   4X,6('-'),5X,5('-'),5X,5('-')/1X,I10,6F10.2//
     4   1X,5X,'RFACT'/1X,5X,5('-')/1X,F10.2)
 912  FORMAT(/1X,5X,'IAREA',6X,'AREA',6X,'BAMO',6X,'BAYR',
     1   6X,'EAMO',6X,'EAYR'/1X,5X,5('-'),6X,4('-'),6X,
     2   4('-'),6X,4('-'),6X,4('-'),6X,4('-'))
 913  FORMAT(1X,I10,F10.2,4I10)
 914  FORMAT(/1X,5X,'INODE',5X,'UNODE',5X,'UFACT'/
     1   1X,5X,5('-'),5X,5('-'),5X,5('-'))
 915  FORMAT(1X,2I10,F10.4)
C
C     INPUT OF MONTHLY DELIVERIES
C
      ISEQ=0
      DO 113 IUSER=1,NUSER
      NQDEL(IUSER)=0
      USTART(IUSER)=0
 113  CONTINUE
      WRITE(*,'(1X,''  MONTHLY DELIVERIES ''/5X,A60)') FILE6
      OPEN (UNIT=IR6,FILE=FILE6,STATUS='OLD')
 204  READ(IR6,955,END=210) LINE
C
      IF (LINE(1:1).EQ.'*') GO TO 204
C
      IF (LINE(1:1).EQ.'Q') THEN
      WRITE(BUFFER,954) LINE
      READ(BUFFER,970) XUNUM,XDDMO,XDDYR,XQDEL
      DO 206 I=1,NUSER
      IF (XUNUM.EQ.UNUM(I)) GO TO 2066
 206  CONTINUE
      WRITE(IW1,971) XUNUM
      GO TO 204
 2066 CONTINUE
      IUSER=I
      NQDEL(IUSER)=NQDEL(IUSER)+1
      ISEQ=ISEQ+1
      IF (USTART(IUSER).EQ.0) USTART(IUSER)=ISEQ
      DDMO(ISEQ)=XDDMO
      DDYR(ISEQ)=XDDYR
      QDEL(ISEQ)=XQDEL
      ENDIF
      GO TO 204
```

```
C
 210   CONTINUE
       CLOSE (IR6)
C
 954   FORMAT(A252)
 955   FORMAT(A252)
 970   FORMAT(2X,A8,I5,I5,F10.0)
 971   FORMAT(1X,'USER',A8,' NOT IN INVENTORY FILE')
C
C      OUTPUT OF MONTHLY DELIVERIES
C
       IF (IOUT5.NE.1) GO TO 345
       OPEN (UNIT=IW6,FILE=FILE18,STATUS='UNKNOWN')
       WRITE(IW6,994)
       DO 344 IUSER=1,NUSER
       IF (NQDEL(IUSER).EQ.0) GO TO 344
       WRITE(IW6,908)
       WRITE(IW6,909) IUSER,UNUM(IUSER)
       WRITE(IW6,906)
       ISEQ1=USTART(IUSER)
       ISEQ2=ISEQ1+NQDEL(IUSER)-1
       DO 343 ISEQ=ISEQ1,ISEQ2
       WRITE(IW6,907) ISEQ,DDMO(ISEQ),DDYR(ISEQ),QDEL(ISEQ)
 343   CONTINUE
 344   CONTINUE
       CLOSE (IW6)
 345   CONTINUE
C
 994   FORMAT(1X,'MONTHLY DELIVERIES FILE'/1X,23('-'))
 908   FORMAT(/1X,5X,'IUSER',6X,'UNUM'/
     1    1X,5X,5('-'),6X,4('-'))
 909   FORMAT(1X,I10,6X,A8)
 906   FORMAT(/1X,6X,'ISEQ',6X,'DDMO',6X,'DDYR',6X,'QDEL'/
     1    1X,5X,5('-'),6X,4('-'),6X,4('-'),6X,4('-'))
 907   FORMAT(1X,3I10,F10.2)
C
C      INPUT OF DELIVERY CONSTRUCTION
C
       ICMO=0
       WRITE(*,'(1X,''  DELIVERY CONSTRUCTION ''/5X,A60)') FILE7
       OPEN (UNIT=IR7,FILE=FILE7,STATUS='OLD')
 320   READ(IR7,9561,END=321) LINE
C
       IF (LINE(1:1).EQ.'*') GO TO 320
C
       IF (LINE(1:1).EQ.'M') THEN
       IDEL=0
       ICMO=ICMO+1
       WRITE(BUFFER,9561) LINE
       READ(BUFFER,9581) DCMO(ICMO),DCYR(ICMO)
       IF (DCMO(ICMO).NE.PCMO(ICMO) .OR.
     1   DCYR(ICMO).NE.PCYR(ICMO)) WRITE(IW1,9888) ICMO
       GO TO 320
       ENDIF
C
       IF (LINE(1:1).EQ.'N') THEN
       IDEL=IDEL+1
       WRITE(BUFFER,9561) LINE
       READ(BUFFER,959) UTCODE(ICMO,IDEL),DDFILE(ICMO,IDEL),
     1   RDEL(ICMO,IDEL),DUMO(ICMO,IDEL),DUYR(ICMO,IDEL),
     2   DFACT(ICMO,IDEL)
       NDEL(ICMO)=IDEL
```

```
      GO TO 320
      ENDIF
C
 321  CONTINUE
      IF (ICMO.NE.NCMO) WRITE(IW1,9989)
      CLOSE (IR7)
C
 9561 FORMAT(A252)
 9581 FORMAT(1X,I4,I5)
 959  FORMAT(2X,I8,1X,I9,1X,F9.0,I5,I5,1X,F9.0)
 9888 FORMAT(1X,'MONTH OR YEAR FOR DELIVERY CONSTRUCTION'/
     1    4X,'DOES NOT MATCH PUMPING CONSTRUCTION AT MONTH',I9)
 9989 FORMAT(1X,'TOTAL MONTHS FOR DELIVERY CONSTRUCTION'/
     1    4X,'DOES NOT MATCH PUMPING CONSTRUCTION')
C
C     OUTPUT OF DELIVERY CONSTRUCTION
C
      IF (IOUT6.NE.1) GO TO 322
      OPEN (UNIT=IW7,FILE=FILE19,STATUS='UNKNOWN')
      WRITE(IW7,996)
      DO 324 ICMO=1,NCMO
      WRITE(IW7,997) ICMO,DCMO(ICMO),DCYR(ICMO)
      WRITE(IW7,998)
      IDELX=NDEL(ICMO)
      DO 323 IDEL=1,IDELX
      WRITE(IW7,999) IDEL,UTCODE(ICMO,IDEL),DDFILE(ICMO,IDEL),
     1    RDEL(ICMO,IDEL),DUMO(ICMO,IDEL),DUYR(ICMO,IDEL),
     2    DFACT(ICMO,IDEL)
 323  CONTINUE
 324  CONTINUE
      CLOSE (IW7)
 322  CONTINUE
C
 996  FORMAT(1X,'DELIVERY CONSTRUCTION FILE'/1X,26('-'))
 997  FORMAT(1X,6X,'ICMO',6X,'DCMO',6X,'DCYR'/
     1    1X,6X,4('-'),6X,4('-'),6X,4('-')/
     2    1X,3I10)
 998  FORMAT(1X,6X,'IDEL',4X,'UTCODE',4X,'DDFILE',6X,'RDEL',
     1    6X,'DUMO',6X,'DUYR',5X,'DFACT'/
     2    1X,6X,4('-'),4X,6('-'),4X,6('-'),6X,4('-'),
     3    6X,4('-'),6X,4('-'),5X,5('-'))
 999  FORMAT(1X,3I10,F10.1,2I10,F10.2)
C
C     INPUT CROP INVENTORY
C
      DO 560 IUSER=1,NUSER
      NREC(IUSER)=0
 560  CONTINUE
      WRITE(*,'(1X,''  CROP INVENTORY ''/5X,A60)') FILE8
      OPEN (UNIT=IR8,FILE=FILE8,STATUS='OLD')
 506  READ(IR8,956,END=507) LINE
C
      IF (LINE(1:1).EQ.'*') GO TO 506
C
      IF (LINE(1:1).EQ.'P') THEN
      WRITE(BUFFER,956) LINE
      READ(BUFFER,957) XUNUM,XYR,XCODE,XPOR
      DO 561 IUSER=1,NUSER
      IF (UNUM(IUSER).EQ.XUNUM) THEN
      NREC(IUSER)=NREC(IUSER)+1
      IREC=NREC(IUSER)
      INYR(IUSER,IREC)=XYR
```

```
      INCODE(IUSER,IREC)=XCODE
      INPOR(IUSER,IREC)=XPOR
      GO TO 506
      ENDIF
  561 CONTINUE
      WRITE(IW1,9000) XUNUM
      GO TO 506
      ENDIF
C
      WRITE(IW1,958)
      GO TO 506
  507 CONTINUE
      CLOSE (IR8)
C
  956 FORMAT(A252)
  957 FORMAT(2X,A8,I10,1X,I9,21X,F9.0)
  958 FORMAT(1X,'ERROR IN RECORD TYPE IN CROP INVENTORY')
 9000 FORMAT(1X,'USER ',A8,' NOT FOUND IN USER INVENTORY')
C
C     OUTPUT CROP INVENTORY
C
      IF (IOUT7.NE.1) GO TO 511
      OPEN (UNIT=IW8,FILE=FILE20,STATUS='UNKNOWN')
      WRITE(IW8,9892)
      DO 509 IUSER=1,NUSER
      WRITE(IW8,9891) IUSER,UNUM(IUSER)
      IRECX=NREC(IUSER)
      IF (IRECX.EQ.0) WRITE(IW1,9001) UNUM(IUSER)
      DO 510 IREC=1,IRECX
      WRITE(IW8,969) IREC,INYR(IUSER,IREC),INCODE(IUSER,IREC),
     1   INPOR(IUSER,IREC)
  510 CONTINUE
  509 CONTINUE
      CLOSE (IW8)
  511 CONTINUE
C
 9892 FORMAT(1X,'CROP INVENTORY'/1X,19('-'))
 9891 FORMAT(1X,5X,'IUSER',6X,'UNUM'/
     1   1X,5X,5('-'),6X,4('-')/
     2   1X,I10,6X,A8/
     3   1X,6X,'IREC',6X,'INYR',4X,'INCODE',5X,'INPOR'/
     4   1X,6X,4('-'),4X,6('-'),5X,5('-'),5X,5('-'))
  969 FORMAT(1X,3I10,F10.2)
 9001 FORMAT(1X,'NO CROP INVENTORY FOR USER ',A8)
C
C     INPUT OF ROOTING DEPTHS
C
      ICROP=0
      WRITE(*,'(1X,''  ROOTING DEPTHS ''/5X,A60)') FILE9
      OPEN (UNIT=IR9,FILE=FILE9,STATUS='OLD')
  512 READ(IR9,9100,END=513) LINE
C
      IF (LINE(1:1).EQ.'*') GO TO 512
C
      IF (LINE(1:1).EQ.'W') THEN
      ICROP=ICROP+1
      WRITE(BUFFER,9100) LINE
      READ(BUFFER,9102) CCODE(ICROP),ROOT(ICROP)
      NCROP=ICROP
      GO TO 512
      ENDIF
```

```
C
      WRITE(IW1,9103)
      GO TO 512
 513  CONTINUE
      CLOSE (IR9)
C
 9100 FORMAT(A252)
 9102 FORMAT(2X,I8,20X,F10.0)
 9103 FORMAT(1X,'ERROR IN RECORD TYPE FOR ROOTING DEPTH')
C
C     OUTPUT OF ROOTING DEPTHS
C
      IF (IOUT8.NE.1) GO TO 519
      OPEN (UNIT=IW9,FILE=FILE21,STATUS='UNKNOWN')
      WRITE(IW9,9105)
      DO 515 ICROP=1,NCROP
      WRITE(IW9,9106) ICROP,CCODE(ICROP),ROOT(ICROP)
 515  CONTINUE
      CLOSE (IW9)
 519  CONTINUE
C
 9105 FORMAT(1X,'ROOTING DEPTH FILE'/1X,18('-')//
     1   1X,5X,'ICROP',5X,'CCODE',6X,'ROOT'/
     1   1X,5X,5('-'),5X,5('-'),6X,4('-')/)
 9106 FORMAT(1X,2I10,F10.2)
C
C     INPUT OF MONTHLY PRECIPITATION
C
      DO 531 ICMO=1,NCMO
      RAIN(ICMO)=-9.99
 531  CONTINUE
      ICMO=0
      WRITE(*,'(1X,''  MONTHLY CONSTRUCTION ''/5X,A60)') FILE10
      OPEN (UNIT=IR10,FILE=FILE10,STATUS='OLD')
 530  READ(IR10,9120,END=535) LINE
C
      IF (LINE(1:1).EQ.'*') GO TO 530
C
      IF (LINE(1:1).EQ.'U') THEN
      WRITE(BUFFER,9120) LINE
      READ(BUFFER,9111) XYR
C
      IF (XYR.EQ.PCYR(1)) THEN
      READ(BUFFER,9112) (XRAIN(I),I=1,12)
      I1=PCMO(1)
      I2=12
      DO 528 I=I1,I2
      ICMO=ICMO+1
      RAIN(ICMO)=XRAIN(I)
 528  CONTINUE
      GO TO 530
      ENDIF
C
      IF (XYR.EQ.PCYR(NCMO)) THEN
      READ(BUFFER,9112) (XRAIN(I),I=1,12)
      I1=1
      I2=PCMO(NCMO)
      DO 529 I=I1,I2
      ICMO=ICMO+1
      RAIN(ICMO)=XRAIN(I)
```

```
  529   CONTINUE
        GO TO 530
        ENDIF
C
        IF (XYR.GT.PCYR(1) .AND. XYR.LT.PCYR(NCMO)) THEN
        READ(BUFFER,9112) (XRAIN(I),I=1,12)
        DO 5301 I=1,12
        ICMO=ICMO+1
        RAIN(ICMO)=XRAIN(I)
 5301   CONTINUE
        GO TO 530
        ENDIF
        GO TO 530
        ENDIF
        WRITE(IW1,9130)
        GO TO 530
C
  535   CONTINUE
        CLOSE (IR10)
C
 9120   FORMAT(A252)
 9111   FORMAT(1X,I8)
 9112   FORMAT(9X,12F6.0)
 9130   FORMAT(1X,'ERROR IN RECORD TYPE FOR PRECIPITATION')
C
C       OUTPUT OF MONTHLY PRECIPITATION
C
        IF (IOUT9.NE.1) GO TO 566
        OPEN (UNIT=IW10,FILE=FILE22,STATUS='UNKNOWN')
        WRITE(IW10,9113) PCMO(1),PCYR(1),PCMO(NCMO),PCYR(NCMO)
        WRITE(IW10,9114) (RAIN(ICMO),ICMO=1,NCMO)
        CLOSE (IW10)
  566   CONTINUE
 9113   FORMAT(/1X,'MONTHLY PRECIPITATION'/1X,21('-')/
     1     1X,'BEGIN DATE ',I2,1X,I4/
     2     1X,'END DATE   ',I2,1X,I4/)
 9114   FORMAT(1X,12F6.2)
C
C       INPUT OF MONTHLY EVAPOTRANSPIRATION
C
        DO 537 ICROP=1,NCROP
        NPET(ICROP)=0
        DO 532 ICMO=1,NCMO
        PET(ICROP,ICMO)=-9.99
  532   CONTINUE
  537   CONTINUE
        WRITE(*,'(1X,''  MONTHLY EVAPOTRANSPIRATION ''/5X,A60)') FILE11
        OPEN (UNIT=IR11,FILE=FILE11,STATUS='OLD')
  539   READ(IR11,9116,END=545) LINE
C
        IF (LINE(1:1).EQ.'*') GO TO 539
C
        IF (LINE(1:1).EQ.'V') THEN
        WRITE(BUFFER,9116) LINE
        READ(BUFFER,9117) XCODE,XYR
        DO 536 I=1,NCROP
        IF (XCODE.EQ.CCODE(I)) THEN
        ICROP=I
        GO TO 538
        ENDIF
```

```
 536   CONTINUE
       WRITE(IW1,9186) XCODE,XYR
       GOTO 539
 538   ICMO=NPET(ICROP)
C
       IF (XYR.EQ.PCYR(1)) THEN
       READ(BUFFER,9118) (XPET(I),I=1,12)
       I1=PCMO(1)
       I2=12
       DO 540 I=I1,I2
       ICMO=ICMO+1
       NPET(ICROP)=ICMO
       PET(ICROP,ICMO)=XPET(I)
 540   CONTINUE
       GO TO 539
       ENDIF
C
       IF (XYR.EQ.PCYR(NCMO)) THEN
       READ(BUFFER,9118) (XPET(I),I=1,12)
       I1=1
       I2=PCMO(NCMO)
       DO 541 I=I1,I2
       ICMO=ICMO+1
       NPET(ICROP)=ICMO
       PET(ICROP,ICMO)=XPET(I)
 541   CONTINUE
       GO TO 539
       ENDIF
C
       IF (XYR.GT.PCYR(1) .AND. XYR.LT.PCYR(NCMO)) THEN
       READ(BUFFER,9118) (XPET(I),I=1,12)
       DO 542 I=1,12
       ICMO=ICMO+1
       NPET(ICROP)=ICMO
       PET(ICROP,ICMO)=XPET(I)
 542   CONTINUE
       GO TO 539
       ENDIF
       GO TO 539
       ENDIF
       WRITE(IW1,9121)
       GO TO 539
C
 545   CONTINUE
       CLOSE (IR11)
C
 9116  FORMAT(A252)
 9117  FORMAT(1X,I3,I5)
 9118  FORMAT(9X,12F6.0)
 9121  FORMAT(1X,'ERROR IN RECORD TYPE FOR PET')
 9186  FORMAT(1X,'CROP',I3,' IN YEAR',I5,' NOT FOUND')
C
C      OUTPUT OF MONTHLY EVAPOTRANSPIRATION
C
       IF (IOUT10.NE.1) GO TO 567
       OPEN (UNIT=IW11,FILE=FILE23,STATUS='UNKNOWN')
       DO 568 ICROP=1,NCROP
       WRITE(IW11,9131) CCODE(ICROP),PCMO(1),PCYR(1),PCMO(NCMO),
     1   PCYR(NCMO)
       WRITE(IW11,9115) (PET(ICROP,ICMO),ICMO=1,NCMO)
```

```
 568  CONTINUE
      CLOSE (IW11)
 567  CONTINUE
 9131 FORMAT(/1X,'MONTHLY POTENTIAL EVAPOTRANSPIRATION'/1X,36('-')/
     1   1X,'CROP CODE  ',I3/
     2   1X,'BEGIN DATE ',I2,1X,I4/
     3   1X,'END DATE   ',I2,1X,I4/)
 9115 FORMAT(1X,12F6.2)
C
C     INPUT CROP CONSTRUCTION FILE
C
      DO 588 ICMO=1,NCMO
      NCREC(ICMO)=0
 588  CONTINUE
      ICMO=0
      WRITE(*,'(1X,''  CROP CONSTRUCTION ''/5X,A60)') FILE12
      OPEN (UNIT=IR12,FILE=FILE12,STATUS='OLD')
 500  READ(IR12,904,END=501) LINE
C
      IF (LINE(1:1).EQ.'*') GO TO 500
C
      IF (LINE(1:1).EQ.'S') THEN
      ICREC=0
      ICMO=ICMO+1
      WRITE(BUFFER,904) LINE
      READ(BUFFER,905) CCMO(ICMO),CCYR(ICMO)
      GO TO 500
      ENDIF
C
      IF (LINE(1:1).EQ.'T') THEN
      ICREC=ICREC+1
      WRITE(BUFFER,904) LINE
      READ(BUFFER,938) ETCODE(ICMO,ICREC),CUYR(ICMO,ICREC),
     1   EUMO(ICMO,ICREC),EUYR(ICMO,ICREC)
      NCREC(ICMO)=ICREC
      GO TO 500
      ENDIF
C
      WRITE(IW1,939)
      GO TO 500
 501  CONTINUE
      CLOSE (IR12)
C
 904  FORMAT(A252)
 905  FORMAT(1X,I4,I5)
 938  FORMAT(2X,I8,I10,I5,I5)
 939  FORMAT(1X,'ERROR IN RECORD TYPE FOR CROP CONSTRUCTION')
C
C     OUTPUT CROP CONSTRUCTION
C
      IF (IOUT11.NE.1) GO TO 502
      OPEN (UNIT=IW12,FILE=FILE24,STATUS='UNKNOWN')
      WRITE(IW12,947)
      DO 504 ICMO=1,NCMO
      WRITE(IW12,948) ICMO,CCMO(ICMO),CCYR(ICMO)
      ICRECX=NCREC(ICMO)
      DO 503 ICREC=1,ICRECX
      WRITE(IW12,949) ICREC,ETCODE(ICMO,ICREC),CUYR(ICMO,ICREC),
     1   EUMO(ICMO,ICREC),EUYR(ICMO,ICREC)
```

```
 503   CONTINUE
 504   CONTINUE
       CLOSE (IW12)
 502   CONTINUE
C
C      INITIALIZE OUTPUT ARRAYS
C
       DO 1553 ICMO=1,NCMO
       DO 1551 IPUMP=1,MAXPCE
       PU(ICMO,IPUMP)=0.0
       P(ICMO,IPUMP)=0.0
 1551  CONTINUE
       DO 1552 IDEL=1,MAXUST
       DU(ICMO,IDEL)=0.0
       RU(ICMO,IDEL)=0.0
       EU(ICMO,IDEL)=0.0
 1552  CONTINUE
 1553  CONTINUE
       RETURN
C
 947   FORMAT(1X,'CROP CONSTRUCTION FILE'/1X,22('-'))
 948   FORMAT(1X,6X,'ICMO',6X,'CCMO',6X,'CCYR'/
      1   1X,6X,4('-'),6X,4('-'),6X,4('-')/
      2   1X,3I10/
      3   1X,5X,'ICREC',4X,'ETCODE',6X,'CUYR',6X,'EUMO',6X,'EUYR'/
      4   1X,5X,5('-'),4X,6('-'),6X,4('-'),6X,4('-'),6X,4('-'))
 949   FORMAT(1X,5I10)
C
C
C      ----------------------------------------------------
       ENTRY WATER2(KNS,MAXKNS,ICMO1,ICMO2,ISS,PNODE,RNODE,
      1   SUMP,SUMR,IOUT,NN)
C      ----------------------------------------------------
C
C      INITIALIZE PUMPING AND RECHARGE FOR MODEL STEP
C
       IF (ICONT.EQ.0) RETURN
       DO 155 INODE=1,NN
       PNODE(KNS,INODE)=0.0
       RNODE(KNS,INODE)=0.0
 155   CONTINUE
       SUMR=0.0
       SUMP=0.0
C
C      BEGIN LOOP OVER SUPPLY STEP FOR MODEL STEP
C
       DO 590 ICMO=ICMO1,ICMO2
C
C      INITIALIZE PUMPING TO USER FOR SUPPLY STEP
C
       DO 156 IUSER=1,NUSER
       PUSER(IUSER)=0.0
 156   CONTINUE
C
C      PROCESS PUMPING CONSTRUCTION TO DETERMINE THE PUMPING
C         FROM WELLS, THE DISTRIBUTION OF PUMPING TO
C         MODEL NODES, AND THE PRIVATE PUMPING TO USERS
```

```
C
      IPUMPX=NPUMP(ICMO)
      DO 146 IPUMP=1,IPUMPX
      XPU=0.0
      XP=0.0
C
C     DISTRIBUTE TOTAL PUMPING TO NODES BASE ON WELL CAPACITY
C
      IF (PDFILE(ICMO,IPUMP).EQ.0) THEN
C
C     FIND WELLS WITH SPECIFED STATUS
C
      IGET=0
      NGET=0
      DO 470 IWELL=1,NWELL
      ISTATX=NSTAT(IWELL)
      DO 478 ISTAT=1,ISTATX
      IF (WSTAT(IWELL,ISTAT).NE.
     1   PSCODE(ICMO,IPUMP)) GO TO 478
      I1=12*(BSYR(IWELL,ISTAT)-PCYR(1))+BSMO(IWELL,ISTAT)-PCMO(1)+1
      I2=10000
      IF (ESYR(IWELL,ISTAT).NE.0)
     1   I2=12*(ESYR(IWELL,ISTAT)-PCYR(1))+ESMO(IWELL,ISTAT)-PCMO(1)+1
      II=ICMO
      IF (PUMO(ICMO,IPUMP).NE.0)
     1   II=12*(PUYR(ICMO,IPUMP)-PCYR(1))+PUMO(ICMO,IPUMP)-PCMO(1)+1
      IF (II.GE.I1 .AND. II.LE.I2) THEN
      IGET=IGET+1
      WGET(IGET)=IWELL
      SGET(IGET)=ISTAT
      NGET=IGET
      GO TO 470
      ENDIF
  478 CONTINUE
  470 CONTINUE
      IF (NGET.EQ.0) GO TO 149
C
C     FIND TOTAL CAPACITY OF WELLS WITH SPECIFIED STATUS
C
      SWCAP=0.0
      DO 472 IGET=1,NGET
      IWELL=WGET(IGET)
      SWCAP=SWCAP+WCAP(IWELL)
  472 CONTINUE
C
C     ASSIGN PUMPING TO WELLS, DISTRIBUTE PUMPING TO NODES,
C        AND ASSIGN PUMPING TO USERS
C
      DO 474 IGET=1,NGET
      IWELL=WGET(IGET)
      ISTAT=SGET(IGET)
      QWELL=RPUMP(ICMO,IPUMP)*PFACT(ICMO,IPUMP)*
     1   WCAP(IWELL)/SWCAP
      XP=XP+QWELL
      INODEX=NWNODE(IWELL)
      DO 473 I=1,INODEX
      INODE=WNODE(IWELL,I)
      PNODE(KNS,INODE)=PNODE(KNS,INODE)-QWELL*WFACT(IWELL,I)
      SUMP=SUMP-QWELL*WFACT(IWELL,I)
```

```
 473  CONTINUE
      DO 169 IUSER=1,NUSER
      IF (UNUM(IUSER).EQ.WUSER(IWELL,ISTAT)) THEN
      JJ=PSCODE(ICMO,IPUMP)
      DO 4733 IPOUT=1,NPOUT
      IF (JJ.EQ.POUT(IPOUT)) GOTO 474
 4733 CONTINUE
      IMO=PCMO(ICMO)
      PUSER(IUSER)=PUSER(IUSER)+QWELL*PORDEL(IUSER,IMO)
      XPU=XPU+QWELL*PORDEL(IUSER,IMO)
      GO TO 474
      ENDIF
 169  CONTINUE
 474  CONTINUE
      GO TO 149
      ENDIF
C
C     FETCH PUMPING FOR INDIVIDUAL WELLS FROM MONTHLY PUMPING FILE
C
      IF (PDFILE(ICMO,IPUMP).EQ.1) THEN
C
C     FIND WELLS WITH SPECIFIED STATUS
C
      IGET=0
      NGET=0
      DO 460 IWELL=1,NWELL
      DO 4689 IEXC=1,NEXC
      IF (WNUM(IWELL).EQ.EWNUM(IEXC)) GOTO 460
 4689 CONTINUE
      ISTATX=NSTAT(IWELL)
      DO 468 ISTAT=1,ISTATX
      IF (WSTAT(IWELL,ISTAT).NE.PSCODE(ICMO,IPUMP)) GO TO 468
      I1=12*(BSYR(IWELL,ISTAT)-PCYR(1)) +BSMO(IWELL,ISTAT)-PCMO(1)+1
      I2=10000
      IF (ESYR(IWELL,ISTAT).NE.0)
     1   I2=12*(ESYR(IWELL,ISTAT)-PCYR(1))+ESMO(IWELL,ISTAT)-PCMO(1)+1
      II=ICMO
      IF (PUMO(ICMO,IPUMP).NE.0)
     1   II=12*(PUYR(ICMO,IPUMP)-PCYR(1))+PUMO(ICMO,IPUMP)-PCMO(1)+1
      IF (II.GE.I1 .AND. II.LE.I2) THEN
      IGET=IGET+1
      WGET(IGET)=IWELL
      SGET(IGET)=ISTAT
      NGET=IGET
      GO TO 460
      ENDIF
 468  CONTINUE
 460  CONTINUE
      IF (NGET.EQ.0) GO TO 149
C
C     FIND PUMPING VALUES FOR WELLS, DISTRIBUTE TO NODES, ASSIGN
C        PUMPING TO USERS
C
      DO 463 IGET=1,NGET
      IWELL=WGET(IGET)
      ISTAT=SGET(IGET)
      ISEQ1=WSTART(IWELL)
      ISEQ2=ISEQ1+NQPUMP(IWELL)-1
      DO 462 ISEQ=ISEQ1,ISEQ2
      I1=12*(PDYR(ISEQ)-PCYR(1))+PDMO(ISEQ)-PCMO(1)+1
      II=ICMO
      IF (PUMO(ICMO,IPUMP).NE.0)
```

```
     1   II=12*(PUYR(ICMO,IPUMP)-PCYR(1))+PUMO(ICMO,IPUMP)-PCMO(1)+1
      IF (II.EQ.I1) THEN
      QWELL=QPUMP(ISEQ)*PFACT(ICMO,IPUMP)
      XP=XP+QWELL
      SUMP=SUMP-QWELL
      INODEX=NWNODE(IWELL)
      DO 461 I=1,INODEX
      INODE=WNODE(IWELL,I)
      PNODE(KNS,INODE)=PNODE(KNS,INODE)-QWELL*WFACT(IWELL,I)
 461  CONTINUE
      GO TO 162
      ENDIF
 462  CONTINUE
 162  CONTINUE
      DO 163 IUSER=1,NUSER
      IF (UNUM(IUSER).EQ.WUSER(IWELL,ISTAT)) THEN
      JJ=PSCODE(ICMO,IPUMP)
      DO 4638 IPOUT=1,NPOUT
      IF (JJ.EQ.POUT(IPOUT)) GOTO 463
 4638 CONTINUE
      IMO=PCMO(ICMO)
      PUSER(IUSER)=PUSER(IUSER)+QWELL*PORDEL(IUSER,IMO)
      XPU=XPU+QWELL*PORDEL(IUSER,IMO)
      GO TO 463
      ENDIF
 163  CONTINUE
 463  CONTINUE
      ENDIF
 149  CONTINUE
C
C     SAVE PUMPING FOR OUTPUT
C
      IF (IOUT.EQ.1 .AND. (ISS.EQ.0 .OR. KNS.NE.1)) THEN
      DO 1801 I=1,NPINDX
      IF (PSCODE(ICMO,IPUMP).EQ.PINDX(I)) THEN
      INDX=I
      GO TO 1802
      ENDIF
 1801 CONTINUE
      NPINDX=NPINDX+1
      PINDX(NPINDX)=PSCODE(ICMO,IPUMP)
      INDX=NPINDX
 1802 CONTINUE
      PU(ICMO,INDX)=PU(ICMO,INDX)+XPU
      P(ICMO,INDX)=P(ICMO,INDX)+XP
      ENDIF
C
 146  CONTINUE
C
C     PROCESS DELIVERY CONSTRUCTION FILE TO DETERMINE CANAL DELIVERIES
C        TO USERS
C
      DO 138 IUSER=1,NUSER
      DUSER(IUSER)=0.0
 138  CONTINUE
      IDELX=NDEL(ICMO)
      DO 137 IDEL=1,IDELX
      XDU=0.0
```

```
C
C     IDENTIFY ACTIVE USERS
C
      IGET=0
      NGET=0
      DO 141 IUSER=1,NUSER
      IF (UTYPE(IUSER).NE.UTCODE(ICMO,IDEL)) GO TO 141
      IAREAX=NAREA(IUSER)
      DO 140 IAREA=1,IAREAX
      I1=12*(BAYR(IUSER,IAREA)-PCYR(1))+BAMO(IUSER,IAREA)-PCMO(1)+1
      I2=10000
      IF (EAYR(IUSER,IAREA).NE.0)
     1   I2=12*(EAYR(IUSER,IAREA)-PCYR(1))+EAMO(IUSER,IAREA)-PCMO(1)+1
      II=ICMO
      IF (DUMO(ICMO,IDEL).NE.0)
     1   II=12*(DUYR(ICMO,IDEL)-PCYR(1))+DUMO(ICMO,IDEL)-PCMO(1)+1
      IF (II.GE.I1 .AND. II.LE.I2) THEN
      IGET=IGET+1
      NGET=IGET
      DGET(IGET)=IUSER
      AGET(IGET)=IAREA
      GO TO 141
      ENDIF
 140  CONTINUE
 141  CONTINUE
      IF (NGET.EQ.0) GO TO 139
C
C     DISTRIBUTE TOTAL DELIVERY TO USERS BASED ON AREA
C
      IF (DDFILE(ICMO,IDEL).EQ.0) THEN
      SAREA=0.0
      DO 1423 IGET=1,NGET
      IUSER=DGET(IGET)
      IAREA=AGET(IGET)
      SAREA=SAREA+AREA(IUSER,IAREA)
 1423 CONTINUE
      DO 142 IGET=1,NGET
      IUSER=DGET(IGET)
      IAREA=AGET(IGET)
      IMO=PCMO(ICMO)
      DUSER(IUSER)=RDEL(ICMO,IDEL)*DFACT(ICMO,IDEL)*
     1   AREA(IUSER,IAREA)*PORDEL(IUSER,IMO)/SAREA
      XDU=XDU+DUSER(IUSER)
 142  CONTINUE
      GO TO 139
      ENDIF
C
C     FETCH DELIVERY FOR INDIVIDUAL USERS FROM MONTHLY
C        DELIVERY FILE
C
      IF (DDFILE(ICMO,IDEL).EQ.1) THEN
      DO 171 IGET=1,NGET
      IUSER=DGET(IGET)
      IAREA=AGET(IGET)
      ISEQ1=USTART(IUSER)
      ISEQ2=ISEQ1+NQDEL(IUSER)-1
      DO 172 ISEQ=ISEQ1,ISEQ2
      I1=12*(DDYR(ISEQ)-PCYR(1))+DDMO(ISEQ)-PCMO(1)+1
      JJ=ICMO
      IF (DUMO(ICMO,IDEL).NE.0)
```

```
    1   JJ=12*(DUYR(ICMO,IDEL)-PCYR(1))+DUMO(ICMO,IDEL)-PCMO(1)+1
      IF (JJ.EQ.I1) THEN
      IMO=PCMO(ICMO)
      DUSER(IUSER)=QDEL(ISEQ)*DFACT(ICMO,IDEL)*PORDEL(IUSER,IMO)
      XDU=XDU+DUSER(IUSER)
      GO TO 171
      ENDIF
 172  CONTINUE
 170  CONTINUE
 171  CONTINUE
      GO TO 139
      ENDIF
 139  CONTINUE
C
C     SAVE DELIVERY FOR OUTPUT
C
      IF (IOUT.EQ.1 .AND. (ISS.EQ.0 .OR. KNS.NE.1)) THEN
      DO 1803 I=1,NDINDX
      IF (UTCODE(ICMO,IDEL).EQ.DINDX(I)) THEN
      INDX=I
      GO TO 1804
      ENDIF
 1803 CONTINUE
      NDINDX=NDINDX+1
      DINDX(NDINDX)=UTCODE(ICMO,IDEL)
      INDX=NDINDX
 1804 CONTINUE
      DU(ICMO,INDX)=DU(ICMO,INDX)+XDU
      ENDIF
 137  CONTINUE
C
C     PROCESS CROP AND WEATHER CONSTRUCTION FILE TO DETERMINE CROP
C        MIX, ROOTING DEPTH, PET, AND PRECIPITATION FOR USER
C        AND THE RESULTING RECHARGE TO NODES
C
      DO 148 IUSER=1,NUSER
      DEPTH(IUSER)=0.0
      RUSER(IUSER)=0.0
      ETUSER(IUSER)=0.0
      DO 147 ICROP=1,NCROP
      CPOR(IUSER,ICROP)=0.0
 147  CONTINUE
 148  CONTINUE
      ICRECX=NCREC(ICMO)
      DO 165 ICREC=1,ICRECX
C
C     IDENTIFY ACTIVE USERS
C
      IGET=0
      NGET=0
      SAREA=0.0
      DO 1418 IUSER=1,NUSER
      IF (UTYPE(IUSER).NE.ETCODE(ICMO,ICREC)) GO TO 1418
      IAREAX=NAREA(IUSER)
      DO 1408 IAREA=1,IAREAX
      I1=12*(BAYR(IUSER,IAREA)-PCYR(1))+BAMO(IUSER,IAREA)-PCMO(1)+1
      I2=10000
      IF (EAYR(IUSER,IAREA).NE.0)
    1   I2=12*(EAYR(IUSER,IAREA)-PCYR(1))+EAMO(IUSER,IAREA)-PCMO(1)+1
      II=ICMO
      IF (DUMO(ICMO,IDEL).NE.0)
```

```
     1   II=12*(DUYR(ICMO,IDEL)-PCYR(1))+DUMO(ICMO,IDEL)-PCMO(1)+1
      IF (II.GE.I1 .AND. II.LE.I2) THEN
      IGET=IGET+1
      NGET=IGET
      DGET(IGET)=IUSER
      AGET(IGET)=IAREA
      GO TO 1418
      ENDIF
 1408 CONTINUE
 1418 CONTINUE
      IF (NGET.EQ.0) GO TO 165
      XRU=0.0
      XEU=0.0
C
C     FIND PROPORTION OF CROP AND WEIGHTED ROOTING DEPTH
C        FOR EACH USER
C
      DO 154 IGET=1,NGET
      IUSER=DGET(IGET)
      IYR=CUYR(ICMO,ICREC)
      IRECX=NREC(IUSER)
      DO 153 IREC=1,IRECX
      IF (INYR(IUSER,IREC).EQ.IYR) THEN
      DO 152 ICROP=1,NCROP
      IF (CCODE(ICROP).EQ.INCODE(IUSER,IREC)) THEN
      CPOR(IUSER,ICROP)=CPOR(IUSER,ICROP)+INPOR(IUSER,IREC)
      DEPTH(IUSER)=DEPTH(IUSER)+ROOT(ICROP)*CPOR(IUSER,ICROP)
      ENDIF
 152  CONTINUE
      ENDIF
 153  CONTINUE
      IF (DEPTH(IUSER).LE.0.0) WRITE(IW1,9736) UNUM(IUSER),DEPTH(IUSER)
 154  CONTINUE
 9736 FORMAT(1X,'ERROR IN CROP INVENTORY OF USER ',A8,F10.2)
C
C     DETERMINE WEIGHTED PET
C
      II=ICMO
      IF (EUYR(ICMO,ICREC).NE.0)
     1   II=12*(EUYR(ICMO,ICREC)-PCYR(1))+EUMO(ICMO,ICREC)-PCMO(1)+1
      DO 158 IGET=1,NGET
      IUSER=DGET(IGET)
      DO 157 ICROP=1,NCROP
      ETUSER(IUSER)=ETUSER(IUSER)+PET(ICROP,II)*CPOR(IUSER,ICROP)
 157  CONTINUE
C MC CHANGE ERROR STATEMENT:
C A) IF PET=0
C B) IF PET<0
      IF (ETUSER(IUSER).LT.0.0) WRITE(IW1,9737) UNUM(IUSER)
      IF (ETUSER(IUSER).EQ.0.0) WRITE(IW1,9738) UNUM(IUSER),II
 158  CONTINUE
 9737 FORMAT(1X,'ERROR IN PET FOR USER ',A8)
 9738 FORMAT(1X,'PET=0 FOR USER ',A8,5X,'MONTH ',I5)
C
C     DETERMINE PRECIPITATION FOR USER
C
      II=ICMO
      IF (EUYR(ICMO,ICREC).NE.0)
```

```
     1   II=12*(EUYR(ICMO,ICREC)-PCYR(1))+EUMO(ICMO,ICREC)-PCMO(1)+1
      DO 159 IGET=1,NGET
      IUSER=DGET(IGET)
      RUSER(IUSER)=RAIN(II)*RFACT(IUSER)
 159  CONTINUE
C
C     CALCULATE RECHARGE AND DISTRIBUTE TO MODEL NODES
C
      DO 161 IGET=1,NGET
      IUSER=DGET(IGET)
      IAREA=AGET(IGET)
      CALL LAND(IUSER,NUSER,ICMO,AREA(IUSER,IAREA),DUSER(IUSER),
     1   RUSER(IUSER)/12.0,ETUSER(IUSER)/12.0,PUSER(IUSER),
     2   DEPTH(IUSER)/12.0,THETA0(IUSER),CLOSS(IUSER),TAILF(IUSER),
     3   AWC(IUSER),GAMMA(IUSER),UCOEF(IUSER),RECH,IOUT,EVAP)
      XEU=XEU+EVAP
      INODEX=NUNODE(IUSER,IAREA)
      DO 167 I=1,INODEX
      INODE=UNODE(IUSER,IAREA,I)
      RNODE(KNS,INODE)=RNODE(KNS,INODE)+RECH*UFACT(IUSER,IAREA,I)*
     1   RDEEP(INODE)
      SUMR=SUMR+RECH*UFACT(IUSER,IAREA,I)*RDEEP(INODE)
      XRU=XRU+RECH*UFACT(IUSER,IAREA,I)*RDEEP(INODE)
 167  CONTINUE
 160  CONTINUE
 161  CONTINUE
      IF (IOUT.EQ.1 .AND. (ISS.EQ.0 .OR. KNS.NE.1)) THEN
      DO 1853 I=1,NRINDX
      IF (ETCODE(ICMO,ICREC).EQ.RINDX(I)) THEN
      INDX=I
      GO TO 1854
      ENDIF
 1853 CONTINUE
      NRINDX=NRINDX+1
      RINDX(NRINDX)=ETCODE(ICMO,ICREC)
      INDX=NRINDX
 1854 CONTINUE
      RU(ICMO,INDX)=RU(ICMO,INDX)+XRU
      EU(ICMO,INDX)=EU(ICMO,INDX)+XEU
      ENDIF
 165  CONTINUE
C
C     END LOOP OVER SUPPLY STEPS
C
 590  CONTINUE
C
C     CONVERT TO CFS FOR MODEL
C
      CFS=1.0/(60.33*(ICMO2-ICMO1+1))
      SUMR=0.0
      SUMP=0.0
      DO 589 I=1,NN
      PNODE(KNS,I)=PNODE(KNS,I)*CFS
      RNODE(KNS,I)=RNODE(KNS,I)*CFS
      SUMP=SUMP+PNODE(KNS,I)
      SUMR=SUMR+RNODE(KNS,I)
```

```
 589  CONTINUE
C
C     OUTPUT PUMPING AND DELIVERIES
C
      IF (MAXKNS.NE.KNS .OR. IOUT.NE.1) RETURN
C
      NCMO=ICMO2
      DO 1819 INDX=1,NPINDX
      WRITE(IW1,981) PINDX(INDX)
      WRITE(IW1,982)
      WRITE(IW1,983) (PCMO(ICMO),ICMO=1,12)
      WRITE(IW1,984)
      ICMO1=-11
      SUMXXP=0.0
 1817 ICMO1=ICMO1+12
      ICMO2=ICMO1+11
      IF (ICMO1.GT.NCMO) THEN
      WRITE(IW1,9857) SUMXXP
      GO TO 1819
      ENDIF
      IF (ICMO2.GT.NCMO) ICMO2=NCMO
      XXP=0.0
      DO 1818 ICMO=ICMO1,ICMO2
      XXP=XXP+P(ICMO,INDX)
 1818 CONTINUE
      SUMXXP=SUMXXP+XXP
      WRITE(IW1,985) PCYR(ICMO2),XXP,(P(ICMO,INDX),ICMO=ICMO1,ICMO2)
      GO TO 1817
 1819 CONTINUE
C
      DO 1822 INDX=1,NPINDX
      WRITE(IW1,986) PINDX(INDX)
      WRITE(IW1,982)
      WRITE(IW1,983) (PCMO(ICMO),ICMO=1,12)
      WRITE(IW1,984)
      ICMO1=-11
      SUMXXPU=0.0
 1820 ICMO1=ICMO1+12
      ICMO2=ICMO1+11
      IF (ICMO1.GT.NCMO) THEN
      WRITE(IW1,9857) SUMXXPU
      GO TO 1822
      ENDIF
      IF (ICMO2.GT.NCMO) ICMO2=NCMO
      XXPU=0.0
      DO 1821 ICMO=ICMO1,ICMO2
      XXPU=XXPU+PU(ICMO,INDX)
 1821 CONTINUE
      SUMXXPU=SUMXXPU+XXPU
      WRITE(IW1,985) PCYR(ICMO2),XXPU,(PU(ICMO,INDX),ICMO=ICMO1,ICMO2)
      GO TO 1820
 1822 CONTINUE
C
      DO 1825 INDX=1,NDINDX
      WRITE(IW1,987) DINDX(INDX)
      WRITE(IW1,982)
      WRITE(IW1,983) (PCMO(ICMO),ICMO=1,12)
      WRITE(IW1,984)
      ICMO1=-11
      SUMXXDU=0.0
```

```
 1823 ICMO1=ICMO1+12
      ICMO2=ICMO1+11
      IF (ICMO1.GT.NCMO) THEN
      WRITE(IW1,9857) SUMXXDU
      GO TO 1825
      ENDIF
      IF (ICMO2.GT.NCMO) ICMO2=NCMO
      XXDU=0.0
      DO 1824 ICMO=ICMO1,ICMO2
      XXDU=XXDU+DU(ICMO,INDX)
 1824 CONTINUE
      SUMXXDU=SUMXXDU+XXDU
      WRITE(IW1,985) PCYR(ICMO2),XXDU,(DU(ICMO,INDX),ICMO=ICMO1,ICMO2)
      GO TO 1823
 1825 CONTINUE
C
      DO 1828 INDX=1,NRINDX
      WRITE(IW1,988) RINDX(INDX)
      WRITE(IW1,982)
      WRITE(IW1,983) (PCMO(ICMO),ICMO=1,12)
      WRITE(IW1,984)
      ICMO1=-11
      SUMXXRU=0.0
 1826 ICMO1=ICMO1+12
      ICMO2=ICMO1+11
      IF (ICMO1.GT.NCMO) THEN
      WRITE(IW1,9857) SUMXXRU
      GO TO 1828
      ENDIF
      IF (ICMO2.GT.NCMO) ICMO2=NCMO
      XXRU=0.0
      DO 1827 ICMO=ICMO1,ICMO2
      XXRU=XXRU+RU(ICMO,INDX)
 1827 CONTINUE
      SUMXXRU=SUMXXRU+XXRU
      WRITE(IW1,985) PCYR(ICMO2),XXRU,(RU(ICMO,INDX),ICMO=ICMO1,ICMO2)
      GO TO 1826
 1828 CONTINUE
C
      DO 1831 INDX=1,NRINDX
      WRITE(IW1,989) RINDX(INDX)
      WRITE(IW1,982)
      WRITE(IW1,983) (PCMO(ICMO),ICMO=1,12)
      WRITE(IW1,984)
      ICMO1=-11
      SUMXXEU=0.0
 1829 ICMO1=ICMO1+12
      ICMO2=ICMO1+11
      IF (ICMO1.GT.NCMO) THEN
      WRITE(IW1,9857) SUMXXEU
      GO TO 1831
      ENDIF
      IF (ICMO2.GT.NCMO) ICMO2=NCMO
      XXEU=0.0
      DO 1830 ICMO=ICMO1,ICMO2
      XXEU=XXEU+EU(ICMO,INDX)
```

```
1830 CONTINUE
     SUMXXEU=SUMXXEU+XXEU
     WRITE(IW1,985) PCYR(ICMO2),XXEU,(EU(ICMO,INDX),ICMO=ICMO1,ICMO2)
     GO TO 1829
1831 CONTINUE
981  FORMAT(/1X,'PUMPING FROM WELLS FOR PSCODE',I3/1X,32('-'))
982  FORMAT(/1X,6X,'YEAR',5X,'TOTAL',3X,'MONTH'/1X,6X,4('-'),
    1   5X,5('-'),3X,5('-'))
983  FORMAT(1X,20X,12I8)
984  FORMAT(1X)
985  FORMAT(1X,I10,F10.1,12F8.1)
9857 FORMAT(11X,F10.1)
986  FORMAT(/1X,'PUMPING TO USERS FOR PSCODE',I3/1X,30('-'))
987  FORMAT(/1X,'DELIVERIES TO USERS FOR UTCODE',I3/1X,33('-'))
988  FORMAT(/1X,'RECHARGE TO USERS FOR UTCODE',I3/1X,31('-'))
989  FORMAT(/1X,'ET FOR USERS FOR UTCODE',I3/1X,26('-'))
     RETURN
     END
```

```
C     -----------------------------------------------------
      SUBROUTINE LAND(IUSER,NUSER,ICMO,AREA,DUSER,RUSER,
     1   ETUSER,PUSER,ROOT,THETA0,CLOSS,TAILF,THAWC,GAMMA,
     2   UCOEF,RECH,IOUT,EVAP)
C     -----------------------------------------------------
C
      INCLUDE 'PARAM2.FOR'
      IMPLICIT INTEGER*2 (I-N)
      REAL THETA(MAXUSE,5)
C
C     COMPUTE GW RECHARGE, ET, AND SOIL MOISTURE
C
C     THAWC   AVAILABLE WATER CAPACITY OF SOIL=FIELD CAPACITY-WILTING POINT
C     DRECH   GROUND WATER RECHARGE
C     GAMMA   CRITICAL SOIL MOISTURE
C     PET     POTENTIAL EVAPOTRANSPIRATION FOR USER
C     PPT     PRECIPITATION (INCHES)
C     PPTEFF  EFFECTIVE PRECIPITATION
C     PUSER   PUMPED IRRIGATION WATER
C     RETFLO  RETURN FLOW FROM APPLIED WATER AND PRECIP FOR USER
C     THETA0  INITIAL SOIL MOISTURE FOR USER, REFERENCED TO WILTING POINT
C     THETA   SOIL MOISTURE PER USER, PER ZONE, REFERNCED TO WILTING POINT
C     UCOEF   UNIFORMITY COEFFICIENT FOR DISTRIBUTING APPLIED WATER
C     TAILF   TAILWATER FACTOR FOR SURFACE RUNOFF FROM FIELD FURROWS
C
C     INITIALIZE MASS BALANCE CHECKS FOR MONTH
C
      IF (IUSER.EQ.1) THEN
      XPPT=0.0
      XPUMP=0.0
      XCANAL=0.0
      XDP=0.0
      XTAIL=0.0
      XRO=0.0
      XET=0.0
      XDS=0.0
      ENDIF
C
C     INITIALIZE SOIL MOISTURE FOR EACH ZONE
C
      IF (ICMO.EQ.1) THEN
      SXRES=0.0
      DO 120 IZONE=1,5
      THETA(IUSER,IZONE)=THETA0
  120 CONTINUE
      ENDIF
C
C     USE USBR METHOD TO COMPUTE EFFECTIVE PRECIPITATION
C
      PPT=RUSER*12.0
      IF (PPT.LT.1.0) THEN
      PPTEFF=PPT*0.95
      GO TO 130
      ELSE
      PPTEFF=0.95
      PPT=PPT-1.0
      ENDIF
      IF (PPT.LT.1.0) THEN
      PPTEFF=PPTEFF+PPT*0.90
      GO TO 130
      ELSE
      PPTEFF=PPTEFF+0.90
```

```
      PPT=PPT-1.0
      ENDIF
      IF (PPT.LT.1.0) THEN
      PPTEFF=PPTEFF+PPT*0.825
      GO TO 130
      ELSE
      PPTEFF=PPTEFF+0.825
      PPT=PPT-1.0
      ENDIF
      IF (PPT.LT.1.0) THEN
      PPTEFF=PPTEFF+PPT*0.65
      GO TO 130
      ELSE
      PPTEFF=PPTEFF+0.65
      PPT=PPT-1.0
      ENDIF
      IF (PPT.LT.1.0) THEN
      PPTEFF=PPTEFF+PPT*0.45
      GO TO 130
      ELSE
      PPTEFF=PPTEFF+0.45
      PPT=PPT-1.0
      ENDIF
      IF (PPT.LT.1.0) THEN
      PPTEFF=PPTEFF+PPT*0.25
      GO TO 130
      ELSE
      PPTEFF=PPTEFF+0.25
      PPT=PPT-1.0
      ENDIF
      PPTEFF=PPTEFF+PPT*0.05
  130 CONTINUE
      PPTEFF=PPTEFF/12.0
      RO=RUSER-PPTEFF
C
C     WATER INFILTRATED INTO SOILS FOR USER
C
      APPLY=DUSER*(1.0-CLOSS)+PUSER
      TAIL=TAILF*APPLY
      APPLY=APPLY-TAIL
      APPLY=APPLY/AREA
C
C     SOIL MOISTURE WHERE ET CURVE BREAKS
C
      THETAG=GAMMA*THAWC
C
C     DIVIDE FIELD INTO FIVE ZONES
C
      PET=ETUSER
      DS=0.0
      DP=0.0
      ET=0.0
      DO 140 IZONE=1,5
C
C     APPLIED WATER TO ZONE
C
      UC=UCOEF
      IF (UCOEF.LT.0.4) UC=0.4
      AFAC=1.0 + ((1.0-UC)*5.0)*((3-IZONE)/6.0)
      AWAT=APPLY*AFAC
      AWAT=AWAT+PPTEFF
```

```
C
C     COMPUTE EVAPOTRANSPIRATION FOR CASE OF APPLIED WATER
C        GREATER THAN PET
C
      THETAI=THETA(IUSER,IZONE)
      IF (AWAT.GE.PET) THEN
      DELTH=(AWAT-PET)/ROOT
      THETAF=THETAI+DELTH
      DRAIN=0.0
      IF (THETAF.GT.THAWC) THEN
      DRAIN=(THETAF-THAWC)*ROOT
      THETAF=THAWC
      DELTH=THETAF-THETAI
      ENDIF
      ET=ET+PET
      DP=DP+DRAIN
      DS=DS+DELTH*ROOT
      GO TO 135
      ENDIF
C
C     COMPUTE EVAPOTRANSPIRATION FOR CASE OF APPLIED WATER
C        LESS THAN PET
C
      IF (AWAT.LT.PET) THEN
      ASTORE=0.0
      IF (THETAI.GT.THETAG) ASTORE=(THETAI-THETAG)*ROOT
C
C     ALL ET AT PET RATE
C
      IF (AWAT+ASTORE.GE.PET) THEN
      DELTH=(AWAT-PET)/ROOT
      THETAF=THETAI+DELTH
      DRAIN=0.0
      ET=ET+PET
      DP=DP+DRAIN
      DS=DS+DELTH*ROOT
      GO TO 135
      ENDIF
C
C     ONLY SOME ET AT PET RATE
C
      IF (AWAT+ASTORE.LT.PET) THEN
      DELT=1.0-(AWAT+ASTORE)/PET
      XTHETA=THETAG
      IF (THETAI.LT.THETAG) XTHETA=THETAI
      THETAF=XTHETA+XTHETA*(EXP(-PET*DELT/(ROOT*THETAG))-1.0)
      DELTH=THETAF-THETAI
      DRAIN=0.0
      ET=ET+AWAT+ASTORE-(THETAF-XTHETA)*ROOT
      DP=DP+DRAIN
      DS=DS+DELTH*ROOT
      GO TO 135
      ENDIF
      ENDIF
 135  CONTINUE
      THETA(IUSER,IZONE)=THETAF
 140  CONTINUE
C
C     AVERAGE OVER FIVE ZONES
```

```
C
      ET=ET/5.0
      DP=DP/5.0
      DS=DS/5.0
      RECH=DP*AREA+CLOSS*DUSER
      EVAP=ET*AREA
C
C     WATER BUDGET FOR LAND-SURFACE SYSTEM
C
      XET=XET+ET*AREA
      XDP=XDP+RECH
      XDS=XDS+DS*AREA
      XPPT=XPPT+RUSER*AREA
      XCANAL=XCANAL+DUSER
      XPUMP=XPUMP+PUSER
      XTAIL=XTAIL+TAIL
      XRO=XRO+RO*AREA
      IF (IUSER.EQ.NUSER) THEN
      XRES=XPUMP+XCANAL+XPPT-XTAIL-XRO-XET-XDP-XDS
      SXRES=SXRES+XRES
      IF (IOUT.EQ.1) THEN
      ENDIF
      ENDIF
 900  FORMAT(/1X,'LAND WATER BUDGET'/1X,17('-')/
     1   1X,'TIME STEP                   ',I10/
     2   1X,'PUMPING TO USERS            ',F10.2/
     3   1X,'CANAL DELIVERIES TO USERS ',F10.2/
     4   1X,'PRECIPITATION               ',F10.2/
     5   1X,'TAIL WATER FROM FIELDS      ',F10.2/
     6   1X,'RUNOFF                      ',F10.2/
     7   1X,'EVAPOTRANSPIRATION          ',F10.2/
     8   1X,'DEEP PERCOLATION            ',F10.2/
     9   1X,'MOISTURE STORAGE CHANGE     ',F10.2/
     1   1X,'RESIDUAL                    ',F10.2/
     2   1X,'CUMULATIVE RESIDUAL         ',F10.2)
      RETURN
      END
```

```
C     -----------------
      SUBROUTINE FAULT1
C     -----------------
C
      INCLUDE 'PARAM.FOR'
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*2 (I-N)
      REAL*8 H(MAXNN),QFNET(MAXNN)
      REAL*8 TRAN(MAXFLN),HEIGHT(MAXFLN),LENGTH(MAXFLN),
     1   QFAULT(MAXFLN,2)
      REAL*8 AE(2,2)
      REAL*8 A(MAXNN,MAXNB)
      INTEGER*2 FNODE(MAXFLN,2),ICOL(MAXNN,MAXNB),NCOL(MAXNN)
      IR1=11
      IW1=13
C
C     INPUT DATA ON LOCATION AND CHARACTER OF FAULTS
C
      WRITE(*,'(1X,''READING IN FAULT'')')
      READ(IR1,900) NFN,IECHO1,IECHO2
      WRITE(IW1,904) NFN,IECHO1,IECHO2
      IF (NFN.EQ.0) RETURN
      IF (IECHO1.NE.0) WRITE(IW1,901)
      DO 110 L=1,NFN
      READ(IR1,902) FNODE(L,1),FNODE(L,2),TRAN(L),LENGTH(L),
     1   HEIGHT(L)
      IF (IECHO1.NE.0) THEN
      WRITE(IW1,903) L,FNODE(L,1),FNODE(L,2),TRAN(L),LENGTH(L),
     1   HEIGHT(L)
      ENDIF
      TRAN(L)=TRAN(L)/(24.0*3600.0)
 110  CONTINUE
 900  FORMAT(3I10)
 902  FORMAT(2I10,3E10.0)
 901  FORMAT(/1X,'FAULT NODE PAIRS'/1X,16('-')/
     1   1X,5X,'FAULT',5X,'NODE1',5X,'NODE2',6X,'TRAN',
     1   4X,'LENGTH',4X,'HEIGHT'/)
 903  FORMAT(1X,3I10,3F10.2)
 904  FORMAT(/1X,'FAULT NODES'/1X,11('-')/
     1   1X,'NUMBER OF FAULT-NODE PAIRS                ',I10/
     2   1X,'SWITCH FOR ECHO OF INPUT                  ',I10/
     3   1X,'SWITCH FOR ECHO OF OUTPUT                 ',I10)
      RETURN
C
C     ------------------------------
      ENTRY FAULT2(A,ICOL,NCOL,ISOLVE)
C     ------------------------------
C
C     ADD FAULTS TO MATRIX
C
      IF (NFN.EQ.0) RETURN
      AE(1,1)=1.0
      AE(1,2)=-1.0
      AE(2,1)=-1.0
      AE(2,2)=1.0
```

```
C
      IF (ISOLVE.EQ.1) THEN
      DO 1808 L=1,NFN
      DO 1708 I=1,2
      II=FNODE(L,I)
      DO 1608 J=1,2
      JJ=FNODE(L,J)-II+1
      IF (JJ.LT.1) GO TO 4078
      A(II,JJ)=A(II,JJ)+AE(I,J)*TRAN(L)*HEIGHT(L)/LENGTH(L)
 4078 CONTINUE
 1608 CONTINUE
 1708 CONTINUE
 1808 CONTINUE
      ENDIF
C
      IF (ISOLVE.EQ.2) THEN
      DO 680 L=1,NFN
      DO 611 I=1,2
      II=FNODE(L,I)
      DO 610 J=1,2
      JJ=FNODE(L,J)
      NC=NCOL(II)
      DO 6071 IC=1,NC
      IF (ICOL(II,IC).EQ.JJ) THEN
      A(II,IC)=A(II,IC)+AE(I,J)*TRAN(L)*HEIGHT(L)/LENGTH(L)
      GOTO 610
      ENDIF
 6071 CONTINUE
      WRITE(*,*) 'DID NOT FILL'
      STOP
 610  CONTINUE
 611  CONTINUE
 680  CONTINUE
      ENDIF
      RETURN
C
C     ------------------------------
      ENTRY FAULT3(H,SFAULT,NN,IOUT)
C     ------------------------------
C
C     FLUX ALONG FAULT
C
      SFAULT=0.0
      DO 150 I=1,NN
      QFNET(I)=0.0
 150  CONTINUE
      IF (NFN.EQ.0) RETURN
C
      DO 200 L=1,NFN
      I1=FNODE(L,1)
      I2=FNODE(L,2)
      QFAULT(L,1)=-TRAN(L)*HEIGHT(L)*(H(I2)-H(I1))/LENGTH(L)
      QFAULT(L,2)=-QFAULT(L,1)
      QFNET(I1)=QFNET(I1)+QFAULT(L,1)
      QFNET(I2)=QFNET(I2)+QFAULT(L,2)
 200  CONTINUE
      SFAULT=0.0
      DO 152 I=1,NN
      SFAULT=SFAULT+QFNET(I)
```

```
152  CONTINUE
     IF (NFN.EQ.0 .OR. IOUT.EQ.0 .OR. IECHO2.EQ.0) RETURN
     WRITE(IW1,915)
     WRITE(IW1,917) (FNODE(L,1),FNODE(L,2),QFAULT(L,1),L=1,NFN)
     WRITE(IW1,920) SFAULT
     DO 151 I=1,NN
     IF (QFNET(I).NE.0.0) THEN
     WRITE(IW1,921) I,QFNET(I)
     ENDIF
151  CONTINUE
915  FORMAT(/1X,'FAULT PAIR FLUXES'/1X,17('-')/
    1   1X,5X,'NODE1',5X,'NODE2',6X,'RATE'/)
917  FORMAT(1X,2I10,1PE10.2)
920  FORMAT(/1X,'NET FAULT FLUXES'/1X,16('-')/
    1   1X,'CUMULATIVE RATE      ',1PE10.2//
    2   1X,6X,'NODE',6X,'RATE'/)
921  FORMAT(1X,I10,1PE10.2)
     RETURN
     END
```

```
C      -----------------------------------------------------------
       SUBROUTINE SINK1(NN,NE,NE2,IEL,HP,POR2,IPRISM,X,Y)
C      -----------------------------------------------------------
C
       INCLUDE 'PARAM.FOR'
       IMPLICIT REAL*8 (A-H,O-Z)
       IMPLICIT INTEGER*2 (I-N)
       REAL*8 SSKE(MAXNE),SSKE2(MAXNE2),SSKV(MAXNE),SSKV2(MAXNE2),
      1  SSK2(MAXNE2)
       REAL*8 POR(MAXNE),POR2(MAXNE2)
       REAL*8 HP(MAXNN),H(MAXNN),HL(MAXNN),X(MAXNN),Y(MAXNN),Z(MAXNN)
       REAL*8 XE(4),YE(4),ZE(4),CE(4,4),DE(4,4)
       INTEGER*2 IEL(MAXNE2),IN(MAXNE2,4),ISTATE(MAXNE2)
       INTEGER*2 COL(MAXCOL,MAXNUM),NUM(MAXCOL),IPRISM(MAXNE,6)
       REAL*8 AREA(MAXCOL),QSKE(MAXNE),QSKV(MAXNE),LAND(MAXCOL)
       IR1=11
       IW1=13
C
       WRITE(*,'(1X,''READING IN SINK'')')
       READ(IR1,900) ISINK,IECHO1,IECHO2,IECHO3,IECHO4,IECHO5
       WRITE(IW1,903) IECHO1,IECHO2,IECHO3,IECHO4,IECHO5
       DO 101 I=1,NN
       HP(I)=0.0
  101  CONTINUE
       DO 102 I=1,NE2
       POR2(I)=0.0
  102  CONTINUE
       IF (ISINK.EQ.0) RETURN
C
C      PARAMETERS FOR DEFORMABLE BEDS
C
       READ(IR1,901) FACSKE,FACSKV,FACPOR
       READ(IR1,902) (SSKE(I),SSKV(I),POR(I),I=1,NE)
       DO 105 I=1,NE
       SSKE(I)=SSKE(I)*FACSKE
       SSKV(I)=SSKV(I)*FACSKV
       POR(I)=POR(I)*FACPOR
  105  CONTINUE
       DO 104 LT=1,NE2
       LP=IEL(LT)
       SSKE2(LT)=SSKE(LP)*FACSKE
       SSKV2(LT)=SSKV(LP)*FACSKV
       POR2(LT)=POR(LP)
  104  CONTINUE
       IF (IECHO1.NE.0) THEN
       WRITE(IW1,905) FACSKE,FACSKV,FACPOR
       WRITE(IW1,904) (I,SSKE(I),SSKV(I),  POR(I),I=1,NE)
       ENDIF
  900  FORMAT(6I10)
  901  FORMAT(3F10.0)
  902  FORMAT(10X,3F10.0)
  903  FORMAT(/1X,'LAND-SUBSIDENCE PARAMETERS'/1X,26('-')//
      2   1X,'SWITCH FOR ECHO OF PARAMETERS            ',I10/
      3   1X,'SWITCH FOR ECHO OF INITIAL CONSOLIDATION',I10/
      4   1X,'SWITCH FOR OUTPUT OF CONSOLIDATION      ',I10/
      5   1X,'SWITCH FOR ECHO OF COLUMNS               ',I10/
      6   1X,'SWITCH FOR OUTPUT OF SUBSIDENCE          ',I10)
```

```
 905  FORMAT(/1X,'AQUITARD PARAMETERS'/1X,19('-')//
     1    1X,'FACTOR FOR SSKE                              ',1PE10.2/
     2    1X,'FACTOR FOR SSKV                              ',1PE10.2/
     3    1X,'FACTOR FOR POROSITY                          ',1PE10.2//
     4    1X,6X,'ELEM',6X,'SSKE',6X,'SSKV',7X,'POR'//)
 904  FORMAT((1X,I10,3(1PE10.2)))
C
C     PRE-CONSOLIDATION HEADS
C
      READ(IR1,928) (HP(I),I=1,NN)
      IF (IECHO2.NE.0) THEN
      WRITE(IW1,929)
      IDONE=0
      I1=-4
 566  I1=I1+5
      I2=I1+4
      IF (I2.GE.NN) THEN
      I2=NN
      IDONE=1
      ENDIF
      WRITE(IW1,930) I1,(HP(I),I=I1,I2)
      IF (IDONE.EQ.0) GO TO 566
      ENDIF
 928  FORMAT(5F10.0)
 929  FORMAT(/1X,'INITIAL PRE-CONSOLIDATION HEADS'/1X,31('-')/
     1    1X,6X,'NODE',4X,'HEADS'/)
 930  FORMAT(1X,I10,5F10.2)
C
C     ELEMENT COLUMNS
C     INPUT COLUMNS
C
      READ(IR1,981) NCOL,NSTACK
      WRITE(IW1,982) NCOL,NSTACK
      IF (NCOL.NE.0) THEN
      DO 161 ICOL=1,NCOL
      READ(IR1,983) (COL(ICOL,I),I=1,NSTACK)
      LAND(ICOL)=0.0
 161  CONTINUE
C
      DO 164 ICOL=1,NCOL
      DO 162 I=1,NSTACK
      IF (COL(ICOL,I).NE.0) GOTO 162
      NUM(ICOL)=I-1
      GOTO 164
 162  CONTINUE
      NUM(ICOL)=NSTACK
 164  CONTINUE
C
      IF (IECHO4.EQ.1) THEN
      WRITE(IW1,984)
      DO 165 ICOL=1,NCOL
      N=NUM(ICOL)
      IF (N.LE.5) WRITE(IW1,985) ICOL,(COL(ICOL,I),I=1,N)
      IF (N.GT.5) THEN
      WRITE(IW1,985) ICOL,(COL(ICOL,I),I=1,5)
      WRITE(IW1,986) (COL(ICOL,I),I=6,N)
      ENDIF
 165  CONTINUE
      ENDIF
      ENDIF
```

```
 981   FORMAT(2I10)
 982   FORMAT(/1X,'COMPACTION COLUMNS'/1X,18('-')/
     1    1X,'NUMBER OF COLUMNS                              ',I10/
     2    1X,'MAXIMUM NUMBER OF ELEMENTS IN STACK            ',I10)
 983   FORMAT(10X,5I10)
 984   FORMAT(/1X,4X,'COLUMN',2X,'ELEMENTS'/)
 985   FORMAT((1X,6I10))
 986   FORMAT((1X,10X,5I10))
C
C      COLUMN AREAS
C
       DO 167 ICOL=1,NCOL
       L=COL(ICOL,1)
       I1=IPRISM(L,1)
       I2=IPRISM(L,2)
       I3=IPRISM(L,3)
       D1=Y(I2)-Y(I3)
       D2=Y(I3)-Y(I1)
       D3=Y(I1)-Y(I2)
       AREA(ICOL)=(D1*X(I1)+D2*X(I2)+D3*X(I3))*0.5
       IF (AREA(ICOL).LE.0.0) WRITE(*,987) ICOL,AREA(ICOL)
 167   CONTINUE
 987   FORMAT(1X,'BAD AREA FOR ELEMENT',I10,1PE10.2)
       RETURN
C
C      ---------------------------------------
       ENTRY SINK2(XE,YE,ZE,CE,DE,LL,IHP,POR2)
C      ---------------------------------------
C
C      INITIALIZE MATRICIES
C
       DO 120 I=1,4
       DO 110 J=1,4
       CE(I,J)=0.0
       DE(I,J)=0.0
 110   CONTINUE
 120   CONTINUE
       IF (ISINK.EQ.0) RETURN
C
C      VOLUME INTEGRATIONS
C      RE-ASSIGN NODAL COORDINATES
C
       X1=XE(1)
       X2=XE(2)
       X3=XE(3)
       X4=XE(4)
       Y1=YE(1)
       Y2=YE(2)
       Y3=YE(3)
       Y4=YE(4)
       Z1=ZE(1)
       Z2=ZE(2)
       Z3=ZE(3)
       Z4=ZE(4)
C
C      COFACTORS OF THE VOLUME DETERMINATE
```

```
C
      A1=X2*(Y3*Z4-Y4*Z3)-X3*(Y2*Z4-Y4*Z2)+X4*(Y2*Z3-Y3*Z2)
      B1=-Y3*Z4+Y4*Z3+Y2*Z4-Y4*Z2-Y2*Z3+Y3*Z2
      C1=X3*Z4-X4*Z3-X2*Z4+X4*Z2+X2*Z3-X3*Z2
      D1=-X3*Y4+X4*Y3+X2*Y4-X4*Y2-X2*Y3+X3*Y2
      VOL=(A1+B1*X1+C1*Y1+D1*Z1)/6.0
      IF (VOL.LE.0.0) WRITE(*,'(1X,''NEGATIVE VOLUME FOR'',I10)') LL
C
C     ELEMENT MATRICIES
C
      SSK2(LL)=SSKE2(LL)
      IF (IHP.EQ.1) THEN
      SSK2(LL)=SSKV2(LL)
      ISTATE(LL)=1
      ELSE
      SSK2(LL)=SSKE2(LL)
      ISTATE(LL)=0
      ENDIF
      DO 201 I=1,4
      DO 200 J=1,4
      CE(I,J)=POR2(LL)*SSK2(LL)*VOL/20.0
      IF (I.EQ.J) CE(I,J)=CE(I,J)*2.0
      DE(I,J)=POR2(LL)*SSKE2(LL)*VOL/20.0
      IF (I.EQ.J) DE(I,J)=DE(I,J)*2.0
 200  CONTINUE
 201  CONTINUE
      RETURN
C     ---------------------------------------------------------------
      ENTRY SINK3(X,Y,Z,XE,YE,ZE,IN,NN,NE,NE2,IEL,POR2,H,HL,HP,SUMQE,
     1   SUMQV,FSS,DELT,IOUT)
C     ---------------------------------------------------------------
C
      SUMQE=0.0
      SUMQV=0.0
      DO 392 LP=1,NE
      QSKE(LP)=0.0
      QSKV(LP)=0.0
 392  CONTINUE
      IF (ISINK.EQ.0) RETURN
C
C     STORAGE CHANGE DUE TO ELASTIC RESPONSE
C
      DO 302 LT=1,NE2
C
C     RE-ASSIGN NODAL COORDINATES
C
      SUMH=0.0
      SUMHL=0.0
      SUMHP=0.0
      DO 300 I=1,4
      II=IN(LT,I)
      XE(I)=X(II)
      YE(I)=Y(II)
      ZE(I)=Z(II)
      SUMH=SUMH+H(II)*0.25
      SUMHL=SUMHL+HL(II)*0.25
      SUMHP=SUMHP+HP(II)*0.25
```

```
 300  CONTINUE
      X1=XE(1)
      X2=XE(2)
      X3=XE(3)
      X4=XE(4)
      Y1=YE(1)
      Y2=YE(2)
      Y3=YE(3)
      Y4=YE(4)
      Z1=ZE(1)
      Z2=ZE(2)
      Z3=ZE(3)
      Z4=ZE(4)
C
C     VOLUME OF ELEMENT
C
      A1=X2*(Y3*Z4-Y4*Z3)-X3*(Y2*Z4-Y4*Z2)+X4*(Y2*Z3-Y3*Z2)
      B1=-Y3*Z4+Y4*Z3+Y2*Z4-Y4*Z2-Y2*Z3+Y3*Z2
      C1=X3*Z4-X4*Z3-X2*Z4+X4*Z2+X2*Z3-X3*Z2
      D1=-X3*Y4+X4*Y3+X2*Y4-X4*Y2-X2*Y3+X3*Y2
      VOL=(A1+B1*X1+C1*Y1+D1*Z1)/6.0
C
C     STORAGE CHANGE IN ELEMENT
C
      LP=IEL(LT)
      IF (ISTATE(LT).EQ.0) THEN
      XQE=POR2(LT)*SSKE2(LT)*(SUMH-SUMHL)*FSS*VOL/DELT
      SUMQE=SUMQE+XQE
      QSKE(LP)=QSKE(LP)+XQE*DELT
      ENDIF
      IF (ISTATE(LT).EQ.1) THEN
      XQE=POR2(LT)*SSKE2(LT)*(SUMHP-SUMHL)*FSS*VOL/DELT
      SUMQE=SUMQE+XQE
      QSKE(LP)=QSKE(LP)+XQE*DELT
      XQV=POR2(LT)*SSK2(LT)*(SUMH-SUMHP)*FSS*VOL/DELT
      SUMQV=SUMQV+XQV
      QSKV(LP)=QSKV(LP)+XQV*DELT
      ENDIF
 302  CONTINUE
C
C     LAND SUBSIDENCE
C
      IF (NCOL.NE.0 .AND. IECHO5.EQ.1 .AND. IOUT.EQ.1) THEN
      WRITE(IW1,988)
      DO 352 ICOL=1,NCOL
      SUMSKE=0.0
      SUMSKV=0.0
      N=NUM(ICOL)
      DO 351 I=1,N
      LP=COL(ICOL,I)
      SUMSKE=SUMSKE+QSKE(LP)
      SUMSKV=SUMSKV+QSKV(LP)
 351  CONTINUE
      LAND(ICOL)=LAND(ICOL)+(SUMSKE+SUMSKV)/AREA(ICOL)
      IF (IECHO5.EQ.1) WRITE(IW1,989) ICOL,COL(ICOL,1),SUMSKE/
     1   AREA(ICOL),SUMSKV/AREA(ICOL),(SUMSKE+SUMSKV)/AREA(ICOL),
     2   LAND(ICOL)
 352  CONTINUE
      ENDIF
```

```
 988  FORMAT(/1X,'LAND SUBSIDENCE'/1X,15('-')/
     1    1X,4X,'COLUMN',6X,'ELEM',3X,'ELASTIC',1X,'INELASTIC',
     2    6X,'BOTH',5X,'TOTAL'/)
 989  FORMAT((1X,2I10,4F10.3))
C
C     RESET PRE-CONSOLIDATION HEAD
C
      DO 303 I=1,NN
      IF (H(I).LT.HP(I)) HP(I)=H(I)
 303  CONTINUE
      IF (IECHO3.EQ.1 .AND. IOUT.EQ.1) THEN
      WRITE(IW1,939)
      IDONE=0
      I1=-4
 567  I1=I1+5
      I2=I1+4
      IF (I2.GE.NN) THEN
      I2=NN
      IDONE=1
      ENDIF
      WRITE(IW1,940) I1,(HP(I),I=I1,I2)
      IF (IDONE.EQ.0) GO TO 567
      ENDIF
 939  FORMAT(/1X,'PRE-CONSOLIDATION HEADS'/1X,23('-')/
     1    1X,6X,'NODE',4X,'HEADS'/)
 940  FORMAT(1X,I10,5F10.2)
      RETURN
      END
```

```
C     ------------------------------------
      SUBROUTINE FMERGE(FILE1,FILE2,NFILE)
C     ------------------------------------
C
      INCLUDE 'PARAM.FOR'
      IMPLICIT REAL*8 (A-H,O-Z)
      IMPLICIT INTEGER*2 (I-N)
      CHARACTER*60 FILE1(MAXFLS),FILE2
      CHARACTER*90 LINE
C
      OPEN (UNIT=100,FILE=FILE2,STATUS='UNKNOWN')
      DO 220 IFILE=1,NFILE
      OPEN (UNIT=99,FILE=FILE1(IFILE),STATUS='OLD',ERR=210)
 208  READ(99,902,END=218) LINE
      DO 209 ILAST=90,1,-1
      IF (LINE(ILAST:ILAST).NE.' ') GOTO 207
 209  CONTINUE
 207  CONTINUE
      WRITE(100,903) (LINE(I:I),I=1,ILAST)
      GOTO 208
 218  CLOSE (99)
 220  CONTINUE
      CLOSE (100)
      RETURN
 210  WRITE(*,901) FILE1(IFILE)
      STOP
C
 901  FORMAT(1X,'***FILE NOT FOUND ',A60)
 902  FORMAT(A90)
 903  FORMAT(90A1)
      END
```

```
      PARAMETER (MAXPAR=2,MAXWL=2,MAXNN=1000,MAXNE=1000,
     1   MAXNE2=4000,MAXNB=400,MAXETN=2,MAXSTP=2000,MAXRIV=2,MAXRCH=2,
     2   MAXINF=2,MAXCHN=100,MAXTAB=100,MAXSET=100,MAXNUM=40,MAXCOL=100,
     3   MAXVFB=100,MAXFLN=100,MAXFLS=100,MAXSEC=2,MAXPTS=2)
C
C NOTE:  FOR SOME FORTRAN COMPILERS, ALL PARAMETERS MUST BE SET
C        EQUAL TO 2 OR GREATER
C
C     MAXPAR     MAXIMUM NUMBER OF PARAMETERS
C     MAXWL      MAXIMUM NUMBER OF WATER-LEVEL MEASUREMENTS
C     MAXNN      MAXIMUM NUMBER OF NODES
C     MAXNE      MAXIMUM NUMBER OF PRISMATIC ELEMENTS
C     MAXNE2     MAXIMUM NUMBER OF TETRAHEDRAL ELEMENTS
C     MAXNB      MAXIMUM HALF-BAND WIDTH
C     MAXETN     MAXIMUM NUMBER OF ET NODES
C     MAXSTP     MAXIMUM NUMBER OF TIME STEPS
C     MAXRIV     MAXIMUM NUMBER OF RIVER NODES IN REACH
C     MAXINF     MAXIMUM NUMBER OF RIVER INFLOWS
C     MAXRCH     MAXIMUM NUMBER OF RIVER REACHES
C     MAXCHN     MAXIMUM NUMBER OF CONSTANT-HEAD NODES
C     MAXTAB     MAXIMUM NUMBER OF ENTRIES IN CONSTANT-HEAD TABLES
C     MAXSET     MAXIMUM NUMBER OF FLUX DATA SETS
C     MAXCOL     MAXIMUM NUMBER OF COLUMNS
C     MAXNUM     MAXIMUM NUMBER OF NODES OR ELEMENTS IN COLUMN
C     MAXVFB     MAXIMUM NUMBER OF VARIABLE-FLUX NODES
C     MAXFLN     MAXIMUM NUMBER OF FAULT-NODE PAIRS
C     MAXFLS     MAXIMUM NUMBER OF APPENDED FILES
C     MAXSEC     MAXIMUM NUMBER OF TABLES OF CHANNEL GEOMETRY
C     MAXPTS     MAXIMUM NUMBER OF POINTS IN TABLE OF CHANNEL GEOMETRY
C
```

```
      PARAMETER (MAXWEL=2,MAXWST=2,MAXWND=2,MAXWSQ=2,
     1    MAXPCE=2,MAXUSE=2,MAXUST=2,MAXUND=2,
     2    MAXUSQ=2,MAXDCE=2,MAXCON=2,MAXCRP=2,
     3    MAXREC=2)
C
C NOTE:  FOR SOME FORTRAN COMPILERS, ALL PARAMETERS MUST BE SET
C        EQUAL TO 2 OR GREATER
C
C     MAXWEL     MAXIMUM NUMBER OF WELLS
C     MAXWST     MAXIMUM NUMBER OF WELL STATUS CHANGES
C     MAXWND     MAXIMUM NUMBER OF WELL NODES
C     MAXWSQ     MAXIMUM NUMBER OF MONTHLY PUMPING VALUES
C     MAXPCE     MAXIMUM NUMBER OF WELL STATUSES
C     MAXUSE     MAXIMUM NUMBER OF USERS
C     MAXUST     MAXIMUM NUMBER OF USER AREA CHANGES
C     MAXUND     MAXIMUM NUMBER OF USER NODES
C     MAXUSQ     MAXIMUM NUMBER OF MONTHLY DELIVERY VALUES
C     MAXDCE     MAXIMUM NUMBER OF USER TYPES IN CONSTRUCTION
C     MAXCON     MAXIMUM NUMBER OF MONTHS IN CONSTRUCTIONS
C     MAXCRP     MAXIMUM NUMBER OF CROPS
C     MAXREC     MAXIMUM NUMBER OF CROP INVENTORY RECORDS PER USER
C
```

APPENDIX B

EXAMPLE PROBLEM

TRANSIENT, CONFINED, RADIAL FLOW TO A WELL

.0INPUT FILE

```
RANSIENT, CONFINED, RADIAL FLOW TO A WELL
       2         1         1     0.005          1          0          0          1
       1
       1         2         1       1.0
       1         1         0
     414       440                                                    GRID
     1.0       1.0       1.0         1
       1      0.98     -0.21      0.00
       2      0.99     -0.21    -20.00
       3      0.98     -0.21    -40.00
       4      0.98     -0.21    -60.00
       5      0.98     -0.21    -80.00
       6      0.98     -0.21   -100.00
       7      1.00      0.00      0.00
       8      1.01      0.00    -20.00
       9      1.01      0.00    -40.00
      10      1.00      0.00    -60.00
      11      1.00      0.00    -80.00
      12      1.00      0.00   -100.00
      13      0.98      0.21      0.00
      14      0.99      0.21    -20.00
      15      0.98      0.21    -40.00
      16      0.98      0.21    -60.00
      17      0.98      0.21    -80.00
      18      0.98      0.21   -100.00
      19      2.45     -0.52      0.00
      20      2.45     -0.52    -20.00
      21      2.45     -0.52    -40.00
      22      2.45     -0.52    -60.00
      23      2.45     -0.52    -80.00
      24      2.45     -0.52   -100.00
      25      2.50      0.00      0.00
      26      2.51      0.00    -20.00
      27      2.51      0.00    -40.00
      28      2.50      0.00    -60.00
      29      2.50      0.00    -80.00
      30      2.50      0.00   -100.00
      31      2.45      0.52      0.00
      32      2.45      0.52    -20.00
      33      2.45      0.52    -40.00
      34      2.45      0.52    -60.00
      35      2.45      0.52    -80.00
      36      2.45      0.52   -100.00
      37      4.65     -0.99      0.00
      38      4.65     -0.99    -20.00
      39      4.65     -0.99    -40.00
      40      4.65     -0.99    -60.00
      41      4.65     -0.99    -80.00
      42      4.65     -0.99   -100.00
      43      4.75      0.00      0.00
      44      4.76      0.00    -20.00
      45      4.76      0.00    -40.00
      46      4.75      0.00    -60.00
      47      4.75      0.00    -80.00
      48      4.75      0.00   -100.00
      49      4.65      0.99      0.00
      50      4.65      0.99    -20.00
      51      4.65      0.99    -40.00
      52      4.65      0.99    -60.00
      53      4.65      0.99    -80.00
```

54	4.65	0.99	-100.00
55	7.95	-1.69	0.00
56	7.96	-1.69	-20.00
57	7.96	-1.69	-40.00
58	7.96	-1.69	-60.00
59	7.95	-1.69	-80.00
60	7.95	-1.69	-100.00
61	8.13	0.00	0.00
62	8.14	0.00	-20.00
63	8.14	0.00	-40.00
64	8.13	0.00	-60.00
65	8.13	0.00	-80.00
66	8.13	0.00	-100.00
67	7.95	1.69	0.00
68	7.96	1.69	-20.00
69	7.96	1.69	-40.00
70	7.96	1.69	-60.00
71	7.95	1.69	-80.00
72	7.95	1.69	-100.00
73	12.90	-2.74	0.00
74	12.91	-2.75	-20.00
75	12.91	-2.75	-40.00
76	12.91	-2.75	-60.00
77	12.90	-2.74	-80.00
78	12.90	-2.74	-100.00
79	13.19	0.00	0.00
80	13.20	0.00	-20.00
81	13.20	0.00	-40.00
82	13.19	0.00	-60.00
83	13.19	0.00	-80.00
84	13.19	0.00	-100.00
85	12.90	2.74	0.00
86	12.91	2.75	-20.00
87	12.91	2.75	-40.00
88	12.91	2.75	-60.00
89	12.90	2.74	-80.00
90	12.90	2.74	-100.00
91	20.33	-4.32	0.00
92	20.33	-4.32	-20.00
93	20.33	-4.32	-40.00
94	20.33	-4.32	-60.00
95	20.33	-4.32	-80.00
96	20.33	-4.32	-100.00
97	20.78	0.00	0.00
98	20.79	0.00	-20.00
99	20.79	0.00	-40.00
100	20.78	0.00	-60.00
101	20.78	0.00	-80.00
102	20.78	0.00	-100.00
103	20.33	4.32	0.00
104	20.33	4.32	-20.00
105	20.33	4.32	-40.00
106	20.33	4.32	-60.00
107	20.33	4.32	-80.00
108	20.33	4.32	-100.00
109	31.47	-6.69	0.00
110	31.47	-6.69	-20.00
111	31.47	-6.69	-40.00
112	31.47	-6.69	-60.00
113	31.47	-6.69	-80.00
114	31.47	-6.69	-100.00
115	32.17	0.00	0.00

116	32.18	0.00	-20.00
117	32.18	0.00	-40.00
118	32.17	0.00	-60.00
119	32.17	0.00	-80.00
120	32.17	0.00	-100.00
121	31.47	6.69	0.00
122	31.47	6.69	-20.00
123	31.47	6.69	-40.00
124	31.47	6.69	-60.00
125	31.47	6.69	-80.00
126	31.47	6.69	-100.00
127	48.18	-10.24	0.00
128	48.19	-10.25	-20.00
129	48.19	-10.25	-40.00
130	48.19	-10.24	-60.00
131	48.19	-10.24	-80.00
132	48.18	-10.24	-100.00
133	49.26	0.00	0.00
134	49.27	0.00	-20.00
135	49.27	0.00	-40.00
136	49.26	0.00	-60.00
137	49.26	0.00	-80.00
138	49.26	0.00	-100.00
139	48.18	10.24	0.00
140	48.19	10.25	-20.00
141	48.19	10.25	-40.00
142	48.19	10.24	-60.00
143	48.19	10.24	-80.00
144	48.18	10.24	-100.00
145	73.25	-15.57	0.00
146	73.26	-15.57	-20.00
147	73.26	-15.57	-40.00
148	73.26	-15.57	-60.00
149	73.26	-15.57	-80.00
150	73.25	-15.57	-100.00
151	74.89	0.00	0.00
152	74.90	0.00	-20.00
153	74.90	0.00	-40.00
154	74.89	0.00	-60.00
155	74.89	0.00	-80.00
156	74.89	0.00	-100.00
157	73.25	15.57	0.00
158	73.26	15.57	-20.00
159	73.26	15.57	-40.00
160	73.26	15.57	-60.00
161	73.26	15.57	-80.00
162	73.25	15.57	-100.00
163	110.85	-23.56	0.00
164	110.86	-23.57	-20.00
165	110.86	-23.57	-40.00
166	110.86	-23.57	-60.00
167	110.86	-23.57	-80.00
168	110.85	-23.56	-100.00
169	113.33	0.00	0.00
170	113.34	0.00	-20.00
171	113.34	0.00	-40.00
172	113.33	0.00	-60.00
173	113.33	0.00	-80.00
174	113.33	0.00	-100.00
175	110.85	23.56	0.00
176	110.86	23.57	-20.00
177	110.86	23.57	-40.00

178	110.86	23.57	-60.00
179	110.86	23.57	-80.00
180	110.85	23.56	-100.00
181	167.26	-35.55	0.00
182	167.27	-35.56	-20.00
183	167.27	-35.56	-40.00
184	167.27	-35.56	-60.00
185	167.27	-35.56	-80.00
186	167.26	-35.55	-100.00
187	171.00	0.00	0.00
188	171.01	0.00	-20.00
189	171.01	0.00	-40.00
190	171.00	0.00	-60.00
191	171.00	0.00	-80.00
192	171.00	0.00	-100.00
193	167.26	35.55	0.00
194	167.27	35.56	-20.00
195	167.27	35.56	-40.00
196	167.27	35.56	-60.00
197	167.27	35.56	-80.00
198	167.26	35.55	-100.00
199	251.86	-53.54	0.00
200	251.87	-53.54	-20.00
201	251.87	-53.54	-40.00
202	251.87	-53.54	-60.00
203	251.87	-53.54	-80.00
204	251.86	-53.54	-100.00
205	257.49	0.00	0.00
206	257.50	0.00	-20.00
207	257.50	0.00	-40.00
208	257.49	0.00	-60.00
209	257.49	0.00	-80.00
210	257.49	0.00	-100.00
211	251.86	53.54	0.00
212	251.87	53.54	-20.00
213	251.87	53.54	-40.00
214	251.87	53.54	-60.00
215	251.87	53.54	-80.00
216	251.86	53.54	-100.00
217	378.78	-80.51	0.00
218	378.79	-80.52	-20.00
219	378.78	-80.51	-40.00
220	378.78	-80.51	-60.00
221	378.78	-80.51	-80.00
222	378.78	-80.51	-100.00
223	387.24	0.00	0.00
224	387.25	0.00	-20.00
225	387.25	0.00	-40.00
226	387.24	0.00	-60.00
227	387.24	0.00	-80.00
228	387.24	0.00	-100.00
229	378.78	80.51	0.00
230	378.79	80.52	-20.00
231	378.78	80.51	-40.00
232	378.78	80.51	-60.00
233	378.78	80.51	-80.00
234	378.78	80.51	-100.00
235	569.14	-120.98	0.00
236	569.15	-120.98	-20.00
237	569.15	-120.98	-40.00
238	569.15	-120.98	-60.00
239	569.15	-120.98	-80.00

240	569.14	-120.98	-100.00
241	581.86	0.00	0.00
242	581.87	0.00	-20.00
243	581.87	0.00	-40.00
244	581.86	0.00	-60.00
245	581.86	0.00	-80.00
246	581.86	0.00	-100.00
247	569.14	120.98	0.00
248	569.15	120.98	-20.00
249	569.15	120.98	-40.00
250	569.15	120.98	-60.00
251	569.15	120.98	-80.00
252	569.14	120.98	-100.00
253	854.70	-181.67	0.00
254	854.70	-181.67	-20.00
255	854.70	-181.67	-40.00
256	854.70	-181.67	-60.00
257	854.70	-181.67	-80.00
258	854.70	-181.67	-100.00
259	873.79	0.00	0.00
260	873.80	0.00	-20.00
261	873.80	0.00	-40.00
262	873.79	0.00	-60.00
263	873.79	0.00	-80.00
264	873.79	0.00	-100.00
265	854.70	181.67	0.00
266	854.70	181.67	-20.00
267	854.70	181.67	-40.00
268	854.70	181.67	-60.00
269	854.70	181.67	-80.00
270	854.70	181.67	-100.00
271	1283.02	-272.72	0.00
272	1283.02	-272.72	-20.00
273	1283.02	-272.72	-40.00
274	1283.02	-272.72	-60.00
275	1283.02	-272.72	-80.00
276	1283.02	-272.72	-100.00
277	1311.68	0.00	0.00
278	1311.69	0.00	-20.00
279	1311.69	0.00	-40.00
280	1311.68	0.00	-60.00
281	1311.68	0.00	-80.00
282	1311.68	0.00	-100.00
283	1283.02	272.72	0.00
284	1283.02	272.72	-20.00
285	1283.02	272.72	-40.00
286	1283.02	272.72	-60.00
287	1283.02	272.72	-80.00
288	1283.02	272.72	-100.00
289	1925.50	-409.28	0.00
290	1925.51	-409.28	-20.00
291	1925.51	-409.28	-40.00
292	1925.51	-409.28	-60.00
293	1925.51	-409.28	-80.00
294	1925.50	-409.28	-100.00
295	1968.52	0.00	0.00
296	1968.53	0.00	-20.00
297	1968.53	0.00	-40.00
298	1968.52	0.00	-60.00
299	1968.52	0.00	-80.00
300	1968.52	0.00	-100.00
301	1925.50	409.28	0.00

302	1925.51	409.28	-20.00
303	1925.51	409.28	-40.00
304	1925.51	409.28	-60.00
305	1925.51	409.28	-80.00
306	1925.50	409.28	-100.00
307	2889.23	-614.13	0.00
308	2889.24	-614.13	-20.00
309	2889.24	-614.13	-40.00
310	2889.24	-614.13	-60.00
311	2889.23	-614.13	-80.00
312	2889.23	-614.13	-100.00
313	2953.78	0.00	0.00
314	2953.79	0.00	-20.00
315	2953.79	0.00	-40.00
316	2953.78	0.00	-60.00
317	2953.78	0.00	-80.00
318	2953.78	0.00	-100.00
319	2889.23	614.13	0.00
320	2889.24	614.13	-20.00
321	2889.24	614.13	-40.00
322	2889.24	614.13	-60.00
323	2889.23	614.13	-80.00
324	2889.23	614.13	-100.00
325	4334.84	-921.40	0.00
326	4334.85	-921.40	-20.00
327	4334.84	-921.40	-40.00
328	4334.84	-921.40	-60.00
329	4334.84	-921.40	-80.00
330	4334.84	-921.40	-100.00
331	4431.68	0.00	0.00
332	4431.69	0.00	-20.00
333	4431.69	0.00	-40.00
334	4431.68	0.00	-60.00
335	4431.68	0.00	-80.00
336	4431.68	0.00	-100.00
337	4334.84	921.40	0.00
338	4334.85	921.40	-20.00
339	4334.84	921.40	-40.00
340	4334.84	921.40	-60.00
341	4334.84	921.40	-80.00
342	4334.84	921.40	-100.00
343	6503.22	-1382.30	0.00
344	6503.23	-1382.31	-20.00
345	6503.23	-1382.31	-40.00
346	6503.23	-1382.30	-60.00
347	6503.23	-1382.30	-80.00
348	6503.22	-1382.30	-100.00
349	6648.51	0.00	0.00
350	6648.52	0.00	-20.00
351	6648.52	0.00	-40.00
352	6648.51	0.00	-60.00
353	6648.51	0.00	-80.00
354	6648.51	0.00	-100.00
355	6503.22	1382.30	0.00
356	6503.23	1382.31	-20.00
357	6503.23	1382.31	-40.00
358	6503.23	1382.30	-60.00
359	6503.23	1382.30	-80.00
360	6503.22	1382.30	-100.00
361	9755.82	-2073.66	0.00
362	9755.83	-2073.67	-20.00
363	9755.83	-2073.67	-40.00

364	9755.82	-2073.66	-60.00			
365	9755.82	-2073.66	-80.00			
366	9755.82	-2073.66	-100.00			
367	9973.77	0.00	0.00			
368	9973.78	0.00	-20.00			
369	9973.78	0.00	-40.00			
370	9973.77	0.00	-60.00			
371	9973.77	0.00	-80.00			
372	9973.77	0.00	-100.00			
373	9755.82	2073.66	0.00			
374	9755.83	2073.67	-20.00			
375	9755.83	2073.67	-40.00			
376	9755.82	2073.66	-60.00			
377	9755.82	2073.66	-80.00			
378	9755.82	2073.66	-100.00			
379	14634.71	-3110.70	0.00			
380	14634.72	-3110.71	-20.00			
381	14634.72	-3110.71	-40.00			
382	14634.72	-3110.70	-60.00			
383	14634.71	-3110.70	-80.00			
384	14634.71	-3110.70	-100.00			
385	14961.66	0.00	0.00			
386	14961.67	0.00	-20.00			
387	14961.67	0.00	-40.00			
388	14961.66	0.00	-60.00			
389	14961.66	0.00	-80.00			
390	14961.66	0.00	-100.00			
391	14634.71	3110.70	0.00			
392	14634.72	3110.71	-20.00			
393	14634.72	3110.71	-40.00			
394	14634.72	3110.70	-60.00			
395	14634.71	3110.70	-80.00			
396	14634.71	3110.70	-100.00			
397	21953.04	-4666.26	0.00			
398	21953.04	-4666.26	-20.00			
399	21953.04	-4666.26	-40.00			
400	21953.04	-4666.26	-60.00			
401	21953.04	-4666.26	-80.00			
402	21953.04	-4666.26	-100.00			
403	22443.48	0.00	0.00			
404	22443.49	0.00	-20.00			
405	22443.49	0.00	-40.00			
406	22443.48	0.00	-60.00			
407	22443.48	0.00	-80.00			
408	22443.48	0.00	-100.00			
409	21953.04	4666.26	0.00			
410	21953.04	4666.26	-20.00			
411	21953.04	4666.26	-40.00			
412	21953.04	4666.26	-60.00			
413	21953.04	4666.26	-80.00			
414	21953.04	4666.26	-100.00			
1	0					
1	7	1	19	8	2	20
2	7	19	25	8	20	26
3	25	19	37	26	20	38
4	25	37	43	26	38	44
5	43	37	55	44	38	56
6	43	55	61	44	56	62
7	61	55	73	62	56	74
8	61	73	79	62	74	80
9	79	73	91	80	74	92
10	79	91	97	80	92	98

11	97	91	109	98	92	110
12	97	109	115	98	110	116
13	115	109	127	116	110	128
14	115	127	133	116	128	134
15	133	127	145	134	128	146
16	133	145	151	134	146	152
17	151	145	163	152	146	164
18	151	163	169	152	164	170
19	169	163	181	170	164	182
20	169	181	187	170	182	188
21	187	181	199	188	182	200
22	187	199	205	188	200	206
23	205	199	217	206	200	218
24	205	217	223	206	218	224
25	223	217	235	224	218	236
26	223	235	241	224	236	242
27	241	235	253	242	236	254
28	241	253	259	242	254	260
29	259	253	271	260	254	272
30	259	271	277	260	272	278
31	277	271	289	278	272	290
32	277	289	295	278	290	296
33	295	289	307	296	290	308
34	295	307	313	296	308	314
35	313	307	325	314	308	326
36	313	325	331	314	326	332
37	331	325	343	332	326	344
38	331	343	349	332	344	350
39	349	343	361	350	344	362
40	349	361	367	350	362	368
41	367	361	379	368	362	380
42	367	379	385	368	380	386
43	385	379	397	386	380	398
44	385	397	403	386	398	404
45	7	25	31	8	26	32
46	13	7	31	14	8	32
47	25	43	49	26	44	50
48	31	25	49	32	26	50
49	43	61	67	44	62	68
50	49	43	67	50	44	68
51	61	79	85	62	80	86
52	67	61	85	68	62	86
53	79	97	103	80	98	104
54	85	79	103	86	80	104
55	97	115	121	98	116	122
56	103	97	121	104	98	122
57	115	133	139	116	134	140
58	121	115	139	122	116	140
59	133	151	157	134	152	158
60	139	133	157	140	134	158
61	151	169	175	152	170	176
62	157	151	175	158	152	176
63	169	187	193	170	188	194
64	175	169	193	176	170	194
65	187	205	211	188	206	212
66	193	187	211	194	188	212
67	205	223	229	206	224	230
68	211	205	229	212	206	230
69	223	241	247	224	242	248
70	229	223	247	230	224	248
71	241	259	265	242	260	266
72	247	241	265	248	242	266

73	259	277	283	260	278	284
74	265	259	283	266	260	284
75	277	295	301	278	296	302
76	283	277	301	284	278	302
77	295	313	319	296	314	320
78	301	295	319	302	296	320
79	313	331	337	314	332	338
80	319	313	337	320	314	338
81	331	349	355	332	350	356
82	337	331	355	338	332	356
83	349	367	373	350	368	374
84	355	349	373	356	350	374
85	367	385	391	368	386	392
86	373	367	391	374	368	392
87	385	403	409	386	404	410
88	391	385	409	392	386	410
89	8	2	20	9	3	21
90	8	20	26	9	21	27
91	26	20	38	27	21	39
92	26	38	44	27	39	45
93	44	38	56	45	39	57
94	44	56	62	45	57	63
95	62	56	74	63	57	75
96	62	74	80	63	75	81
97	80	74	92	81	75	93
98	80	92	98	81	93	99
99	98	92	110	99	93	111
100	98	110	116	99	111	117
101	116	110	128	117	111	129
102	116	128	134	117	129	135
103	134	128	146	135	129	147
104	134	146	152	135	147	153
105	152	146	164	153	147	165
106	152	164	170	153	165	171
107	170	164	182	171	165	183
108	170	182	188	171	183	189
109	188	182	200	189	183	201
110	188	200	206	189	201	207
111	206	200	218	207	201	219
112	206	218	224	207	219	225
113	224	218	236	225	219	237
114	224	236	242	225	237	243
115	242	236	254	243	237	255
116	242	254	260	243	255	261
117	260	254	272	261	255	273
118	260	272	278	261	273	279
119	278	272	290	279	273	291
120	278	290	296	279	291	297
121	296	290	308	297	291	309
122	296	308	314	297	309	315
123	314	308	326	315	309	327
124	314	326	332	315	327	333
125	332	326	344	333	327	345
126	332	344	350	333	345	351
127	350	344	362	351	345	363
128	350	362	368	351	363	369
129	368	362	380	369	363	381
130	368	380	386	369	381	387
131	386	380	398	387	381	399
132	386	398	404	387	399	405
133	8	26	32	9	27	33
134	14	8	32	15	9	33

135	26	44	50	27	45	51
136	32	26	50	33	27	51
137	44	62	68	45	63	69
138	50	44	68	51	45	69
139	62	80	86	63	81	87
140	68	62	86	69	63	87
141	80	98	104	81	99	105
142	86	80	104	87	81	105
143	98	116	122	99	117	123
144	104	98	122	105	99	123
145	116	134	140	117	135	141
146	122	116	140	123	117	141
147	134	152	158	135	153	159
148	140	134	158	141	135	159
149	152	170	176	153	171	177
150	158	152	176	159	153	177
151	170	188	194	171	189	195
152	176	170	194	177	171	195
153	188	206	212	189	207	213
154	194	188	212	195	189	213
155	206	224	230	207	225	231
156	212	206	230	213	207	231
157	224	242	248	225	243	249
158	230	224	248	231	225	249
159	242	260	266	243	261	267
160	248	242	266	249	243	267
161	260	278	284	261	279	285
162	266	260	284	267	261	285
163	278	296	302	279	297	303
164	284	278	302	285	279	303
165	296	314	320	297	315	321
166	302	296	320	303	297	321
167	314	332	338	315	333	339
168	320	314	338	321	315	339
169	332	350	356	333	351	357
170	338	332	356	339	333	357
171	350	368	374	351	369	375
172	356	350	374	357	351	375
173	368	386	392	369	387	393
174	374	368	392	375	369	393
175	386	404	410	387	405	411
176	392	386	410	393	387	411
177	9	3	21	10	4	22
178	9	21	27	10	22	28
179	27	21	39	28	22	40
180	27	39	45	28	40	46
181	45	39	57	46	40	58
182	45	57	63	46	58	64
183	63	57	75	64	58	76
184	63	75	81	64	76	82
185	81	75	93	82	76	94
186	81	93	99	82	94	100
187	99	93	111	100	94	112
188	99	111	117	100	112	118
189	117	111	129	118	112	130
190	117	129	135	118	130	136
191	135	129	147	136	130	148
192	135	147	153	136	148	154
193	153	147	165	154	148	166
194	153	165	171	154	166	172
195	171	165	183	172	166	184
196	171	183	189	172	184	190

197	189	183	201	190	184	202
198	189	201	207	190	202	208
199	207	201	219	208	202	220
200	207	219	225	208	220	226
201	225	219	237	226	220	238
202	225	237	243	226	238	244
203	243	237	255	244	238	256
204	243	255	261	244	256	262
205	261	255	273	262	256	274
206	261	273	279	262	274	280
207	279	273	291	280	274	292
208	279	291	297	280	292	298
209	297	291	309	298	292	310
210	297	309	315	298	310	316
211	315	309	327	316	310	328
212	315	327	333	316	328	334
213	333	327	345	334	328	346
214	333	345	351	334	346	352
215	351	345	363	352	346	364
216	351	363	369	352	364	370
217	369	363	381	370	364	382
218	369	381	387	370	382	388
219	387	381	399	388	382	400
220	387	399	405	388	400	406
221	9	27	33	10	28	34
222	15	9	33	16	10	34
223	27	45	51	28	46	52
224	33	27	51	34	28	52
225	45	63	69	46	64	70
226	51	45	69	52	46	70
227	63	81	87	64	82	88
228	69	63	87	70	64	88
229	81	99	105	82	100	106
230	87	81	105	88	82	106
231	99	117	123	100	118	124
232	105	99	123	106	100	124
233	117	135	141	118	136	142
234	123	117	141	124	118	142
235	135	153	159	136	154	160
236	141	135	159	142	136	160
237	153	171	177	154	172	178
238	159	153	177	160	154	178
239	171	189	195	172	190	196
240	177	171	195	178	172	196
241	189	207	213	190	208	214
242	195	189	213	196	190	214
243	207	225	231	208	226	232
244	213	207	231	214	208	232
245	225	243	249	226	244	250
246	231	225	249	232	226	250
247	243	261	267	244	262	268
248	249	243	267	250	244	268
249	261	279	285	262	280	286
250	267	261	285	268	262	286
251	279	297	303	280	298	304
252	285	279	303	286	280	304
253	297	315	321	298	316	322
254	303	297	321	304	298	322
255	315	333	339	316	334	340
256	321	315	339	322	316	340
257	333	351	357	334	352	358
258	339	333	357	340	334	358

259	351	369	375	352	370	376
260	357	351	375	358	352	376
261	369	387	393	370	388	394
262	375	369	393	376	370	394
263	387	405	411	388	406	412
264	393	387	411	394	388	412
265	10	4	22	11	5	23
266	10	22	28	11	23	29
267	28	22	40	29	23	41
268	28	40	46	29	41	47
269	46	40	58	47	41	59
270	46	58	64	47	59	65
271	64	58	76	65	59	77
272	64	76	82	65	77	83
273	82	76	94	83	77	95
274	82	94	100	83	95	101
275	100	94	112	101	95	113
276	100	112	118	101	113	119
277	118	112	130	119	113	131
278	118	130	136	119	131	137
279	136	130	148	137	131	149
280	136	148	154	137	149	155
281	154	148	166	155	149	167
282	154	166	172	155	167	173
283	172	166	184	173	167	185
284	172	184	190	173	185	191
285	190	184	202	191	185	203
286	190	202	208	191	203	209
287	208	202	220	209	203	221
288	208	220	226	209	221	227
289	226	220	238	227	221	239
290	226	238	244	227	239	245
291	244	238	256	245	239	257
292	244	256	262	245	257	263
293	262	256	274	263	257	275
294	262	274	280	263	275	281
295	280	274	292	281	275	293
296	280	292	298	281	293	299
297	298	292	310	299	293	311
298	298	310	316	299	311	317
299	316	310	328	317	311	329
300	316	328	334	317	329	335
301	334	328	346	335	329	347
302	334	346	352	335	347	353
303	352	346	364	353	347	365
304	352	364	370	353	365	371
305	370	364	382	371	365	383
306	370	382	388	371	383	389
307	388	382	400	389	383	401
308	388	400	406	389	401	407
309	10	28	34	11	29	35
310	16	10	34	17	11	35
311	28	46	52	29	47	53
312	34	28	52	35	29	53
313	46	64	70	47	65	71
314	52	46	70	53	47	71
315	64	82	88	65	83	89
316	70	64	88	71	65	89
317	82	100	106	83	101	107
318	88	82	106	89	83	107
319	100	118	124	101	119	125
320	106	100	124	107	101	125

321	118	136	142	119	137	143
322	124	118	142	125	119	143
323	136	154	160	137	155	161
324	142	136	160	143	137	161
325	154	172	178	155	173	179
326	160	154	178	161	155	179
327	172	190	196	173	191	197
328	178	172	196	179	173	197
329	190	208	214	191	209	215
330	196	190	214	197	191	215
331	208	226	232	209	227	233
332	214	208	232	215	209	233
333	226	244	250	227	245	251
334	232	226	250	233	227	251
335	244	262	268	245	263	269
336	250	244	268	251	245	269
337	262	280	286	263	281	287
338	268	262	286	269	263	287
339	280	298	304	281	299	305
340	286	280	304	287	281	305
341	298	316	322	299	317	323
342	304	298	322	305	299	323
343	316	334	340	317	335	341
344	322	316	340	323	317	341
345	334	352	358	335	353	359
346	340	334	358	341	335	359
347	352	370	376	353	371	377
348	358	352	376	359	353	377
349	370	388	394	371	389	395
350	376	370	394	377	371	395
351	388	406	412	389	407	413
352	394	388	412	395	389	413
353	11	5	23	12	6	24
354	11	23	29	12	24	30
355	29	23	41	30	24	42
356	29	41	47	30	42	48
357	47	41	59	48	42	60
358	47	59	65	48	60	66
359	65	59	77	66	60	78
360	65	77	83	66	78	84
361	83	77	95	84	78	96
362	83	95	101	84	96	102
363	101	95	113	102	96	114
364	101	113	119	102	114	120
365	119	113	131	120	114	132
366	119	131	137	120	132	138
367	137	131	149	138	132	150
368	137	149	155	138	150	156
369	155	149	167	156	150	168
370	155	167	173	156	168	174
371	173	167	185	174	168	186
372	173	185	191	174	186	192
373	191	185	203	192	186	204
374	191	203	209	192	204	210
375	209	203	221	210	204	222
376	209	221	227	210	222	228
377	227	221	239	228	222	240
378	227	239	245	228	240	246
379	245	239	257	246	240	258
380	245	257	263	246	258	264
381	263	257	275	264	258	276
382	263	275	281	264	276	282

383	281	275	293	282	276	294	
384	281	293	299	282	294	300	
385	299	293	311	300	294	312	
386	299	311	317	300	312	318	
387	317	311	329	318	312	330	
388	317	329	335	318	330	336	
389	335	329	347	336	330	348	
390	335	347	353	336	348	354	
391	353	347	365	354	348	366	
392	353	365	371	354	366	372	
393	371	365	383	372	366	384	
394	371	383	389	372	384	390	
395	389	383	401	390	384	402	
396	389	401	407	390	402	408	
397	11	29	35	12	30	36	
398	17	11	35	18	12	36	
399	29	47	53	30	48	54	
400	35	29	53	36	30	54	
401	47	65	71	48	66	72	
402	53	47	71	54	48	72	
403	65	83	89	66	84	90	
404	71	65	89	72	66	90	
405	83	101	107	84	102	108	
406	89	83	107	90	84	108	
407	101	119	125	102	120	126	
408	107	101	125	108	102	126	
409	119	137	143	120	138	144	
410	125	119	143	126	120	144	
411	137	155	161	138	156	162	
412	143	137	161	144	138	162	
413	155	173	179	156	174	180	
414	161	155	179	162	156	180	
415	173	191	197	174	192	198	
416	179	173	197	180	174	198	
417	191	209	215	192	210	216	
418	197	191	215	198	192	216	
419	209	227	233	210	228	234	
420	215	209	233	216	210	234	
421	227	245	251	228	246	252	
422	233	227	251	234	228	252	
423	245	263	269	246	264	270	
424	251	245	269	252	246	270	
425	263	281	287	264	282	288	
426	269	263	287	270	264	288	
427	281	299	305	282	300	306	
428	287	281	305	288	282	306	
429	299	317	323	300	318	324	
430	305	299	323	306	300	324	
431	317	335	341	318	336	342	
432	323	317	341	324	318	342	
433	335	353	359	336	354	360	
434	341	335	359	342	336	360	
435	353	371	377	354	372	378	
436	359	353	377	360	354	378	
437	371	389	395	372	390	396	
438	377	371	395	378	372	396	
439	389	407	413	390	408	414	
440	395	389	413	396	390	414	
1.0	1.0	1.0	1.0	1.0	0	0	AQUIFER
1	10.0	10.0	10.0	1.0E-04	0.0	0	
2	10.0	10.0	10.0	1.0E-04	0.0	0	
3	10.0	10.0	10.0	1.0E-04	0.0	0	

4	10.0	10.0	10.0	1.0E-04	0.0	0
5	10.0	10.0	10.0	1.0E-04	0.0	0
6	10.0	10.0	10.0	1.0E-04	0.0	0
7	10.0	10.0	10.0	1.0E-04	0.0	0
8	10.0	10.0	10.0	1.0E-04	0.0	0
9	10.0	10.0	10.0	1.0E-04	0.0	0
10	10.0	10.0	10.0	1.0E-04	0.0	0
11	10.0	10.0	10.0	1.0E-04	0.0	0
12	10.0	10.0	10.0	1.0E-04	0.0	0
13	10.0	10.0	10.0	1.0E-04	0.0	0
14	10.0	10.0	10.0	1.0E-04	0.0	0
15	10.0	10.0	10.0	1.0E-04	0.0	0
16	10.0	10.0	10.0	1.0E-04	0.0	0
17	10.0	10.0	10.0	1.0E-04	0.0	0
18	10.0	10.0	10.0	1.0E-04	0.0	0
19	10.0	10.0	10.0	1.0E-04	0.0	0
20	10.0	10.0	10.0	1.0E-04	0.0	0
21	10.0	10.0	10.0	1.0E-04	0.0	0
22	10.0	10.0	10.0	1.0E-04	0.0	0
23	10.0	10.0	10.0	1.0E-04	0.0	0
24	10.0	10.0	10.0	1.0E-04	0.0	0
25	10.0	10.0	10.0	1.0E-04	0.0	0
26	10.0	10.0	10.0	1.0E-04	0.0	0
27	10.0	10.0	10.0	1.0E-04	0.0	0
28	10.0	10.0	10.0	1.0E-04	0.0	0
29	10.0	10.0	10.0	1.0E-04	0.0	0
30	10.0	10.0	10.0	1.0E-04	0.0	0
31	10.0	10.0	10.0	1.0E-04	0.0	0
32	10.0	10.0	10.0	1.0E-04	0.0	0
33	10.0	10.0	10.0	1.0E-04	0.0	0
34	10.0	10.0	10.0	1.0E-04	0.0	0
35	10.0	10.0	10.0	1.0E-04	0.0	0
36	10.0	10.0	10.0	1.0E-04	0.0	0
37	10.0	10.0	10.0	1.0E-04	0.0	0
38	10.0	10.0	10.0	1.0E-04	0.0	0
39	10.0	10.0	10.0	1.0E-04	0.0	0
40	10.0	10.0	10.0	1.0E-04	0.0	0
41	10.0	10.0	10.0	1.0E-04	0.0	0
42	10.0	10.0	10.0	1.0E-04	0.0	0
43	10.0	10.0	10.0	1.0E-04	0.0	0
44	10.0	10.0	10.0	1.0E-04	0.0	0
45	10.0	10.0	10.0	1.0E-04	0.0	0
46	10.0	10.0	10.0	1.0E-04	0.0	0
47	10.0	10.0	10.0	1.0E-04	0.0	0
48	10.0	10.0	10.0	1.0E-04	0.0	0
49	10.0	10.0	10.0	1.0E-04	0.0	0
50	10.0	10.0	10.0	1.0E-04	0.0	0
51	10.0	10.0	10.0	1.0E-04	0.0	0
52	10.0	10.0	10.0	1.0E-04	0.0	0
53	10.0	10.0	10.0	1.0E-04	0.0	0
54	10.0	10.0	10.0	1.0E-04	0.0	0
55	10.0	10.0	10.0	1.0E-04	0.0	0
56	10.0	10.0	10.0	1.0E-04	0.0	0
57	10.0	10.0	10.0	1.0E-04	0.0	0
58	10.0	10.0	10.0	1.0E-04	0.0	0
59	10.0	10.0	10.0	1.0E-04	0.0	0
60	10.0	10.0	10.0	1.0E-04	0.0	0
61	10.0	10.0	10.0	1.0E-04	0.0	0
62	10.0	10.0	10.0	1.0E-04	0.0	0
63	10.0	10.0	10.0	1.0E-04	0.0	0
64	10.0	10.0	10.0	1.0E-04	0.0	0
65	10.0	10.0	10.0	1.0E-04	0.0	0

66	10.0	10.0	10.0	1.0E-04	0.0	0
67	10.0	10.0	10.0	1.0E-04	0.0	0
68	10.0	10.0	10.0	1.0E-04	0.0	0
69	10.0	10.0	10.0	1.0E-04	0.0	0
70	10.0	10.0	10.0	1.0E-04	0.0	0
71	10.0	10.0	10.0	1.0E-04	0.0	0
72	10.0	10.0	10.0	1.0E-04	0.0	0
73	10.0	10.0	10.0	1.0E-04	0.0	0
74	10.0	10.0	10.0	1.0E-04	0.0	0
75	10.0	10.0	10.0	1.0E-04	0.0	0
76	10.0	10.0	10.0	1.0E-04	0.0	0
77	10.0	10.0	10.0	1.0E-04	0.0	0
78	10.0	10.0	10.0	1.0E-04	0.0	0
79	10.0	10.0	10.0	1.0E-04	0.0	0
80	10.0	10.0	10.0	1.0E-04	0.0	0
81	10.0	10.0	10.0	1.0E-04	0.0	0
82	10.0	10.0	10.0	1.0E-04	0.0	0
83	10.0	10.0	10.0	1.0E-04	0.0	0
84	10.0	10.0	10.0	1.0E-04	0.0	0
85	10.0	10.0	10.0	1.0E-04	0.0	0
86	10.0	10.0	10.0	1.0E-04	0.0	0
87	10.0	10.0	10.0	1.0E-04	0.0	0
88	10.0	10.0	10.0	1.0E-04	0.0	0
89	10.0	10.0	10.0	1.0E-04	0.0	0
90	10.0	10.0	10.0	1.0E-04	0.0	0
91	10.0	10.0	10.0	1.0E-04	0.0	0
92	10.0	10.0	10.0	1.0E-04	0.0	0
93	10.0	10.0	10.0	1.0E-04	0.0	0
94	10.0	10.0	10.0	1.0E-04	0.0	0
95	10.0	10.0	10.0	1.0E-04	0.0	0
96	10.0	10.0	10.0	1.0E-04	0.0	0
97	10.0	10.0	10.0	1.0E-04	0.0	0
98	10.0	10.0	10.0	1.0E-04	0.0	0
99	10.0	10.0	10.0	1.0E-04	0.0	0
100	10.0	10.0	10.0	1.0E-04	0.0	0
101	10.0	10.0	10.0	1.0E-04	0.0	0
102	10.0	10.0	10.0	1.0E-04	0.0	0
103	10.0	10.0	10.0	1.0E-04	0.0	0
104	10.0	10.0	10.0	1.0E-04	0.0	0
105	10.0	10.0	10.0	1.0E-04	0.0	0
106	10.0	10.0	10.0	1.0E-04	0.0	0
107	10.0	10.0	10.0	1.0E-04	0.0	0
108	10.0	10.0	10.0	1.0E-04	0.0	0
109	10.0	10.0	10.0	1.0E-04	0.0	0
110	10.0	10.0	10.0	1.0E-04	0.0	0
111	10.0	10.0	10.0	1.0E-04	0.0	0
112	10.0	10.0	10.0	1.0E-04	0.0	0
113	10.0	10.0	10.0	1.0E-04	0.0	0
114	10.0	10.0	10.0	1.0E-04	0.0	0
115	10.0	10.0	10.0	1.0E-04	0.0	0
116	10.0	10.0	10.0	1.0E-04	0.0	0
117	10.0	10.0	10.0	1.0E-04	0.0	0
118	10.0	10.0	10.0	1.0E-04	0.0	0
119	10.0	10.0	10.0	1.0E-04	0.0	0
120	10.0	10.0	10.0	1.0E-04	0.0	0
121	10.0	10.0	10.0	1.0E-04	0.0	0
122	10.0	10.0	10.0	1.0E-04	0.0	0
123	10.0	10.0	10.0	1.0E-04	0.0	0
124	10.0	10.0	10.0	1.0E-04	0.0	0
125	10.0	10.0	10.0	1.0E-04	0.0	0
126	10.0	10.0	10.0	1.0E-04	0.0	0
127	10.0	10.0	10.0	1.0E-04	0.0	0

128	10.0	10.0	10.0	1.0E-04	0.0	0
129	10.0	10.0	10.0	1.0E-04	0.0	0
130	10.0	10.0	10.0	1.0E-04	0.0	0
131	10.0	10.0	10.0	1.0E-04	0.0	0
132	10.0	10.0	10.0	1.0E-04	0.0	0
133	10.0	10.0	10.0	1.0E-04	0.0	0
134	10.0	10.0	10.0	1.0E-04	0.0	0
135	10.0	10.0	10.0	1.0E-04	0.0	0
136	10.0	10.0	10.0	1.0E-04	0.0	0
137	10.0	10.0	10.0	1.0E-04	0.0	0
138	10.0	10.0	10.0	1.0E-04	0.0	0
139	10.0	10.0	10.0	1.0E-04	0.0	0
140	10.0	10.0	10.0	1.0E-04	0.0	0
141	10.0	10.0	10.0	1.0E-04	0.0	0
142	10.0	10.0	10.0	1.0E-04	0.0	0
143	10.0	10.0	10.0	1.0E-04	0.0	0
144	10.0	10.0	10.0	1.0E-04	0.0	0
145	10.0	10.0	10.0	1.0E-04	0.0	0
146	10.0	10.0	10.0	1.0E-04	0.0	0
147	10.0	10.0	10.0	1.0E-04	0.0	0
148	10.0	10.0	10.0	1.0E-04	0.0	0
149	10.0	10.0	10.0	1.0E-04	0.0	0
150	10.0	10.0	10.0	1.0E-04	0.0	0
151	10.0	10.0	10.0	1.0E-04	0.0	0
152	10.0	10.0	10.0	1.0E-04	0.0	0
153	10.0	10.0	10.0	1.0E-04	0.0	0
154	10.0	10.0	10.0	1.0E-04	0.0	0
155	10.0	10.0	10.0	1.0E-04	0.0	0
156	10.0	10.0	10.0	1.0E-04	0.0	0
157	10.0	10.0	10.0	1.0E-04	0.0	0
158	10.0	10.0	10.0	1.0E-04	0.0	0
159	10.0	10.0	10.0	1.0E-04	0.0	0
160	10.0	10.0	10.0	1.0E-04	0.0	0
161	10.0	10.0	10.0	1.0E-04	0.0	0
162	10.0	10.0	10.0	1.0E-04	0.0	0
163	10.0	10.0	10.0	1.0E-04	0.0	0
164	10.0	10.0	10.0	1.0E-04	0.0	0
165	10.0	10.0	10.0	1.0E-04	0.0	0
166	10.0	10.0	10.0	1.0E-04	0.0	0
167	10.0	10.0	10.0	1.0E-04	0.0	0
168	10.0	10.0	10.0	1.0E-04	0.0	0
169	10.0	10.0	10.0	1.0E-04	0.0	0
170	10.0	10.0	10.0	1.0E-04	0.0	0
171	10.0	10.0	10.0	1.0E-04	0.0	0
172	10.0	10.0	10.0	1.0E-04	0.0	0
173	10.0	10.0	10.0	1.0E-04	0.0	0
174	10.0	10.0	10.0	1.0E-04	0.0	0
175	10.0	10.0	10.0	1.0E-04	0.0	0
176	10.0	10.0	10.0	1.0E-04	0.0	0
177	10.0	10.0	10.0	1.0E-04	0.0	0
178	10.0	10.0	10.0	1.0E-04	0.0	0
179	10.0	10.0	10.0	1.0E-04	0.0	0
180	10.0	10.0	10.0	1.0E-04	0.0	0
181	10.0	10.0	10.0	1.0E-04	0.0	0
182	10.0	10.0	10.0	1.0E-04	0.0	0
183	10.0	10.0	10.0	1.0E-04	0.0	0
184	10.0	10.0	10.0	1.0E-04	0.0	0
185	10.0	10.0	10.0	1.0E-04	0.0	0
186	10.0	10.0	10.0	1.0E-04	0.0	0
187	10.0	10.0	10.0	1.0E-04	0.0	0
188	10.0	10.0	10.0	1.0E-04	0.0	0
189	10.0	10.0	10.0	1.0E-04	0.0	0

190	10.0	10.0	10.0	1.0E-04	0.0	0
191	10.0	10.0	10.0	1.0E-04	0.0	0
192	10.0	10.0	10.0	1.0E-04	0.0	0
193	10.0	10.0	10.0	1.0E-04	0.0	0
194	10.0	10.0	10.0	1.0E-04	0.0	0
195	10.0	10.0	10.0	1.0E-04	0.0	0
196	10.0	10.0	10.0	1.0E-04	0.0	0
197	10.0	10.0	10.0	1.0E-04	0.0	0
198	10.0	10.0	10.0	1.0E-04	0.0	0
199	10.0	10.0	10.0	1.0E-04	0.0	0
200	10.0	10.0	10.0	1.0E-04	0.0	0
201	10.0	10.0	10.0	1.0E-04	0.0	0
202	10.0	10.0	10.0	1.0E-04	0.0	0
203	10.0	10.0	10.0	1.0E-04	0.0	0
204	10.0	10.0	10.0	1.0E-04	0.0	0
205	10.0	10.0	10.0	1.0E-04	0.0	0
206	10.0	10.0	10.0	1.0E-04	0.0	0
207	10.0	10.0	10.0	1.0E-04	0.0	0
208	10.0	10.0	10.0	1.0E-04	0.0	0
209	10.0	10.0	10.0	1.0E-04	0.0	0
210	10.0	10.0	10.0	1.0E-04	0.0	0
211	10.0	10.0	10.0	1.0E-04	0.0	0
212	10.0	10.0	10.0	1.0E-04	0.0	0
213	10.0	10.0	10.0	1.0E-04	0.0	0
214	10.0	10.0	10.0	1.0E-04	0.0	0
215	10.0	10.0	10.0	1.0E-04	0.0	0
216	10.0	10.0	10.0	1.0E-04	0.0	0
217	10.0	10.0	10.0	1.0E-04	0.0	0
218	10.0	10.0	10.0	1.0E-04	0.0	0
219	10.0	10.0	10.0	1.0E-04	0.0	0
220	10.0	10.0	10.0	1.0E-04	0.0	0
221	10.0	10.0	10.0	1.0E-04	0.0	0
222	10.0	10.0	10.0	1.0E-04	0.0	0
223	10.0	10.0	10.0	1.0E-04	0.0	0
224	10.0	10.0	10.0	1.0E-04	0.0	0
225	10.0	10.0	10.0	1.0E-04	0.0	0
226	10.0	10.0	10.0	1.0E-04	0.0	0
227	10.0	10.0	10.0	1.0E-04	0.0	0
228	10.0	10.0	10.0	1.0E-04	0.0	0
229	10.0	10.0	10.0	1.0E-04	0.0	0
230	10.0	10.0	10.0	1.0E-04	0.0	0
231	10.0	10.0	10.0	1.0E-04	0.0	0
232	10.0	10.0	10.0	1.0E-04	0.0	0
233	10.0	10.0	10.0	1.0E-04	0.0	0
234	10.0	10.0	10.0	1.0E-04	0.0	0
235	10.0	10.0	10.0	1.0E-04	0.0	0
236	10.0	10.0	10.0	1.0E-04	0.0	0
237	10.0	10.0	10.0	1.0E-04	0.0	0
238	10.0	10.0	10.0	1.0E-04	0.0	0
239	10.0	10.0	10.0	1.0E-04	0.0	0
240	10.0	10.0	10.0	1.0E-04	0.0	0
241	10.0	10.0	10.0	1.0E-04	0.0	0
242	10.0	10.0	10.0	1.0E-04	0.0	0
243	10.0	10.0	10.0	1.0E-04	0.0	0
244	10.0	10.0	10.0	1.0E-04	0.0	0
245	10.0	10.0	10.0	1.0E-04	0.0	0
246	10.0	10.0	10.0	1.0E-04	0.0	0
247	10.0	10.0	10.0	1.0E-04	0.0	0
248	10.0	10.0	10.0	1.0E-04	0.0	0
249	10.0	10.0	10.0	1.0E-04	0.0	0
250	10.0	10.0	10.0	1.0E-04	0.0	0
251	10.0	10.0	10.0	1.0E-04	0.0	0

252	10.0	10.0	10.0	1.0E-04	0.0	0
253	10.0	10.0	10.0	1.0E-04	0.0	0
254	10.0	10.0	10.0	1.0E-04	0.0	0
255	10.0	10.0	10.0	1.0E-04	0.0	0
256	10.0	10.0	10.0	1.0E-04	0.0	0
257	10.0	10.0	10.0	1.0E-04	0.0	0
258	10.0	10.0	10.0	1.0E-04	0.0	0
259	10.0	10.0	10.0	1.0E-04	0.0	0
260	10.0	10.0	10.0	1.0E-04	0.0	0
261	10.0	10.0	10.0	1.0E-04	0.0	0
262	10.0	10.0	10.0	1.0E-04	0.0	0
263	10.0	10.0	10.0	1.0E-04	0.0	0
264	10.0	10.0	10.0	1.0E-04	0.0	0
265	10.0	10.0	10.0	1.0E-04	0.0	0
266	10.0	10.0	10.0	1.0E-04	0.0	0
267	10.0	10.0	10.0	1.0E-04	0.0	0
268	10.0	10.0	10.0	1.0E-04	0.0	0
269	10.0	10.0	10.0	1.0E-04	0.0	0
270	10.0	10.0	10.0	1.0E-04	0.0	0
271	10.0	10.0	10.0	1.0E-04	0.0	0
272	10.0	10.0	10.0	1.0E-04	0.0	0
273	10.0	10.0	10.0	1.0E-04	0.0	0
274	10.0	10.0	10.0	1.0E-04	0.0	0
275	10.0	10.0	10.0	1.0E-04	0.0	0
276	10.0	10.0	10.0	1.0E-04	0.0	0
277	10.0	10.0	10.0	1.0E-04	0.0	0
278	10.0	10.0	10.0	1.0E-04	0.0	0
279	10.0	10.0	10.0	1.0E-04	0.0	0
280	10.0	10.0	10.0	1.0E-04	0.0	0
281	10.0	10.0	10.0	1.0E-04	0.0	0
282	10.0	10.0	10.0	1.0E-04	0.0	0
283	10.0	10.0	10.0	1.0E-04	0.0	0
284	10.0	10.0	10.0	1.0E-04	0.0	0
285	10.0	10.0	10.0	1.0E-04	0.0	0
286	10.0	10.0	10.0	1.0E-04	0.0	0
287	10.0	10.0	10.0	1.0E-04	0.0	0
288	10.0	10.0	10.0	1.0E-04	0.0	0
289	10.0	10.0	10.0	1.0E-04	0.0	0
290	10.0	10.0	10.0	1.0E-04	0.0	0
291	10.0	10.0	10.0	1.0E-04	0.0	0
292	10.0	10.0	10.0	1.0E-04	0.0	0
293	10.0	10.0	10.0	1.0E-04	0.0	0
294	10.0	10.0	10.0	1.0E-04	0.0	0
295	10.0	10.0	10.0	1.0E-04	0.0	0
296	10.0	10.0	10.0	1.0E-04	0.0	0
297	10.0	10.0	10.0	1.0E-04	0.0	0
298	10.0	10.0	10.0	1.0E-04	0.0	0
299	10.0	10.0	10.0	1.0E-04	0.0	0
300	10.0	10.0	10.0	1.0E-04	0.0	0
301	10.0	10.0	10.0	1.0E-04	0.0	0
302	10.0	10.0	10.0	1.0E-04	0.0	0
303	10.0	10.0	10.0	1.0E-04	0.0	0
304	10.0	10.0	10.0	1.0E-04	0.0	0
305	10.0	10.0	10.0	1.0E-04	0.0	0
306	10.0	10.0	10.0	1.0E-04	0.0	0
307	10.0	10.0	10.0	1.0E-04	0.0	0
308	10.0	10.0	10.0	1.0E-04	0.0	0
309	10.0	10.0	10.0	1.0E-04	0.0	0
310	10.0	10.0	10.0	1.0E-04	0.0	0
311	10.0	10.0	10.0	1.0E-04	0.0	0
312	10.0	10.0	10.0	1.0E-04	0.0	0
313	10.0	10.0	10.0	1.0E-04	0.0	0

314	10.0	10.0	10.0	1.0E-04	0.0	0
315	10.0	10.0	10.0	1.0E-04	0.0	0
316	10.0	10.0	10.0	1.0E-04	0.0	0
317	10.0	10.0	10.0	1.0E-04	0.0	0
318	10.0	10.0	10.0	1.0E-04	0.0	0
319	10.0	10.0	10.0	1.0E-04	0.0	0
320	10.0	10.0	10.0	1.0E-04	0.0	0
321	10.0	10.0	10.0	1.0E-04	0.0	0
322	10.0	10.0	10.0	1.0E-04	0.0	0
323	10.0	10.0	10.0	1.0E-04	0.0	0
324	10.0	10.0	10.0	1.0E-04	0.0	0
325	10.0	10.0	10.0	1.0E-04	0.0	0
326	10.0	10.0	10.0	1.0E-04	0.0	0
327	10.0	10.0	10.0	1.0E-04	0.0	0
328	10.0	10.0	10.0	1.0E-04	0.0	0
329	10.0	10.0	10.0	1.0E-04	0.0	0
330	10.0	10.0	10.0	1.0E-04	0.0	0
331	10.0	10.0	10.0	1.0E-04	0.0	0
332	10.0	10.0	10.0	1.0E-04	0.0	0
333	10.0	10.0	10.0	1.0E-04	0.0	0
334	10.0	10.0	10.0	1.0E-04	0.0	0
335	10.0	10.0	10.0	1.0E-04	0.0	0
336	10.0	10.0	10.0	1.0E-04	0.0	0
337	10.0	10.0	10.0	1.0E-04	0.0	0
338	10.0	10.0	10.0	1.0E-04	0.0	0
339	10.0	10.0	10.0	1.0E-04	0.0	0
340	10.0	10.0	10.0	1.0E-04	0.0	0
341	10.0	10.0	10.0	1.0E-04	0.0	0
342	10.0	10.0	10.0	1.0E-04	0.0	0
343	10.0	10.0	10.0	1.0E-04	0.0	0
344	10.0	10.0	10.0	1.0E-04	0.0	0
345	10.0	10.0	10.0	1.0E-04	0.0	0
346	10.0	10.0	10.0	1.0E-04	0.0	0
347	10.0	10.0	10.0	1.0E-04	0.0	0
348	10.0	10.0	10.0	1.0E-04	0.0	0
349	10.0	10.0	10.0	1.0E-04	0.0	0
350	10.0	10.0	10.0	1.0E-04	0.0	0
351	10.0	10.0	10.0	1.0E-04	0.0	0
352	10.0	10.0	10.0	1.0E-04	0.0	0
353	10.0	10.0	10.0	1.0E-04	0.0	0
354	10.0	10.0	10.0	1.0E-04	0.0	0
355	10.0	10.0	10.0	1.0E-04	0.0	0
356	10.0	10.0	10.0	1.0E-04	0.0	0
357	10.0	10.0	10.0	1.0E-04	0.0	0
358	10.0	10.0	10.0	1.0E-04	0.0	0
359	10.0	10.0	10.0	1.0E-04	0.0	0
360	10.0	10.0	10.0	1.0E-04	0.0	0
361	10.0	10.0	10.0	1.0E-04	0.0	0
362	10.0	10.0	10.0	1.0E-04	0.0	0
363	10.0	10.0	10.0	1.0E-04	0.0	0
364	10.0	10.0	10.0	1.0E-04	0.0	0
365	10.0	10.0	10.0	1.0E-04	0.0	0
366	10.0	10.0	10.0	1.0E-04	0.0	0
367	10.0	10.0	10.0	1.0E-04	0.0	0
368	10.0	10.0	10.0	1.0E-04	0.0	0
369	10.0	10.0	10.0	1.0E-04	0.0	0
370	10.0	10.0	10.0	1.0E-04	0.0	0
371	10.0	10.0	10.0	1.0E-04	0.0	0
372	10.0	10.0	10.0	1.0E-04	0.0	0
373	10.0	10.0	10.0	1.0E-04	0.0	0
374	10.0	10.0	10.0	1.0E-04	0.0	0
375	10.0	10.0	10.0	1.0E-04	0.0	0

376	10.0	10.0	10.0	1.0E-04	0.0	0
377	10.0	10.0	10.0	1.0E-04	0.0	0
378	10.0	10.0	10.0	1.0E-04	0.0	0
379	10.0	10.0	10.0	1.0E-04	0.0	0
380	10.0	10.0	10.0	1.0E-04	0.0	0
381	10.0	10.0	10.0	1.0E-04	0.0	0
382	10.0	10.0	10.0	1.0E-04	0.0	0
383	10.0	10.0	10.0	1.0E-04	0.0	0
384	10.0	10.0	10.0	1.0E-04	0.0	0
385	10.0	10.0	10.0	1.0E-04	0.0	0
386	10.0	10.0	10.0	1.0E-04	0.0	0
387	10.0	10.0	10.0	1.0E-04	0.0	0
388	10.0	10.0	10.0	1.0E-04	0.0	0
389	10.0	10.0	10.0	1.0E-04	0.0	0
390	10.0	10.0	10.0	1.0E-04	0.0	0
391	10.0	10.0	10.0	1.0E-04	0.0	0
392	10.0	10.0	10.0	1.0E-04	0.0	0
393	10.0	10.0	10.0	1.0E-04	0.0	0
394	10.0	10.0	10.0	1.0E-04	0.0	0
395	10.0	10.0	10.0	1.0E-04	0.0	0
396	10.0	10.0	10.0	1.0E-04	0.0	0
397	10.0	10.0	10.0	1.0E-04	0.0	0
398	10.0	10.0	10.0	1.0E-04	0.0	0
399	10.0	10.0	10.0	1.0E-04	0.0	0
400	10.0	10.0	10.0	1.0E-04	0.0	0
401	10.0	10.0	10.0	1.0E-04	0.0	0
402	10.0	10.0	10.0	1.0E-04	0.0	0
403	10.0	10.0	10.0	1.0E-04	0.0	0
404	10.0	10.0	10.0	1.0E-04	0.0	0
405	10.0	10.0	10.0	1.0E-04	0.0	0
406	10.0	10.0	10.0	1.0E-04	0.0	0
407	10.0	10.0	10.0	1.0E-04	0.0	0
408	10.0	10.0	10.0	1.0E-04	0.0	0
409	10.0	10.0	10.0	1.0E-04	0.0	0
410	10.0	10.0	10.0	1.0E-04	0.0	0
411	10.0	10.0	10.0	1.0E-04	0.0	0
412	10.0	10.0	10.0	1.0E-04	0.0	0
413	10.0	10.0	10.0	1.0E-04	0.0	0
414	10.0	10.0	10.0	1.0E-04	0.0	0
415	10.0	10.0	10.0	1.0E-04	0.0	0
416	10.0	10.0	10.0	1.0E-04	0.0	0
417	10.0	10.0	10.0	1.0E-04	0.0	0
418	10.0	10.0	10.0	1.0E-04	0.0	0
419	10.0	10.0	10.0	1.0E-04	0.0	0
420	10.0	10.0	10.0	1.0E-04	0.0	0
421	10.0	10.0	10.0	1.0E-04	0.0	0
422	10.0	10.0	10.0	1.0E-04	0.0	0
423	10.0	10.0	10.0	1.0E-04	0.0	0
424	10.0	10.0	10.0	1.0E-04	0.0	0
425	10.0	10.0	10.0	1.0E-04	0.0	0
426	10.0	10.0	10.0	1.0E-04	0.0	0
427	10.0	10.0	10.0	1.0E-04	0.0	0
428	10.0	10.0	10.0	1.0E-04	0.0	0
429	10.0	10.0	10.0	1.0E-04	0.0	0
430	10.0	10.0	10.0	1.0E-04	0.0	0
431	10.0	10.0	10.0	1.0E-04	0.0	0
432	10.0	10.0	10.0	1.0E-04	0.0	0
433	10.0	10.0	10.0	1.0E-04	0.0	0
434	10.0	10.0	10.0	1.0E-04	0.0	0
435	10.0	10.0	10.0	1.0E-04	0.0	0
436	10.0	10.0	10.0	1.0E-04	0.0	0
437	10.0	10.0	10.0	1.0E-04	0.0	0

```
 438      10.0      10.0      10.0   1.0E-04      0.0      0
 439      10.0      10.0      10.0   1.0E-04      0.0      0
 440      10.0      10.0      10.0   1.0E-04      0.0      0
   1                                                         HEAD
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
 0.0       0.0       0.0       0.0       0.0
```

```
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0       0.0
0.0       0.0       0.0       0.0
  0                                                           CHEAD
  1         2       1.0         1         1                   FLUX
  1
  7    -0.297
  1       1.0
  2       1.0
  0                                                           RIVER
  0                                                           EVAP
  0                                                           VFLUX
 17         0         0                                       FAULT
  1         2  8.64E+05         1         1
  2         3  8.64E+05         1         1
  3         4  8.64E+05         1         1
  4         5  8.64E+05         1         1
  5         6  8.64E+05         1         1
  7         8  8.64E+05         1         1
  8         9  8.64E+05         1         1
  9        10  8.64E+05         1         1
 10        11  8.64E+05         1         1
 11        12  8.64E+05         1         1
 13        14  8.64E+05         1         1
 14        15  8.64E+05         1         1
 15        16  8.64E+05         1         1
 16        17  8.64E+05         1         1
 17        18  8.64E+05         1         1
  1         7  8.64E+05         1         1
  7        13  8.64E+05         1         1
  0                                                           SINK
  0                                                           WATER
```

2.0 OUTPUT FILES

```
TRANSIENT, CONFINED, RADIAL FLOW TO A WELL
--------------------------------------------------------------------------------

SIMULATION PARAMETERS
---------------------
NUMBER OF TIME STEPS                                 2
TIME-STEP SCHEME                                     1
NUMBER OF ITERATIONS                                 1
CLOSURE CRITERION FOR ITERATIONS              5.00E-03
SWITCH TO SKIP INTEGRATION UPDATE                    1
SWITCH TO FIT MODEL TO DATA                          0
SWITCH FOR STEADY STATE                              0
SWITCH FOR SOLUTION METHOD                           1


TIME-STEP CONFIGURATION
-----------------------

NUMBER OF RESTARTS            1

      STEP       KNS1       KNS2       DELT      FDELT

         1          1          2   1.00E+00   1.00E+00
         2          1          2   1.00E+00   1.00E+00

OUTPUT INTERVALS
----------------
PRINT INTERVAL                                       1
PLOT INTERVAL                                        1
FLUX INTERVAL                                        0

GRID SIZE
---------
NUMBER OF NODES                                    414
NUMBER OF ELEMENTS                                 440

NODE COORDINATES
----------------
FACTOR FOR X                                  1.00E+00
FACTOR FOR Y                                  1.00E+00
FACTOR FOR Z                                  1.00E+00
SWITCH FOR ECHO OF COORDINATES                       1

      NODE          X          Y          Z

         1       0.98      -0.21       0.00
         2       0.99      -0.21     -20.00
         3       0.98      -0.21     -40.00
         4       0.98      -0.21     -60.00
         5       0.98      -0.21     -80.00
         6       0.98      -0.21    -100.00
         7       1.00       0.00       0.00
         8       1.01       0.00     -20.00
         9       1.01       0.00     -40.00
        10       1.00       0.00     -60.00
        11       1.00       0.00     -80.00
        12       1.00       0.00    -100.00
        13       0.98       0.21       0.00
        14       0.99       0.21     -20.00
        15       0.98       0.21     -40.00
```

16	0.98	0.21	-60.00
17	0.98	0.21	-80.00
18	0.98	0.21	-100.00
19	2.45	-0.52	0.00
20	2.45	-0.52	-20.00
21	2.45	-0.52	-40.00
22	2.45	-0.52	-60.00
23	2.45	-0.52	-80.00
24	2.45	-0.52	-100.00
25	2.50	0.00	0.00
26	2.51	0.00	-20.00
27	2.51	0.00	-40.00
28	2.50	0.00	-60.00
29	2.50	0.00	-80.00
30	2.50	0.00	-100.00
31	2.45	0.52	0.00
32	2.45	0.52	-20.00
33	2.45	0.52	-40.00
34	2.45	0.52	-60.00
35	2.45	0.52	-80.00
36	2.45	0.52	-100.00
37	4.65	-0.99	0.00
38	4.65	-0.99	-20.00
39	4.65	-0.99	-40.00
40	4.65	-0.99	-60.00
41	4.65	-0.99	-80.00
42	4.65	-0.99	-100.00
43	4.75	0.00	0.00
44	4.76	0.00	-20.00
45	4.76	0.00	-40.00
46	4.75	0.00	-60.00
47	4.75	0.00	-80.00
48	4.75	0.00	-100.00
49	4.65	0.99	0.00
50	4.65	0.99	-20.00
51	4.65	0.99	-40.00
52	4.65	0.99	-60.00
53	4.65	0.99	-80.00
54	4.65	0.99	-100.00
55	7.95	-1.69	0.00
56	7.96	-1.69	-20.00
57	7.96	-1.69	-40.00
58	7.96	-1.69	-60.00
59	7.95	-1.69	-80.00
60	7.95	-1.69	-100.00
61	8.13	0.00	0.00
62	8.14	0.00	-20.00
63	8.14	0.00	-40.00
64	8.13	0.00	-60.00
65	8.13	0.00	-80.00
66	8.13	0.00	-100.00
67	7.95	1.69	0.00
68	7.96	1.69	-20.00
69	7.96	1.69	-40.00
70	7.96	1.69	-60.00
71	7.95	1.69	-80.00
72	7.95	1.69	-100.00
73	12.90	-2.74	0.00
74	12.91	-2.75	-20.00
75	12.91	-2.75	-40.00
76	12.91	-2.75	-60.00
77	12.90	-2.74	-80.00

78	12.90	-2.74	-100.00
79	13.19	0.00	0.00
80	13.20	0.00	-20.00
81	13.20	0.00	-40.00
82	13.19	0.00	-60.00
83	13.19	0.00	-80.00
84	13.19	0.00	-100.00
85	12.90	2.74	0.00
86	12.91	2.75	-20.00
87	12.91	2.75	-40.00
88	12.91	2.75	-60.00
89	12.90	2.74	-80.00
90	12.90	2.74	-100.00
91	20.33	-4.32	0.00
92	20.33	-4.32	-20.00
93	20.33	-4.32	-40.00
94	20.33	-4.32	-60.00
95	20.33	-4.32	-80.00
96	20.33	-4.32	-100.00
97	20.78	0.00	0.00
98	20.79	0.00	-20.00
99	20.79	0.00	-40.00
100	20.78	0.00	-60.00
101	20.78	0.00	-80.00
102	20.78	0.00	-100.00
103	20.33	4.32	0.00
104	20.33	4.32	-20.00
105	20.33	4.32	-40.00
106	20.33	4.32	-60.00
107	20.33	4.32	-80.00
108	20.33	4.32	-100.00
109	31.47	-6.69	0.00
110	31.47	-6.69	-20.00
111	31.47	-6.69	-40.00
112	31.47	-6.69	-60.00
113	31.47	-6.69	-80.00
114	31.47	-6.69	-100.00
115	32.17	0.00	0.00
116	32.18	0.00	-20.00
117	32.18	0.00	-40.00
118	32.17	0.00	-60.00
119	32.17	0.00	-80.00
120	32.17	0.00	-100.00
121	31.47	6.69	0.00
122	31.47	6.69	-20.00
123	31.47	6.69	-40.00
124	31.47	6.69	-60.00
125	31.47	6.69	-80.00
126	31.47	6.69	-100.00
127	48.18	-10.24	0.00
128	48.19	-10.25	-20.00
129	48.19	-10.25	-40.00
130	48.19	-10.24	-60.00
131	48.19	-10.24	-80.00
132	48.18	-10.24	-100.00
133	49.26	0.00	0.00
134	49.27	0.00	-20.00
135	49.27	0.00	-40.00
136	49.26	0.00	-60.00
137	49.26	0.00	-80.00
138	49.26	0.00	-100.00
139	48.18	10.24	0.00

140	48.19	10.25	-20.00
141	48.19	10.25	-40.00
142	48.19	10.24	-60.00
143	48.19	10.24	-80.00
144	48.18	10.24	-100.00
145	73.25	-15.57	0.00
146	73.26	-15.57	-20.00
147	73.26	-15.57	-40.00
148	73.26	-15.57	-60.00
149	73.26	-15.57	-80.00
150	73.25	-15.57	-100.00
151	74.89	0.00	0.00
152	74.90	0.00	-20.00
153	74.90	0.00	-40.00
154	74.89	0.00	-60.00
155	74.89	0.00	-80.00
156	74.89	0.00	-100.00
157	73.25	15.57	0.00
158	73.26	15.57	-20.00
159	73.26	15.57	-40.00
160	73.26	15.57	-60.00
161	73.26	15.57	-80.00
162	73.25	15.57	-100.00
163	110.85	-23.56	0.00
164	110.86	-23.57	-20.00
165	110.86	-23.57	-40.00
166	110.86	-23.57	-60.00
167	110.86	-23.57	-80.00
168	110.85	-23.56	-100.00
169	113.33	0.00	0.00
170	113.34	0.00	-20.00
171	113.34	0.00	-40.00
172	113.33	0.00	-60.00
173	113.33	0.00	-80.00
174	113.33	0.00	-100.00
175	110.85	23.56	0.00
176	110.86	23.57	-20.00
177	110.86	23.57	-40.00
178	110.86	23.57	-60.00
179	110.86	23.57	-80.00
180	110.85	23.56	-100.00
181	167.26	-35.55	0.00
182	167.27	-35.56	-20.00
183	167.27	-35.56	-40.00
184	167.27	-35.56	-60.00
185	167.27	-35.56	-80.00
186	167.26	-35.55	-100.00
187	171.00	0.00	0.00
188	171.01	0.00	-20.00
189	171.01	0.00	-40.00
190	171.00	0.00	-60.00
191	171.00	0.00	-80.00
192	171.00	0.00	-100.00
193	167.26	35.55	0.00
194	167.27	35.56	-20.00
195	167.27	35.56	-40.00
196	167.27	35.56	-60.00
197	167.27	35.56	-80.00
198	167.26	35.55	-100.00
199	251.86	-53.54	0.00
200	251.87	-53.54	-20.00
201	251.87	-53.54	-40.00

202	251.87	-53.54	-60.00
203	251.87	-53.54	-80.00
204	251.86	-53.54	-100.00
205	257.49	0.00	0.00
206	257.50	0.00	-20.00
207	257.50	0.00	-40.00
208	257.49	0.00	-60.00
209	257.49	0.00	-80.00
210	257.49	0.00	-100.00
211	251.86	53.54	0.00
212	251.87	53.54	-20.00
213	251.87	53.54	-40.00
214	251.87	53.54	-60.00
215	251.87	53.54	-80.00
216	251.86	53.54	-100.00
217	378.78	-80.51	0.00
218	378.79	-80.52	-20.00
219	378.78	-80.51	-40.00
220	378.78	-80.51	-60.00
221	378.78	-80.51	-80.00
222	378.78	-80.51	-100.00
223	387.24	0.00	0.00
224	387.25	0.00	-20.00
225	387.25	0.00	-40.00
226	387.24	0.00	-60.00
227	387.24	0.00	-80.00
228	387.24	0.00	-100.00
229	378.78	80.51	0.00
230	378.79	80.52	-20.00
231	378.78	80.51	-40.00
232	378.78	80.51	-60.00
233	378.78	80.51	-80.00
234	378.78	80.51	-100.00
235	569.14	-120.98	0.00
236	569.15	-120.98	-20.00
237	569.15	-120.98	-40.00
238	569.15	-120.98	-60.00
239	569.15	-120.98	-80.00
240	569.14	-120.98	-100.00
241	581.86	0.00	0.00
242	581.87	0.00	-20.00
243	581.87	0.00	-40.00
244	581.86	0.00	-60.00
245	581.86	0.00	-80.00
246	581.86	0.00	-100.00
247	569.14	120.98	0.00
248	569.15	120.98	-20.00
249	569.15	120.98	-40.00
250	569.15	120.98	-60.00
251	569.15	120.98	-80.00
252	569.14	120.98	-100.00
253	854.70	-181.67	0.00
254	854.70	-181.67	-20.00
255	854.70	-181.67	-40.00
256	854.70	-181.67	-60.00
257	854.70	-181.67	-80.00
258	854.70	-181.67	-100.00
259	873.79	0.00	0.00
260	873.80	0.00	-20.00
261	873.80	0.00	-40.00
262	873.79	0.00	-60.00
263	873.79	0.00	-80.00

264	873.79	0.00	-100.00
265	854.70	181.67	0.00
266	854.70	181.67	-20.00
267	854.70	181.67	-40.00
268	854.70	181.67	-60.00
269	854.70	181.67	-80.00
270	854.70	181.67	-100.00
271	1283.02	-272.72	0.00
272	1283.02	-272.72	-20.00
273	1283.02	-272.72	-40.00
274	1283.02	-272.72	-60.00
275	1283.02	-272.72	-80.00
276	1283.02	-272.72	-100.00
277	1311.68	0.00	0.00
278	1311.69	0.00	-20.00
279	1311.69	0.00	-40.00
280	1311.68	0.00	-60.00
281	1311.68	0.00	-80.00
282	1311.68	0.00	-100.00
283	1283.02	272.72	0.00
284	1283.02	272.72	-20.00
285	1283.02	272.72	-40.00
286	1283.02	272.72	-60.00
287	1283.02	272.72	-80.00
288	1283.02	272.72	-100.00
289	1925.50	-409.28	0.00
290	1925.51	-409.28	-20.00
291	1925.51	-409.28	-40.00
292	1925.51	-409.28	-60.00
293	1925.51	-409.28	-80.00
294	1925.50	-409.28	-100.00
295	1968.52	0.00	0.00
296	1968.53	0.00	-20.00
297	1968.53	0.00	-40.00
298	1968.52	0.00	-60.00
299	1968.52	0.00	-80.00
300	1968.52	0.00	-100.00
301	1925.50	409.28	0.00
302	1925.51	409.28	-20.00
303	1925.51	409.28	-40.00
304	1925.51	409.28	-60.00
305	1925.51	409.28	-80.00
306	1925.50	409.28	-100.00
307	2889.23	-614.13	0.00
308	2889.24	-614.13	-20.00
309	2889.24	-614.13	-40.00
310	2889.24	-614.13	-60.00
311	2889.23	-614.13	-80.00
312	2889.23	-614.13	-100.00
313	2953.78	0.00	0.00
314	2953.79	0.00	-20.00
315	2953.79	0.00	-40.00
316	2953.78	0.00	-60.00
317	2953.78	0.00	-80.00
318	2953.78	0.00	-100.00
319	2889.23	614.13	0.00
320	2889.24	614.13	-20.00
321	2889.24	614.13	-40.00
322	2889.24	614.13	-60.00
323	2889.23	614.13	-80.00
324	2889.23	614.13	-100.00
325	4334.84	-921.40	0.00

326	4334.85	-921.40	-20.00
327	4334.84	-921.40	-40.00
328	4334.84	-921.40	-60.00
329	4334.84	-921.40	-80.00
330	4334.84	-921.40	-100.00
331	4431.68	0.00	0.00
332	4431.69	0.00	-20.00
333	4431.69	0.00	-40.00
334	4431.68	0.00	-60.00
335	4431.68	0.00	-80.00
336	4431.68	0.00	-100.00
337	4334.84	921.40	0.00
338	4334.85	921.40	-20.00
339	4334.84	921.40	-40.00
340	4334.84	921.40	-60.00
341	4334.84	921.40	-80.00
342	4334.84	921.40	-100.00
343	6503.22	-1382.30	0.00
344	6503.23	-1382.31	-20.00
345	6503.23	-1382.31	-40.00
346	6503.23	-1382.30	-60.00
347	6503.23	-1382.30	-80.00
348	6503.22	-1382.30	-100.00
349	6648.51	0.00	0.00
350	6648.52	0.00	-20.00
351	6648.52	0.00	-40.00
352	6648.51	0.00	-60.00
353	6648.51	0.00	-80.00
354	6648.51	0.00	-100.00
355	6503.22	1382.30	0.00
356	6503.23	1382.31	-20.00
357	6503.23	1382.31	-40.00
358	6503.23	1382.30	-60.00
359	6503.23	1382.30	-80.00
360	6503.22	1382.30	-100.00
361	9755.82	-2073.66	0.00
362	9755.83	-2073.67	-20.00
363	9755.83	-2073.67	-40.00
364	9755.82	-2073.66	-60.00
365	9755.82	-2073.66	-80.00
366	9755.82	-2073.66	-100.00
367	9973.77	0.00	0.00
368	9973.78	0.00	-20.00
369	9973.78	0.00	-40.00
370	9973.77	0.00	-60.00
371	9973.77	0.00	-80.00
372	9973.77	0.00	-100.00
373	9755.82	2073.66	0.00
374	9755.83	2073.67	-20.00
375	9755.83	2073.67	-40.00
376	9755.82	2073.66	-60.00
377	9755.82	2073.66	-80.00
378	9755.82	2073.66	-100.00
379	14634.71	-3110.70	0.00
380	14634.72	-3110.71	-20.00
381	14634.72	-3110.71	-40.00
382	14634.72	-3110.70	-60.00
383	14634.71	-3110.70	-80.00
384	14634.71	-3110.70	-100.00
385	14961.66	0.00	0.00
386	14961.67	0.00	-20.00
387	14961.67	0.00	-40.00

388	14961.66	0.00	-60.00
389	14961.66	0.00	-80.00
390	14961.66	0.00	-100.00
391	14634.71	3110.70	0.00
392	14634.72	3110.71	-20.00
393	14634.72	3110.71	-40.00
394	14634.72	3110.70	-60.00
395	14634.71	3110.70	-80.00
396	14634.71	3110.70	-100.00
397	21953.04	-4666.26	0.00
398	21953.04	-4666.26	-20.00
399	21953.04	-4666.26	-40.00
400	21953.04	-4666.26	-60.00
401	21953.04	-4666.26	-80.00
402	21953.04	-4666.26	-100.00
403	22443.48	0.00	0.00
404	22443.49	0.00	-20.00
405	22443.49	0.00	-40.00
406	22443.48	0.00	-60.00
407	22443.48	0.00	-80.00
408	22443.48	0.00	-100.00
409	21953.04	4666.26	0.00
410	21953.04	4666.26	-20.00
411	21953.04	4666.26	-40.00
412	21953.04	4666.26	-60.00
413	21953.04	4666.26	-80.00
414	21953.04	4666.26	-100.00

TRIANGULAR-PRISM ELEMENT INCIDENCES

SWITCH FOR ECHO OF PRISMATIC NODES 1
SWITCH FOR ECHO OF TETRAHEDRAL NODES 0

ELEM	NODES					
1	7	1	19	8	2	20
2	7	19	25	8	20	26
3	25	19	37	26	20	38
4	25	37	43	26	38	44
5	43	37	55	44	38	56
6	43	55	61	44	56	62
7	61	55	73	62	56	74
8	61	73	79	62	74	80
9	79	73	91	80	74	92
10	79	91	97	80	92	98
11	97	91	109	98	92	110
12	97	109	115	98	110	116
13	115	109	127	116	110	128
14	115	127	133	116	128	134
15	133	127	145	134	128	146
16	133	145	151	134	146	152
17	151	145	163	152	146	164
18	151	163	169	152	164	170
19	169	163	181	170	164	182
20	169	181	187	170	182	188
21	187	181	199	188	182	200
22	187	199	205	188	200	206
23	205	199	217	206	200	218
24	205	217	223	206	218	224
25	223	217	235	224	218	236
26	223	235	241	224	236	242
27	241	235	253	242	236	254

28	241	253	259	242	254	260
29	259	253	271	260	254	272
30	259	271	277	260	272	278
31	277	271	289	278	272	290
32	277	289	295	278	290	296
33	295	289	307	296	290	308
34	295	307	313	296	308	314
35	313	307	325	314	308	326
36	313	325	331	314	326	332
37	331	325	343	332	326	344
38	331	343	349	332	344	350
39	349	343	361	350	344	362
40	349	361	367	350	362	368
41	367	361	379	368	362	380
42	367	379	385	368	380	386
43	385	379	397	386	380	398
44	385	397	403	386	398	404
45	7	25	31	8	26	32
46	13	7	31	14	8	32
47	25	43	49	26	44	50
48	31	25	49	32	26	50
49	43	61	67	44	62	68
50	49	43	67	50	44	68
51	61	79	85	62	80	86
52	67	61	85	68	62	86
53	79	97	103	80	98	104
54	85	79	103	86	80	104
55	97	115	121	98	116	122
56	103	97	121	104	98	122
57	115	133	139	116	134	140
58	121	115	139	122	116	140
59	133	151	157	134	152	158
60	139	133	157	140	134	158
61	151	169	175	152	170	176
62	157	151	175	158	152	176
63	169	187	193	170	188	194
64	175	169	193	176	170	194
65	187	205	211	188	206	212
66	193	187	211	194	188	212
67	205	223	229	206	224	230
68	211	205	229	212	206	230
69	223	241	247	224	242	248
70	229	223	247	230	224	248
71	241	259	265	242	260	266
72	247	241	265	248	242	266
73	259	277	283	260	278	284
74	265	259	283	266	260	284
75	277	295	301	278	296	302
76	283	277	301	284	278	302
77	295	313	319	296	314	320
78	301	295	319	302	296	320
79	313	331	337	314	332	338
80	319	313	337	320	314	338
81	331	349	355	332	350	356
82	337	331	355	338	332	356
83	349	367	373	350	368	374
84	355	349	373	356	350	374
85	367	385	391	368	386	392
86	373	367	391	374	368	392
87	385	403	409	386	404	410
88	391	385	409	392	386	410
89	8	2	20	9	3	21

90	8	20	26	9	21	27
91	26	20	38	27	21	39
92	26	38	44	27	39	45
93	44	38	56	45	39	57
94	44	56	62	45	57	63
95	62	56	74	63	57	75
96	62	74	80	63	75	81
97	80	74	92	81	75	93
98	80	92	98	81	93	99
99	98	92	110	99	93	111
100	98	110	116	99	111	117
101	116	110	128	117	111	129
102	116	128	134	117	129	135
103	134	128	146	135	129	147
104	134	146	152	135	147	153
105	152	146	164	153	147	165
106	152	164	170	153	165	171
107	170	164	182	171	165	183
108	170	182	188	171	183	189
109	188	182	200	189	183	201
110	188	200	206	189	201	207
111	206	200	218	207	201	219
112	206	218	224	207	219	225
113	224	218	236	225	219	237
114	224	236	242	225	237	243
115	242	236	254	243	237	255
116	242	254	260	243	255	261
117	260	254	272	261	255	273
118	260	272	278	261	273	279
119	278	272	290	279	273	291
120	278	290	296	279	291	297
121	296	290	308	297	291	309
122	296	308	314	297	309	315
123	314	308	326	315	309	327
124	314	326	332	315	327	333
125	332	326	344	333	327	345
126	332	344	350	333	345	351
127	350	344	362	351	345	363
128	350	362	368	351	363	369
129	368	362	380	369	363	381
130	368	380	386	369	381	387
131	386	380	398	387	381	399
132	386	398	404	387	399	405
133	8	26	32	9	27	33
134	14	8	32	15	9	33
135	26	44	50	27	45	51
136	32	26	50	33	27	51
137	44	62	68	45	63	69
138	50	44	68	51	45	69
139	62	80	86	63	81	87
140	68	62	86	69	63	87
141	80	98	104	81	99	105
142	86	80	104	87	81	105
143	98	116	122	99	117	123
144	104	98	122	105	99	123
145	116	134	140	117	135	141
146	122	116	140	123	117	141
147	134	152	158	135	153	159
148	140	134	158	141	135	159
149	152	170	176	153	171	177
150	158	152	176	159	153	177
151	170	188	194	171	189	195

152	176	170	194	177	171	195
153	188	206	212	189	207	213
154	194	188	212	195	189	213
155	206	224	230	207	225	231
156	212	206	230	213	207	231
157	224	242	248	225	243	249
158	230	224	248	231	225	249
159	242	260	266	243	261	267
160	248	242	266	249	243	267
161	260	278	284	261	279	285
162	266	260	284	267	261	285
163	278	296	302	279	297	303
164	284	278	302	285	279	303
165	296	314	320	297	315	321
166	302	296	320	303	297	321
167	314	332	338	315	333	339
168	320	314	338	321	315	339
169	332	350	356	333	351	357
170	338	332	356	339	333	357
171	350	368	374	351	369	375
172	356	350	374	357	351	375
173	368	386	392	369	387	393
174	374	368	392	375	369	393
175	386	404	410	387	405	411
176	392	386	410	393	387	411
177	9	3	21	10	4	22
178	9	21	27	10	22	28
179	27	21	39	28	22	40
180	27	39	45	28	40	46
181	45	39	57	46	40	58
182	45	57	63	46	58	64
183	63	57	75	64	58	76
184	63	75	81	64	76	82
185	81	75	93	82	76	94
186	81	93	99	82	94	100
187	99	93	111	100	94	112
188	99	111	117	100	112	118
189	117	111	129	118	112	130
190	117	129	135	118	130	136
191	135	129	147	136	130	148
192	135	147	153	136	148	154
193	153	147	165	154	148	166
194	153	165	171	154	166	172
195	171	165	183	172	166	184
196	171	183	189	172	184	190
197	189	183	201	190	184	202
198	189	201	207	190	202	208
199	207	201	219	208	202	220
200	207	219	225	208	220	226
201	225	219	237	226	220	238
202	225	237	243	226	238	244
203	243	237	255	244	238	256
204	243	255	261	244	256	262
205	261	255	273	262	256	274
206	261	273	279	262	274	280
207	279	273	291	280	274	292
208	279	291	297	280	292	298
209	297	291	309	298	292	310
210	297	309	315	298	310	316
211	315	309	327	316	310	328
212	315	327	333	316	328	334
213	333	327	345	334	328	346

214	333	345	351	334	346	352
215	351	345	363	352	346	364
216	351	363	369	352	364	370
217	369	363	381	370	364	382
218	369	381	387	370	382	388
219	387	381	399	388	382	400
220	387	399	405	388	400	406
221	9	27	33	10	28	34
222	15	9	33	16	10	34
223	27	45	51	28	46	52
224	33	27	51	34	28	52
225	45	63	69	46	64	70
226	51	45	69	52	46	70
227	63	81	87	64	82	88
228	69	63	87	70	64	88
229	81	99	105	82	100	106
230	87	81	105	88	82	106
231	99	117	123	100	118	124
232	105	99	123	106	100	124
233	117	135	141	118	136	142
234	123	117	141	124	118	142
235	135	153	159	136	154	160
236	141	135	159	142	136	160
237	153	171	177	154	172	178
238	159	153	177	160	154	178
239	171	189	195	172	190	196
240	177	171	195	178	172	196
241	189	207	213	190	208	214
242	195	189	213	196	190	214
243	207	225	231	208	226	232
244	213	207	231	214	208	232
245	225	243	249	226	244	250
246	231	225	249	232	226	250
247	243	261	267	244	262	268
248	249	243	267	250	244	268
249	261	279	285	262	280	286
250	267	261	285	268	262	286
251	279	297	303	280	298	304
252	285	279	303	286	280	304
253	297	315	321	298	316	322
254	303	297	321	304	298	322
255	315	333	339	316	334	340
256	321	315	339	322	316	340
257	333	351	357	334	352	358
258	339	333	357	340	334	358
259	351	369	375	352	370	376
260	357	351	375	358	352	376
261	369	387	393	370	388	394
262	375	369	393	376	370	394
263	387	405	411	388	406	412
264	393	387	411	394	388	412
265	10	4	22	11	5	23
266	10	22	28	11	23	29
267	28	22	40	29	23	41
268	28	40	46	29	41	47
269	46	40	58	47	41	59
270	46	58	64	47	59	65
271	64	58	76	65	59	77
272	64	76	82	65	77	83
273	82	76	94	83	77	95
274	82	94	100	83	95	101
275	100	94	112	101	95	113

276	100	112	118	101	113	119
277	118	112	130	119	113	131
278	118	130	136	119	131	137
279	136	130	148	137	131	149
280	136	148	154	137	149	155
281	154	148	166	155	149	167
282	154	166	172	155	167	173
283	172	166	184	173	167	185
284	172	184	190	173	185	191
285	190	184	202	191	185	203
286	190	202	208	191	203	209
287	208	202	220	209	203	221
288	208	220	226	209	221	227
289	226	220	238	227	221	239
290	226	238	244	227	239	245
291	244	238	256	245	239	257
292	244	256	262	245	257	263
293	262	256	274	263	257	275
294	262	274	280	263	275	281
295	280	274	292	281	275	293
296	280	292	298	281	293	299
297	298	292	310	299	293	311
298	298	310	316	299	311	317
299	316	310	328	317	311	329
300	316	328	334	317	329	335
301	334	328	346	335	329	347
302	334	346	352	335	347	353
303	352	346	364	353	347	365
304	352	364	370	353	365	371
305	370	364	382	371	365	383
306	370	382	388	371	383	389
307	388	382	400	389	383	401
308	388	400	406	389	401	407
309	10	28	34	11	29	35
310	16	10	34	17	11	35
311	28	46	52	29	47	53
312	34	28	52	35	29	53
313	46	64	70	47	65	71
314	52	46	70	53	47	71
315	64	82	88	65	83	89
316	70	64	88	71	65	89
317	82	100	106	83	101	107
318	88	82	106	89	83	107
319	100	118	124	101	119	125
320	106	100	124	107	101	125
321	118	136	142	119	137	143
322	124	118	142	125	119	143
323	136	154	160	137	155	161
324	142	136	160	143	137	161
325	154	172	178	155	173	179
326	160	154	178	161	155	179
327	172	190	196	173	191	197
328	178	172	196	179	173	197
329	190	208	214	191	209	215
330	196	190	214	197	191	215
331	208	226	232	209	227	233
332	214	208	232	215	209	233
333	226	244	250	227	245	251
334	232	226	250	233	227	251
335	244	262	268	245	263	269
336	250	244	268	251	245	269
337	262	280	286	263	281	287

338	268	262	286	269	263	287
339	280	298	304	281	299	305
340	286	280	304	287	281	305
341	298	316	322	299	317	323
342	304	298	322	305	299	323
343	316	334	340	317	335	341
344	322	316	340	323	317	341
345	334	352	358	335	353	359
346	340	334	358	341	335	359
347	352	370	376	353	371	377
348	358	352	376	359	353	377
349	370	388	394	371	389	395
350	376	370	394	377	371	395
351	388	406	412	389	407	413
352	394	388	412	395	389	413
353	11	5	23	12	6	24
354	11	23	29	12	24	30
355	29	23	41	30	24	42
356	29	41	47	30	42	48
357	47	41	59	48	42	60
358	47	59	65	48	60	66
359	65	59	77	66	60	78
360	65	77	83	66	78	84
361	83	77	95	84	78	96
362	83	95	101	84	96	102
363	101	95	113	102	96	114
364	101	113	119	102	114	120
365	119	113	131	120	114	132
366	119	131	137	120	132	138
367	137	131	149	138	132	150
368	137	149	155	138	150	156
369	155	149	167	156	150	168
370	155	167	173	156	168	174
371	173	167	185	174	168	186
372	173	185	191	174	186	192
373	191	185	203	192	186	204
374	191	203	209	192	204	210
375	209	203	221	210	204	222
376	209	221	227	210	222	228
377	227	221	239	228	222	240
378	227	239	245	228	240	246
379	245	239	257	246	240	258
380	245	257	263	246	258	264
381	263	257	275	264	258	276
382	263	275	281	264	276	282
383	281	275	293	282	276	294
384	281	293	299	282	294	300
385	299	293	311	300	294	312
386	299	311	317	300	312	318
387	317	311	329	318	312	330
388	317	329	335	318	330	336
389	335	329	347	336	330	348
390	335	347	353	336	348	354
391	353	347	365	354	348	366
392	353	365	371	354	366	372
393	371	365	383	372	366	384
394	371	383	389	372	384	390
395	389	383	401	390	384	402
396	389	401	407	390	402	408
397	11	29	35	12	30	36
398	17	11	35	18	12	36
399	29	47	53	30	48	54

400	35	29	53	36	30	54
401	47	65	71	48	66	72
402	53	47	71	54	48	72
403	65	83	89	66	84	90
404	71	65	89	72	66	90
405	83	101	107	84	102	108
406	89	83	107	90	84	108
407	101	119	125	102	120	126
408	107	101	125	108	102	126
409	119	137	143	120	138	144
410	125	119	143	126	120	144
411	137	155	161	138	156	162
412	143	137	161	144	138	162
413	155	173	179	156	174	180
414	161	155	179	162	156	180
415	173	191	197	174	192	198
416	179	173	197	180	174	198
417	191	209	215	192	210	216
418	197	191	215	198	192	216
419	209	227	233	210	228	234
420	215	209	233	216	210	234
421	227	245	251	228	246	252
422	233	227	251	234	228	252
423	245	263	269	246	264	270
424	251	245	269	252	246	270
425	263	281	287	264	282	288
426	269	263	287	270	264	288
427	281	299	305	282	300	306
428	287	281	305	288	282	306
429	299	317	323	300	318	324
430	305	299	323	306	300	324
431	317	335	341	318	336	342
432	323	317	341	324	318	342
433	335	353	359	336	354	360
434	341	335	359	342	336	360
435	353	371	377	354	372	378
436	359	353	377	360	354	378
437	371	389	395	372	390	396
438	377	371	395	378	372	396
439	389	407	413	390	408	414
440	395	389	413	396	390	414

HALF-BAND WIDTH 26
NUMBER OF TETRAHEDRAL ELEMENTS 1320

AQUIFER CHARACTERISTICS

FACTOR FOR KXX	1.00E+00
FACTOR FOR KYY	1.00E+00
FACTOR FOR KZZ	1.00E+00
FACTOR FOR SS	1.00E+00
FACTOR FOR SY	1.00E+00
SWITCH FOR ECHO OF AQUIFER PARAMETERS	0
SWITCH FOR ECHO OF BLOCK INDEXES	0

INITIAL WATER LEVELS

SWITCH FOR ECHO OF WATER LEVELS 1

NODE	LEVELS				
1	0.00	0.00	0.00	0.00	0.00
6	0.00	0.00	0.00	0.00	0.00
11	0.00	0.00	0.00	0.00	0.00
16	0.00	0.00	0.00	0.00	0.00
21	0.00	0.00	0.00	0.00	0.00
26	0.00	0.00	0.00	0.00	0.00
31	0.00	0.00	0.00	0.00	0.00
36	0.00	0.00	0.00	0.00	0.00
41	0.00	0.00	0.00	0.00	0.00
46	0.00	0.00	0.00	0.00	0.00
51	0.00	0.00	0.00	0.00	0.00
56	0.00	0.00	0.00	0.00	0.00
61	0.00	0.00	0.00	0.00	0.00
66	0.00	0.00	0.00	0.00	0.00
71	0.00	0.00	0.00	0.00	0.00
76	0.00	0.00	0.00	0.00	0.00
81	0.00	0.00	0.00	0.00	0.00
86	0.00	0.00	0.00	0.00	0.00
91	0.00	0.00	0.00	0.00	0.00
96	0.00	0.00	0.00	0.00	0.00
101	0.00	0.00	0.00	0.00	0.00
106	0.00	0.00	0.00	0.00	0.00
111	0.00	0.00	0.00	0.00	0.00
116	0.00	0.00	0.00	0.00	0.00
121	0.00	0.00	0.00	0.00	0.00
126	0.00	0.00	0.00	0.00	0.00
131	0.00	0.00	0.00	0.00	0.00
136	0.00	0.00	0.00	0.00	0.00
141	0.00	0.00	0.00	0.00	0.00
146	0.00	0.00	0.00	0.00	0.00
151	0.00	0.00	0.00	0.00	0.00
156	0.00	0.00	0.00	0.00	0.00
161	0.00	0.00	0.00	0.00	0.00
166	0.00	0.00	0.00	0.00	0.00
171	0.00	0.00	0.00	0.00	0.00
176	0.00	0.00	0.00	0.00	0.00
181	0.00	0.00	0.00	0.00	0.00
186	0.00	0.00	0.00	0.00	0.00
191	0.00	0.00	0.00	0.00	0.00
196	0.00	0.00	0.00	0.00	0.00
201	0.00	0.00	0.00	0.00	0.00
206	0.00	0.00	0.00	0.00	0.00
211	0.00	0.00	0.00	0.00	0.00
216	0.00	0.00	0.00	0.00	0.00
221	0.00	0.00	0.00	0.00	0.00
226	0.00	0.00	0.00	0.00	0.00
231	0.00	0.00	0.00	0.00	0.00
236	0.00	0.00	0.00	0.00	0.00
241	0.00	0.00	0.00	0.00	0.00
246	0.00	0.00	0.00	0.00	0.00
251	0.00	0.00	0.00	0.00	0.00
256	0.00	0.00	0.00	0.00	0.00
261	0.00	0.00	0.00	0.00	0.00
266	0.00	0.00	0.00	0.00	0.00
271	0.00	0.00	0.00	0.00	0.00
276	0.00	0.00	0.00	0.00	0.00

281	0.00	0.00	0.00	0.00	0.00
286	0.00	0.00	0.00	0.00	0.00
291	0.00	0.00	0.00	0.00	0.00
296	0.00	0.00	0.00	0.00	0.00
301	0.00	0.00	0.00	0.00	0.00
306	0.00	0.00	0.00	0.00	0.00
311	0.00	0.00	0.00	0.00	0.00
316	0.00	0.00	0.00	0.00	0.00
321	0.00	0.00	0.00	0.00	0.00
326	0.00	0.00	0.00	0.00	0.00
331	0.00	0.00	0.00	0.00	0.00
336	0.00	0.00	0.00	0.00	0.00
341	0.00	0.00	0.00	0.00	0.00
346	0.00	0.00	0.00	0.00	0.00
351	0.00	0.00	0.00	0.00	0.00
356	0.00	0.00	0.00	0.00	0.00
361	0.00	0.00	0.00	0.00	0.00
366	0.00	0.00	0.00	0.00	0.00
371	0.00	0.00	0.00	0.00	0.00
376	0.00	0.00	0.00	0.00	0.00
381	0.00	0.00	0.00	0.00	0.00
386	0.00	0.00	0.00	0.00	0.00
391	0.00	0.00	0.00	0.00	0.00
396	0.00	0.00	0.00	0.00	0.00
401	0.00	0.00	0.00	0.00	0.00
406	0.00	0.00	0.00	0.00	0.00
411	0.00	0.00	0.00	0.00	

CONSTANT-HEAD NODES

NUMBER OF NODES	0
SWITCH FOR ECHO OF INPUT HEADS	0
SWITCH FOR ECHO OF STEP FLUXES	0

SOURCE AND SINK FLUXES

NUMBER OF FLUX DATA SETS	1
NUMBER OF TIME STEPS	2
FACTOR FOR NODAL FLUXES	1.00E+00
SWITCH FOR ECHO OF INPUT FLUXES	1
SWITCH FOR ECHO OF STEP FLUXES	1

FLUX SET 1

NODE	QFLUX
7	-2.97E-01

TIME-STEP INDICATORS AND MULTIPLIERS

STEP	FACTORS
1	1.00E+00
2	1.00E+00

```
EVAPOTRANSPIRATION
------------------
NUMBER OF NODES                                    0
NUMBER OF TIME STEPS                               0
SWITCH FOR ECHO OF INPUT DATA                      0
SWITCH FOR ECHO OF FLUXES                          0

VARIABLE FLUX BOUNDARY NODES
----------------------------
NUMBER OF VARIABLE FLUX NODES                      0
SWITCH FOR ECHO OF PARAMETERS                      0
SWITCH FOR ECHO OF FLUXES                          0

FAULT NODES
-----------
NUMBER OF FAULT-NODE PAIRS                        17
SWITCH FOR ECHO OF INPUT                           0
SWITCH FOR ECHO OF OUTPUT                          0

LAND-SUBSIDENCE PARAMETERS
--------------------------

SWITCH FOR ECHO OF PARAMETERS                      0
SWITCH FOR ECHO OF INITIAL CONSOLIDATION           0
SWITCH FOR OUTPUT OF CONSOLIDATION                 0
SWITCH FOR ECHO OF COLUMNS                         0
SWITCH FOR OUTPUT OF SUBSIDENCE                    0

WATER PARAMETERS
----------------
SWITCH FOR WATER                                   0

TIME STEP        1
------------------
ELAPSED TIME                                1.00E+00
NUMBER OF ITERATIONS                               1
WATER-LEVEL CHANGE ON LAST ITERATION         349.574

COMPUTED WATER LEVELS
---------------------
      NODE    LEVELS

        1   -349.57   -349.56   -349.55   -349.55   -349.55
        6   -349.55   -349.57   -349.56   -349.55   -349.54
       11   -349.54   -349.54   -349.57   -349.56   -349.55
       16   -349.55   -349.55   -349.55   -300.24   -297.69
       21   -297.37   -297.10   -297.06   -293.86   -299.25
       26   -297.54   -297.25   -297.21   -297.18   -295.26
       31   -298.06   -297.65   -297.37   -297.10   -297.10
       36   -296.45   -262.76   -259.83   -259.45   -259.22
       41   -259.07   -255.68   -261.42   -259.73   -259.38
       46   -259.27   -259.16   -257.34   -260.02   -259.78
       51   -259.45   -259.22   -259.15   -258.87   -230.51
       56   -227.73   -227.33   -227.12   -226.99   -223.97
       61   -228.95   -227.68   -227.31   -227.18   -227.02
       66   -225.72   -227.45   -227.66   -227.33   -227.13
       71   -227.10   -227.49   -201.12   -198.74   -198.36
       76   -198.15   -197.93   -195.37   -199.46   -198.68
       81   -198.35   -198.19   -197.98   -197.24   -197.87
       86   -198.63   -198.36   -198.16   -198.08   -199.09
       91   -173.42   -171.47   -171.16   -170.98   -170.68
       96   -168.51   -171.76   -171.38   -171.14   -170.99
```

101	-170.78	-170.42	-170.11	-171.32	-171.16
106	-170.99	-170.88	-172.29	-146.96	-145.32
111	-145.07	-144.92	-144.65	-142.80	-145.33
116	-145.19	-145.04	-144.93	-144.78	-144.65
121	-143.73	-145.09	-145.06	-144.95	-144.92
126	-146.47	-121.49	-120.12	-119.89	-119.78
131	-119.52	-118.00	-119.97	-119.93	-119.86
136	-119.80	-119.71	-119.67	-118.51	-119.77
141	-119.86	-119.83	-119.91	-121.35	-96.95
146	-95.91	-95.68	-95.57	-95.32	-94.17
151	-95.65	-95.64	-95.63	-95.60	-95.57
156	-95.56	-94.43	-95.42	-95.61	-95.66
161	-95.84	-96.96	-73.63	-72.94	-72.71
166	-72.58	-72.33	-71.59	-72.62	-72.62
171	-72.62	-72.62	-72.61	-72.61	-71.72
176	-72.38	-72.60	-72.71	-72.93	-73.66
181	-52.12	-51.73	-51.54	-51.39	-51.19
186	-50.78	-51.42	-51.42	-51.42	-51.43
191	-51.43	-51.43	-50.86	-51.22	-51.40
196	-51.54	-51.73	-52.13	-33.34	-33.15
201	-33.02	-32.91	-32.78	-32.59	-32.91
206	-32.91	-32.91	-32.92	-32.92	-32.92
211	-32.64	-32.80	-32.92	-33.02	-33.14
216	-33.32	-18.38	-18.30	-18.24	-18.18
221	-18.11	-18.03	-18.15	-18.15	-18.16
226	-18.16	-18.16	-18.16	-18.07	-18.13
231	-18.18	-18.23	-18.29	-18.35	-8.10
236	-8.07	-8.05	-8.03	-8.00	-7.98
241	-8.01	-8.01	-8.01	-8.01	-8.01
246	-8.01	-8.00	-8.02	-8.03	-8.05
251	-8.06	-8.08	-2.51	-2.51	-2.50
256	-2.49	-2.49	-2.48	-2.50	-2.50
261	-2.50	-2.50	-2.50	-2.50	-2.49
266	-2.49	-2.49	-2.50	-2.50	-2.50
271	-0.42	-0.42	-0.42	-0.42	-0.42
276	-0.42	-0.44	-0.44	-0.44	-0.44
281	-0.44	-0.44	-0.42	-0.42	-0.42
286	-0.42	-0.42	-0.42	-0.01	-0.01
291	-0.01	-0.01	-0.01	-0.01	-0.02
296	-0.02	-0.02	-0.02	-0.02	-0.02
301	-0.01	-0.01	-0.01	-0.01	-0.01
306	-0.01	0.00	0.00	0.00	0.00
311	0.00	0.00	0.00	0.00	0.00
316	0.00	0.00	0.00	0.00	0.00
321	0.00	0.00	0.00	0.00	0.00
326	0.00	0.00	0.00	0.00	0.00
331	0.00	0.00	0.00	0.00	0.00
336	0.00	0.00	0.00	0.00	0.00
341	0.00	0.00	0.00	0.00	0.00
346	0.00	0.00	0.00	0.00	0.00
351	0.00	0.00	0.00	0.00	0.00
356	0.00	0.00	0.00	0.00	0.00
361	0.00	0.00	0.00	0.00	0.00
366	0.00	0.00	0.00	0.00	0.00
371	0.00	0.00	0.00	0.00	0.00
376	0.00	0.00	0.00	0.00	0.00
381	0.00	0.00	0.00	0.00	0.00
386	0.00	0.00	0.00	0.00	0.00
391	0.00	0.00	0.00	0.00	0.00
396	0.00	0.00	0.00	0.00	0.00
401	0.00	0.00	0.00	0.00	0.00
406	0.00	0.00	0.00	0.00	0.00

```
       411      0.00       0.00       0.00       0.00

SOURCE AND SINK FLUXES
----------------------
TOTAL FLUXES         -2.97E-01
      NODE     QFLUX

         7 -2.97E-01

GROUND-WATER BUDGET
-------------------
SOURCE-SINK NODES                                -2.97E-01
PUMPING                                           0.00E+00
RECHARGE                                          0.00E+00
CONSTANT-HEAD NODES                               0.00E+00
EVAPOTRANSPIRATION                                0.00E+00
STREAM-AQUIFER INTERACTION                        0.00E+00
VARIABLE-FLUX BOUNDARIES                          0.00E+00
FAULT-NODE PAIRS                                  0.00E+00
ELASTIC STORAGE CHANGE IN FINE BEDS               0.00E+00
INELASTIC STORAGE CHANGE IN FINE BEDS             0.00E+00
ELASTIC STORAGE CHANGE IN COARSE BEDS            -2.97E-01
RESIDUAL                                         -1.03E-11

TIME STEP         2
-------------------
ELAPSED TIME                                      2.00E+00
NUMBER OF ITERATIONS                                     1
WATER-LEVEL CHANGE ON LAST ITERATION                30.155

COMPUTED WATER LEVELS
---------------------
      NODE     LEVELS

         1  -379.72    -379.71    -379.71    -379.70    -379.70
         6  -379.70    -379.72    -379.71    -379.70    -379.69
        11  -379.69    -379.69    -379.72    -379.71    -379.71
        16  -379.70    -379.70    -379.70    -330.40    -327.84
        21  -327.52    -327.25    -327.20    -324.00    -329.40
        26  -327.69    -327.39    -327.36    -327.33    -325.40
        31  -328.22    -327.80    -327.52    -327.25    -327.25
        36  -326.59    -292.91    -289.97    -289.59    -289.35
        41  -289.21    -285.80    -291.57    -289.88    -289.51
        46  -289.41    -289.29    -287.46    -290.17    -289.92
        51  -289.59    -289.36    -289.29    -289.00    -260.64
        56  -257.84    -257.44    -257.23    -257.10    -254.06
        61  -259.08    -257.80    -257.42    -257.29    -257.12
        66  -255.82    -257.58    -257.77    -257.44    -257.24
        71  -257.21    -257.60    -231.19    -228.80    -228.41
        76  -228.20    -227.97    -225.41    -229.53    -228.74
        81  -228.41    -228.25    -228.03    -227.28    -227.93
        86  -228.69    -228.41    -228.21    -228.13    -229.14
        91  -203.38    -201.41    -201.10    -200.91    -200.61
        96  -198.42    -201.71    -201.32    -201.08    -200.92
       101  -200.71    -200.35    -200.05    -201.26    -201.09
       106  -200.92    -200.81    -202.22    -176.69    -175.02
       111  -174.77    -174.62    -174.34    -172.47    -175.05
       116  -174.90    -174.75    -174.63    -174.48    -174.34
       121  -173.42    -174.79    -174.76    -174.64    -174.62
       126  -176.18    -150.77    -149.37    -149.13    -149.01
       131  -148.75    -147.20    -149.22    -149.18    -149.11
       136  -149.04    -148.95    -148.91    -147.71    -149.01
```

141	-149.10	-149.06	-149.15	-150.62	-125.39
146	-124.31	-124.07	-123.95	-123.69	-122.49
151	-124.05	-124.04	-124.02	-124.00	-123.96
156	-123.95	-122.75	-123.80	-124.00	-124.05
161	-124.24	-125.41	-100.58	-99.84	-99.60
166	-99.45	-99.19	-98.39	-99.52	-99.52
171	-99.52	-99.52	-99.51	-99.50	-98.51
176	-99.23	-99.47	-99.60	-99.84	-100.64
181	-76.58	-76.14	-75.93	-75.76	-75.53
186	-75.07	-75.81	-75.82	-75.82	-75.82
191	-75.82	-75.83	-75.13	-75.55	-75.77
196	-75.93	-76.16	-76.62	-53.99	-53.76
201	-53.60	-53.47	-53.31	-53.07	-53.48
206	-53.48	-53.49	-53.49	-53.49	-53.50
211	-53.11	-53.32	-53.48	-53.61	-53.76
216	-53.99	-33.85	-33.75	-33.66	-33.58
221	-33.49	-33.39	-33.55	-33.55	-33.55
226	-33.56	-33.56	-33.56	-33.41	-33.51
231	-33.59	-33.66	-33.74	-33.84	-17.68
236	-17.64	-17.60	-17.56	-17.53	-17.49
241	-17.51	-17.51	-17.51	-17.52	-17.52
246	-17.52	-17.51	-17.54	-17.57	-17.60
251	-17.63	-17.66	-6.87	-6.85	-6.84
256	-6.83	-6.82	-6.81	-6.80	-6.80
261	-6.80	-6.80	-6.80	-6.80	-6.82
266	-6.83	-6.83	-6.84	-6.85	-6.86
271	-1.60	-1.60	-1.59	-1.59	-1.59
276	-1.58	-1.61	-1.61	-1.61	-1.61
281	-1.61	-1.61	-1.59	-1.59	-1.59
286	-1.59	-1.59	-1.59	-0.12	-0.12
291	-0.12	-0.12	-0.12	-0.12	-0.15
296	-0.15	-0.15	-0.15	-0.15	-0.15
301	-0.12	-0.12	-0.12	-0.12	-0.12
306	-0.12	0.01	0.01	0.01	0.01
311	0.01	0.01	0.00	0.00	0.00
316	0.00	0.00	0.00	0.01	0.01
321	0.01	0.01	0.01	0.01	0.00
326	0.00	0.00	0.00	0.00	0.00
331	0.00	0.00	0.00	0.00	0.00
336	0.00	0.00	0.00	0.00	0.00
341	0.00	0.00	0.00	0.00	0.00
346	0.00	0.00	0.00	0.00	0.00
351	0.00	0.00	0.00	0.00	0.00
356	0.00	0.00	0.00	0.00	0.00
361	0.00	0.00	0.00	0.00	0.00
366	0.00	0.00	0.00	0.00	0.00
371	0.00	0.00	0.00	0.00	0.00
376	0.00	0.00	0.00	0.00	0.00
381	0.00	0.00	0.00	0.00	0.00
386	0.00	0.00	0.00	0.00	0.00
391	0.00	0.00	0.00	0.00	0.00
396	0.00	0.00	0.00	0.00	0.00
401	0.00	0.00	0.00	0.00	0.00
406	0.00	0.00	0.00	0.00	0.00
411	0.00	0.00	0.00	0.00	

SOURCE AND SINK FLUXES

TOTAL FLUXES -2.97E-01

NODE	QFLUX
7	-2.97E-01

GROUND-WATER BUDGET

SOURCE-SINK NODES	-2.97E-01
PUMPING	0.00E+00
RECHARGE	0.00E+00
CONSTANT-HEAD NODES	0.00E+00
EVAPOTRANSPIRATION	0.00E+00
STREAM-AQUIFER INTERACTION	0.00E+00
VARIABLE-FLUX BOUNDARIES	0.00E+00
FAULT-NODE PAIRS	0.00E+00
ELASTIC STORAGE CHANGE IN FINE BEDS	0.00E+00
INELASTIC STORAGE CHANGE IN FINE BEDS	0.00E+00
ELASTIC STORAGE CHANGE IN COARSE BEDS	-2.97E-01
RESIDUAL	-1.12E-11