MODIFICATIONS OF A THREE-DIMENSIONAL GROUND-WATER FLOW MODEL TO ACCOUNT FOR VARIABLE WATER DENSITY AND EFFECTS OF MULTIAQUIFER WELLS

By A. L. Kontis and R. J. Mandle

U.S. GEOLOGICAL SURVEY

Water-Resources Investigations Report 87-4265

Madison, Wisconsin

1988

DEPARTMENT OF THE INTERIOR

WILLIAM P. CLARK, Secretary

U.S. GEOLOGICAL SURVEY

Dallas L. Peck, Director

For additional information
write to:

Office of Ground Water
U.S. Geological Survey, WRD
12201 Sunrise Valley Drive
Reston, Va. 22092

Copies of this report can be
purchased from:

U.S. Geological Survey
Books and Open-File Reports
Federal Center, Bldg. 41
Box 25425
Denver, CO. 80225

CONTENTS

ILLUSTRATIONS

ILLUSTRATIONS (continued)

TABLES

CONVERSION FACTORS AND ABBREVIATIONS

For readers who prefer metric (International System) units rather than the inch-pound units used in this report, conversion factors are listed below:

Multiply inch-pound unit	By	To obtain metric unit
Length		
inch (in.)	25.40	millimeter (mm)
foot (ft)	0.3048	meter (m)
mile (mi)	1.609	kilometer (km)
Area		
square foot (ft^2)	0.09290	square meter (m^2)
square mile (mi^2)	2.590	square kilometer (km^2)
Volume		
gallon (gal)	0.003785	cubic meter (m^3)
million gallons (Mgal)	3,785	cubic meter (m^3)
cubic foot (ft^3)	0.02832	cubic meter (m^3)
Flow		
foot per second (ft/s)	0.3048	meter per second (m/s)
foot per day (ft/d)	0.3048	meter per day (m/d)
cubic foot per second (ft^3/s)	0.02832	cubic meter per second (m^3/s)
cubic foot per second per square mile [$(ft^3/s)mi^2$]	0.01093	cubic meter per second per square kilometer [$(m^3/s)/km^2$]
million gallons per day (Mgal/d)	0.04381	cubic meter per second (m^3/s)
Temperature		
degree Fahrenheit (°F)	°C = 5/9 (°F-32)	degrees Celsius (°C)
Hydraulic Conductivity		
foot per second (ft/s)	0.3048	meter per second (m/s)
Transmissivity		
square foot per day (ft^2/d)	0.09290	square meter per day (m^2/d)
Pressure		
bar	100	kilopascal (kPa)
pound per square inch (lb/in^2)	6.895	kilopascal (kPa)
pound per square inch (lb/in^2)	703.1	kilogram-force per square meter (kg/m^2)

CONVERSION FACTORS AND ABBREVIATIONS (continued)

Dynamic Viscosity		
pound-force-second per square foot (lbf.s/ft^2)	47.88	pascal-second (Pa.s)
slug per foot per second (slug/ft/s)	47.88	pascal-second (Pa.s)
Acceleration		
foot per second squared (ft/s^2)	0.3048	meter per second squared (m/s^2)
Density		
slug per cubic foot (slug/ft^3)	515.4	kilogram per cubic meter (kg/m^3)

EQUIVALENTS

	Inch-pound unit	Metric unit
Acceleration of gravity	32.17 ft/s^2	9.806 m/s^2
Density of pure water (68°F, 20°C)	1.937 slug/ft^3	998.2 kg/m^3
Specific weight of water (58°F, 15°C)	62.4 lb/ft^3	9,800 n/m^3
Standard atmosphere	14.70 lb/in^2	101,325 kPa

Sea Level: In this report "sea level" refers to the National Geodetic Vertical Datum of 1929 (NGVD of 1929)--a geodetic datum derived from a general adjustment of the first-order level nets of both the United States and Canada, formerly called "Sea level datum of 1929."

MODIFICATIONS OF A THREE-DIMENSIONAL GROUND-WATER FLOW MODEL TO ACCOUNT FOR VARIABLE WATER DENSITY AND EFFECTS OF MULTIAQUIFER WELLS

By A. L. Kontis and R. J. Mandle

ABSTRACT

A three-dimensional finite-difference ground-water-flow model was modified and used in the U.S. Geological Survey's Northern Midwest Regional Aquifer System Analysis to simulate the effects of wells open to various combinations of aquifers and the effects of ground-water density that varies spatially but is time invariant. The computer code for the slice-successive-over-relaxation solution method used to solve for head and a description of how it functions is given, as well as modifications that were made to account for the hydraulic effects of multiaquifer-wells. In addition, two programs are presented that compute data required for the simulation of variable-density effects: one program calculates model-block conductances and variable-density terms external to the ground-water-flow model; the other computes ground-water and pure-water density and viscosity for input to the first program.

INTRODUCTION

As part of the U.S. Geological Survey's Regional Aquifer-System Analysis (RASA) program (Bennett, 1979; Steinhilber and Young, 1979; Young and others, 1986), a regional ground-water flow simulation (Mandle and Kontis, in press) of the Cambrian-Ordovician aquifer system of the northern Midwest was developed through a modified version of the three-dimensional finite difference ground-water-flow model (Trescott, 1975; Trescott and Larson, 1976). This model is referred to herein as the Trescott-Larson model. A 378,880-mi^2 area was modeled that includes large areas east and south of the study area (fig. 1) to satisfy boundary conditions.

The effects of two conditions in the model area--the presence of brines in the Illinois and Michigan basins, and the presence of wells open to more than one aquifer--could not be simulated with the documented version of the Trescott-Larson model because it assumes that ground water is of constant density and that a given well is open only to a single aquifer.

To simulate the effect of brines of variable density on the regional flow system, an approach suggested by Bennett (1980) was used, which assumes that, although temperature and solute concentration may vary spatially, they are invariant over the simulated period (120 years). Under this assumption, the finite-difference ground-water-flow equation can be written to accommodate the effects of variations in water density such that the Trescott-Larson model program can be used with only minor modifications to the computer code. This is done by precomputing model hydraulic conductances and water-density terms through a computer program termed VARDEN (VARiable DENsity), which is separate from the Trescott-Larson model.

The majority of wells in the study area are open to more than one aquifer except in Minnesota (fig. 1). As discussed by Bennett and others (1982), multiaquifer wells can markedly affect the nature of flow between aquifers connected by the wells. The flow equation that is solved in the Trescott-Larson model links the head at a given node with only the heads in the four adjacent nodes within a model layer and with the two adjacent nodes in the layers above and below the node. For a well open to more than one aquifer, internal flow between the aquifers through the open well is not accounted for. In addition, most measured water levels in multiaquifer wells that are available for model calibration are a composite of generally unequal heads. Consequently, to simulate the effects of multiaquifer wells and to provide a means of calibrating model results, additional modifications were made to the flow equation and to the Trescott-Larson model code. To implement the modified finite-difference flow equation, the slice-successive-over-relaxation (SSOR) solution scheme was used.

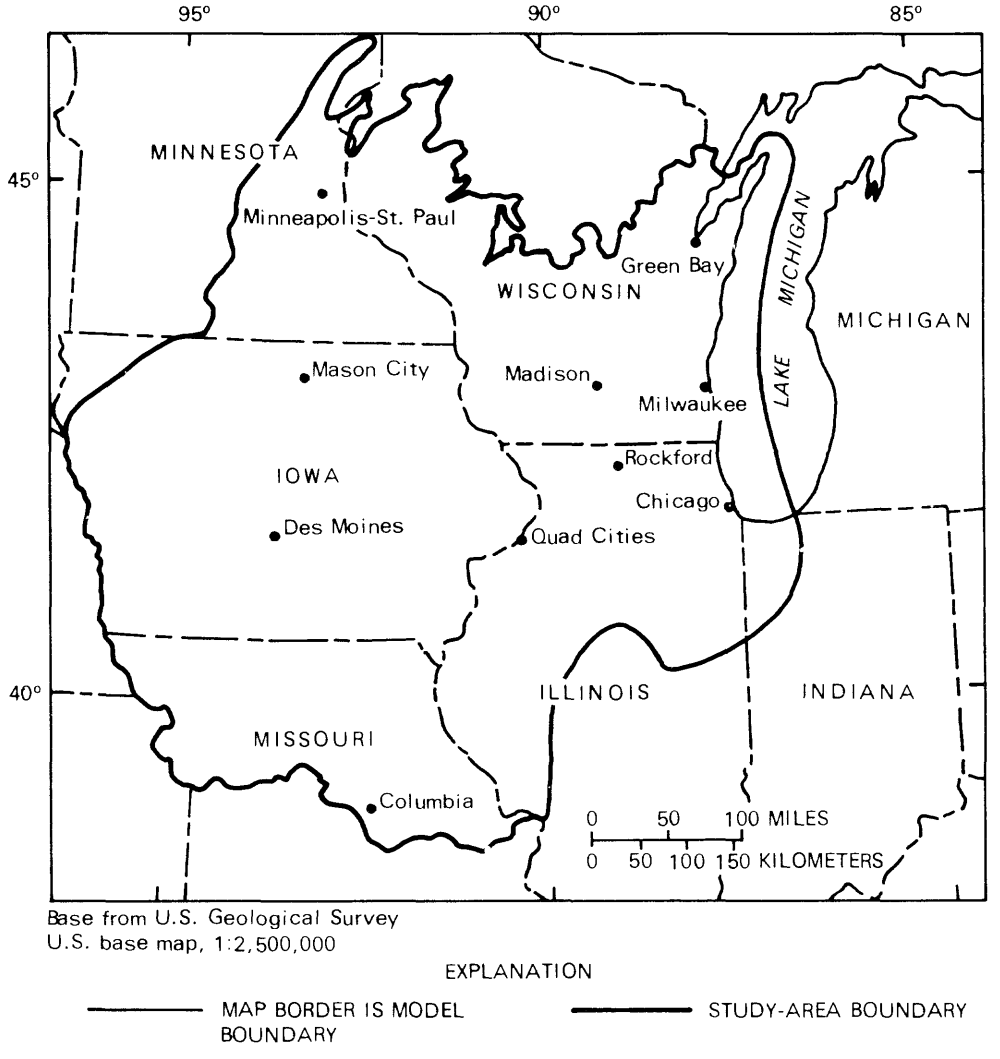


Figure 1.--Northern Midwest Regional Aquifer Systems Analysis study area and boundary of ground-water flow model.

Purpose and Scope

The development of the modified flow equation and results of the northern Midwest ground-water flow simulation are discussed in detail in Mandle and Kontis (in press). This report (1) presents the computer modifications of the Trescott-Larson model and SSOR algorithm with which variable-density and multiaquifer-well effects were simulated, and (2) describes (a) the VARDEN program, which precomputes model hydraulic conductances and variable-density terms, and (b) the STEAM program, which computes ground-water density, pure-water density, and viscosity for input to VARDEN. The original SSOR-solution computer code for use with the Trescott-Larson model was written by S. P. Larson and C. R. Faust (U.S. Geological Survey, written commun., 1979). Because the original code was not formally documented, it is given here in its entirety (appendix F) with a general explanation of how the subroutine functions (appendix A). In addition, the variable-density method is compared with results from a cross-sectional electric analog model of an area in Puerto Rico (Bennett and Guisti, 1971).

The purpose of this report is to present the modifications and support programs in sufficient detail that the computational aspects of the variable-density and multiaquifer-well effects used in the northern Midwest regional simulation (Mandle and Kontis, in press) are clearly understood. The modifications and programs were developed specifically for the northern Midwest study, but the variable-density and multiaquifer-well formulations could be used for other applications as well. Although the report is not intended to be a users manual, the text and comment statements in program listings give sufficient detail that the procedures can be readily adapted to systems with similar hydrogeologic conditions. The reader is assumed to be familar with the Trescott-Larson model or to have access to the original documentation (Trescott, 1975; Trescott and Larson, 1976). Where possible, this report uses the same notation as was used in the original documentation.

MODIFICATIONS

Variable Water Density

Finite-Difference Equation

As shown in Mandle and Kontis (in press), the quasi-three-dimensional finite-difference equation describing ground water flow in terms of freshwater head, h_f, for spatially varying ground-water density, may be expressed as:

$$
\begin{aligned}
&-TR_{i,j-1,k}(h_{f_{i,j,k}}-h_{f_{i,j-1,k}}) + A_{i,j-1,k} + TR_{i,j,k}(h_{f_{i,j+1,k}}-h_{f_{i,j,k}}) - A_{i,j,k} \\
&- TC_{i-1,j,k}(h_{f_{i,j,k}}-h_{f_{i-1,j,k}}) + B_{i-1,j,k} + TC_{i,j,k}(h_{f_{i+1,j,k}}-h_{f_{i,j,k}}) - B_{i,j,k} \\
&- TK_{i,j,k-1}(h_{f_{i,j,k}}-h_{f_{i,j,k-1}}) + \Gamma_{i,j,k-1} + TK_{i,j,k}(h_{f_{i,j,k+1}}-h_{f_{i,j,k}}) - \Gamma_{i,j,k} \\
&= S^* \frac{\Delta h_{f_{i,j,k}}}{\Delta t} - Q^*
\end{aligned}
\qquad (1)
$$

where:

$TR_{i,j-1,k}$ and $TR_{i,j,k}$ are the aquifer-layer conductances along rows between pairs of nodes (i,j-1,k),(i,j,k) and (i,j,k),(i,j+1,k) (fig. 2C);

$TC_{i-1,j,k}$ and $TC_{i,j,k}$ are the aquifer-layer conductances along columns between pairs of nodes (i-1,j,k),(i,j,k) and (i,j,k),(i+1,j,k) (fig. 2B); and

$TK_{i,j,k}$ and $TK_{i,j,k-1}$ are conductances of the confining layer between pairs of aquifer layer nodes (i,j,k+1),(i,j,k) and (i,j,k),(i,j,k-1) (fig. 2B and 2C).

The nodes are taken to be at the center of model blocks defined to be the midpoint between the top and bottom surfaces of aquifer layers. In equation 1, unscripted terms are understood to be at node i,j,k.

Equation 1 is developed in Mandle and Kontis (in press) by applying the variable-density form of Darcy's law and balancing the mass of equivalent pure water entering and leaving an aquifer element, on the assumption that the dispersion of pure water is negligible. An alternative development of variable density effects, based on the conservation of pure-water mass, is given in Weiss (1982a). A method based on conservation of ground-water mass is described by Kuiper (1983, 1985).

In figure 2, the nodes are depicted as dipping along an α axis inclined a' degrees with respect to the x (horizontal) axis of an x,y,z cartesian coordinate system (fig. 2C) and along a β axis inclined b' degrees with respect to the y (horizontal) axis (fig. 2B). The normal to the plane of α and β defines a γ axis, which is at an angle c' with respect to the z (vertical) axis (fig. 2A).

In equation 1, the A, B, and Γ terms represent the effects of gravity forces in the α, β, and γ directions arising from the presence of variable-density ground water. They are computed from:

$$A_{i,j-1,k} = -TR_{i,j-1,k}(\rho'-1)_{i,j-1/2,k}(Z_{i,j,k}-Z_{i,j-1,k}) \quad (2a)$$

$$A_{i,j,k} = -TR_{i,j,k}(\rho'-1)_{i,j+1/2,k}(Z_{i,j+1,k}-Z_{i,j,k}) \quad (2b)$$

$$B_{i-1,j,k} = -TC_{i-1,j,k}(\rho'-1)_{i-1/2,j,k}(Z_{i,j,k}-Z_{i-1,j,k}) \quad (2c)$$

$$B_{i,j,k} = -TC_{i,j,k}(\rho'-1)_{i+1/2,j,k}(Z_{i+1,j,k}-Z_{i,j,k}) \quad (2d)$$

$$\Gamma_{i,j,k-1} = -TK_{i,j,k-1}b_L(\rho'-1)_{i,j,k-1/2}\left(\frac{\Delta x\Delta y}{\Delta\alpha\Delta\beta}\right)_{i,j,k} \quad (2e)$$

$$\Gamma_{i,j,k} = -TK_{i,j,k}b_L(\rho'-1)_{i,j,k+1/2}\left(\frac{\Delta x\Delta y}{\Delta\alpha\Delta\beta}\right)_{i,j,k} \quad (2f)$$

where ρ' is the relative density defined to be the ratio of the density of ground water of interest, ρ_s, to the density of the equivalent freshwater, ρ_f, at a temperature of 20 °C. The ground-water density (ρ_s) is equal to the combined densities of solvent (pure water) and solutes in the solvent. Relative density at the finite-difference block faces is computed by averaging the densities of adjacent nodes. For example at the i,j + 1/2,k block face,

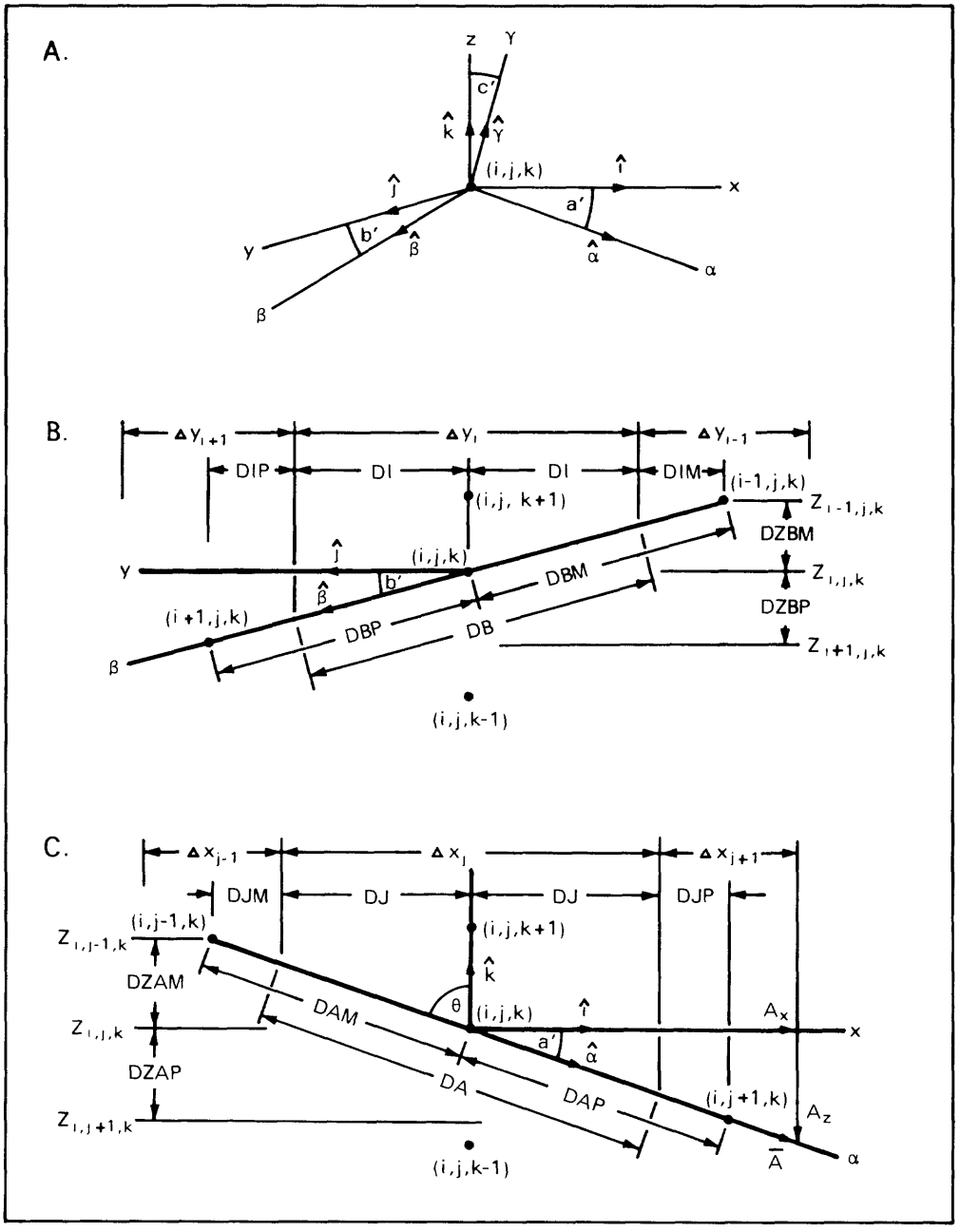


Figure 2.--(A) Variable density α,β,γ coordinate system relative to x,y,z coordinate system, and geometric terms used in program VARDEN in (B), y,β and (C), x,α planes.

$$\rho'_{i,j+1/2,k} = \frac{\rho_{s_{i,j,k}} + \rho_{s_{i,j+1,k}}}{2\rho_f}.$$

The Z terms of equations 2a-d are the differences in elevation between adjacent nodes in the α and β directions. The term $\frac{\Delta x}{\Delta \alpha}\frac{\Delta y}{\Delta \beta}$ of equations 2e and 2f, where Δx and Δy are the horizontal dimensions of the finite-difference block i,j,k of the aquifer, approximates the gradient of elevation in the direction normal to the plane of the aquifer (fig.2A). A derivation of this approximation is given in the discussion of program VARDEN (p. 13).

The conductance terms of equations 1 and 2 formed from the distance-weighted harmonic mean of hydraulic properties of adjacent nodes, are given by:

$$TR_{i,j-1,k} = \frac{\Delta\beta\left(\frac{K_{\alpha\alpha}b\rho_\nu}{\mu'}\right)_{i,j-1/2,k}}{\alpha_{i,j,k} - \alpha_{i,j-1,k}} = \frac{2\Delta\beta\left(\frac{K_{\alpha\alpha}b\rho_\nu}{\mu'}\right)_{i,j,k} \cdot \left(\frac{K_{\alpha\alpha}b\rho_\nu}{\mu'}\right)_{i,j-1,k}}{\Delta\alpha_{i,j,k}\left(\frac{K_{\alpha\alpha}b\rho_\nu}{\mu'}\right)_{i,j-1,k} + \Delta\alpha_{i,j-1,k}\left(\frac{K_{\alpha\alpha}b\rho_\nu}{\mu'}\right)_{i,j,k}}. \quad (3a)$$

$$TR_{i,j,k} = \frac{2\Delta\beta\left(\frac{K_{\alpha\alpha}b\rho_\nu}{\mu'}\right)_{i,j,k} \cdot \left(\frac{K_{\alpha\alpha}b\rho_\nu}{\mu'}\right)_{i,j+1,k}}{\Delta\alpha_{i,j,k}\left(\frac{K_{\alpha\alpha}b\rho_\nu}{\mu'}\right)_{i,j+1,k} + \Delta\alpha_{i,j+1,k}\left(\frac{K_{\alpha\alpha}b\rho_\nu}{\mu'}\right)_{i,j,k}}, \quad (3b)$$

$$TC_{i-1,j,k} = \frac{2\Delta\alpha\left(\frac{K_{\beta\beta}b\rho_\nu}{\mu'}\right)_{i,j,k} \cdot \left(\frac{K_{\beta\beta}b\rho_\nu}{\mu'}\right)_{i-1,j,k}}{\Delta\beta_{i,j,k}\left(\frac{K_{\beta\beta}b\rho_\nu}{\mu'}\right)_{i-1,j,k} + \Delta\beta_{i-1,j,k}\left(\frac{K_{\beta\beta}b\rho_\nu}{\mu'}\right)_{i,j,k}}, \quad (3c)$$

$$TC_{i,j,k} = \frac{2\Delta\alpha\left(\frac{K_{\beta\beta}b\rho_\nu}{\mu'}\right)_{i,j,k} \cdot \left(\frac{K_{\beta\beta}b\rho_\nu}{\mu'}\right)_{i+1,j,k}}{\Delta\beta_{i,j,k}\left(\frac{K_{\beta\beta}b\rho_\nu}{\mu'}\right)_{i+1,j,k} + \Delta\beta_{i+1,j,k}\left(\frac{K_{\beta\beta}b\rho_\nu}{\mu'}\right)_{i,j,k}}, \quad (3d)$$

$$TK_{i,j,k-1} = \Delta\alpha\Delta\beta\left(\frac{K_{\gamma\gamma}\rho_\nu}{b_l\mu'}\right)_{i,j,k-1/2}, \text{ and} \quad (3e)$$

$$TK_{i,j,k} = \Delta\alpha\Delta\beta\left(\frac{K_{\gamma\gamma}\rho_\nu}{b_l\mu'}\right)_{i,j,k+1/2} \quad (3f)$$

In equation set 3, $K_{\alpha\alpha}$, $K_{\beta\beta}$, $K_{\gamma\gamma}$ are the components of the hydraulic-conductivity tensor in the α, β, γ direction (L/T) of the equivalent freshwater aquifer having ground water of density ρ_f and dynamic viscosity of μ_f, and

ρ_p = pure-water density of node i,j,k defined to be the difference between ground-water density, ρ_s, and the mass of solute per unit volume of solution;

μ' = relative viscosity, defined to be the ratio of the dynamic viscosity of the ground water of interest (μ) to the dynamic viscosity (μ_f) of the equivalent freshwater at 20°C (dimensionless);

b = average thickness of aquifer layer node i,j,k (L)

$\Delta\alpha, \Delta\beta$ = inclined dimensions of the finite-difference block of the aquifer, (L) (fig.2); and

b_U, b_L = thickness of confining layers above and below node i,j,k (L).

As discussed in Mandle and Kontis (in press), the variable-density formulation is based on the assumption that ground-water density varies spatially but is time invariant over the period of simulation. If pumping centers are far from areas of variable density, changes in head over time are assumed to not significantly alter the density and viscosity distribution; therefore the formulation (eq. 1) may be applied for transient simulation. Accordingly, on the right-hand side of equation 1:

S* = modified storage coefficient and is the product of the pure-water density (ρ_p), storage coefficient (S), and block area ($\Delta\alpha\Delta\beta$);

Δh_f = change in freshwater head per time increment (Δt);

Q* = product of pure-water density and Q; a source (positive) or sink (negative) term representing addition or removal of water from the node.

Thus:

$$S^* = (\Delta\alpha\Delta\beta\rho_p S)_{i,j,k}, \text{ and}$$

$$Q^* = (\rho_p Q)_{i,j,k}. \tag{4}$$

In equation 1, the variable-density terms given by equations 2a-f are known and are independent of head so that they can be incorporated into a single term ($TS_{i,j,k}$). Thus, equation 1 can be rewritten as:

$$\begin{aligned} &-TR_{i,j-1,k}(h_{f_{i,j,k}} - h_{f_{i,j-1,k}}) + TR_{i,j,k}(h_{f_{i,j+1,k}} - h_{f_{i,j,k}}) \\ &-TC_{i-1,j,k}(h_{f_{i,j,k}} - h_{f_{i-1,j,k}}) + TC_{i,j,k}(h_{f_{i+1,j,k}} - h_{f_{i,j,k}}) \\ &-TK_{i,j,k-1}(h_{f_{i,j,k}} - h_{f_{i,j,k-1}}) + TK_{i,j,k}(h_{f_{i,j,k+1}} - h_{f_{i,j,k}}) \\ &= S^* \frac{\Delta h_{f_{i,j,k}}}{\Delta t} - Q^* - TS_{i,j,k}, \end{aligned} \tag{5a}$$

where

$$TS_{i,j,k} = A_{i,j-1,k} - A_{i,j,k} + B_{i-1,j,k} - B_{i,j,k} + \Gamma_{i,j,k-1} - \Gamma_{i,j,k} . \quad (5b)$$

Also

$$PSA_{i,j,k} = + A_{i,j,k} - A_{i,j-1,k} ,$$
$$PSB_{i,j,k} = + B_{i,j,k} - B_{i-1,j,k} , \text{ and}$$
$$PS3_{i,j,k} = + \Gamma_{i,j,k} - \Gamma_{i,j,k-1} . \quad (5c)$$

The form of equation 5a is the same as that of the finite-difference equation of the Trescott-Larson model except for the TS term. This term represents the net effect of gravity forces exerted on the flow by the mass of ground water in a tilted aquifer block.

In areas containing relatively fresh water, the individual variable-density terms A, B, and Γ (eq. 2a-f) are zero or very small so that the magnitude of flow across model block faces is proportional to freshwater-head gradient, and flow is essentially orthogonal to contours of freshwater head. In areas containing brines, and where aquifers are not horizontal, the flow may be significantly altered by the A, B, and Γ terms, and resultant flow directions will not be orthogonal to lines of equal freshwater head. The condition for no flow in these areas is that the freshwater-head gradient in each direction be exactly opposed by the effect of gravity forces in that direction.

The volumetric flow components for any node i,j,k are computed by:

$$QR_{i,j,k} = [-TR_{i,j,k}(h_{f_{i,j+1,k}} - h_{f_{i,j,k}}) + A_{i,j,k}]/\rho_{p_{i,j+1/2,k}} , \quad (6a)$$

$$QC_{i,j,k} = [-TC_{i,j,k}(h_{f_{i+1,j,k}} - h_{f_{i,j,k}}) + B_{i,j,k}]/\rho_{p_{i+1/2,j,k}} , \text{ and} \quad (6b)$$

$$QK_{i,j,k} = [-TK_{i,j,k}(h_{f_{i,j,k+1}} - h_{f_{i,j,k}}) + \Gamma_{i,j,k}]/\rho_{p_{i,j,k+1/2}} , \quad (6c)$$

where:

$QR_{i,j,k}$ = flow along rows between nodes i,j,k and i,j+1,k (L^3/T)

$QC_{i,j,k}$ = flow along columns between nodes i,j,k and i+1,j,k (L^3/T), and

$QK_{i,j,k}$ = flow across confining layers between nodes i,j,k and i,j,k + 1 (L^3/T), and

the pure-water density term in each expression is the average value for the pair of nodes in question.

The variable-density finite-difference equation in terms of pressure (P) can be readily obtained from the relation $h_f = \frac{P}{\rho_f g} + Z$. The conversion to a pressure formulation may be implemented by (1) replacing all terms in equations 1, 5, and 6 containing h_f by P, (2) dividing all conductances (eq. 3a-f) by $\rho_f g$, (3)replacing all terms containing $(\rho'-1)$ in equations 2a-f by ρ', and (4) multiplying all terms containing ρ' by $\rho_f g$.

The finite-difference variable-density flow formulation (eq. 5a) was tested by comparing model results with the results of a cross-sectional electric analog model of a freshwater/saltwater-interface problem discussed by Bennett and Giusti (1971). Results of the comparison are shown in figure 3, which depicts the simulated north-south steady-state flow system of an alluvial water-table aquifer along the south-central coast of Puerto Rico. In the analog simulation of Bennett and Guisti, the northern boundary (figs. 3B, 3C) and upper boundary (fig. 3A) were treated as constant heads. The northern constant-head boundary represents uniform flow from the north, and the upper constant-head boundary represents a rainy-season, high-water-table condition north of the coast and heads within the seabed seaward of the coast. The bottom of the model (bedrock) and a freshwater/saltwater interface to the south were treated as zero-flow boundaries. With this configuration and a horizontal hydraulic conductivity of 33 ft/d for alluvium and a vertical anisotropy in conductivity of 1,000, Bennett and Guisti obtained a head pattern shown by the dotted contours in figure 3B.

To test the finite-difference variable-density formulation, the cross section was discretized into 11 layers, each with a thickness of 50 feet (ft) and a uniform spacing of 800 ft along each layer. The layers were assumed to dip uniformly seaward at an angle of 0.25° (23 ft per mile). Except for the freshwater/saltwater interface, all boundaries were treated the same as in the analog model. A profile of the constant-head upper layer is shown in figure 3A. For the variable-density formulation, pure water density, ρ_p, and relative viscosity, μ', were taken as unity in equations 3a-f and the density of salty ground water (eq. 2a-f) was set at 1.025 grams per cubic centimeter (g/cm^3).

The freshwater heads obtained with the finite-difference model are shown in figure 3B, and a representation of the flow is given in figure 3C. In figure 3C, the tail of each vector is at a model node, and the length and direction of each vector is the resultant of flow within the layer (eq. 6a) and flow across the layer (eq. 6c). The largest vector has a magnitude of 0.026 cubic feet per second (ft^3/s), and the length of all other vectors is proportional to this maximum length. The magnitude is also depicted by the dotted lines of equal flow in figure 3C. Seaward of the freshwater/saltwater interface, the magnitude of flow is quite small (less than 0.005 ft^3/s), and the direction of flow is essentially random. Within this region, the freshwater-head difference between nodes of adjacent layers (fig. 3B) is about 1.3 ft. The hydraulic conductance between layers is 0.00941 (slug/s)/ft (slug per second per foot,) and the Γ source term (eq. 2f) is -0.012 slug/s. Consequently, from equation 6c, the flow between layers (QK) in this region is (1.3 x 0.0094 - 0.012) divided by the density of pure water--essentially zero flow.

Characteristics of Water-Density Terms

The effects of local variations in water density on the simulated ground-water flow system are manifested directly through the term $TS_{i,j,k}$ of equation 5a. Depending on the sign of this term, a quantity of water is, in effect, added or removed from each model node containing a relative ground-water density, ρ', other than unity. Because of this, the TS term may be referred to as a total-source term and A,B,Γ (eq. 2a-f) and PSA, PSB and PS3 (eq. 5c) as individual or component-source terms. The magnitude and sign of the total-source term depends on the relative magnitudes of hydraulic conduc-

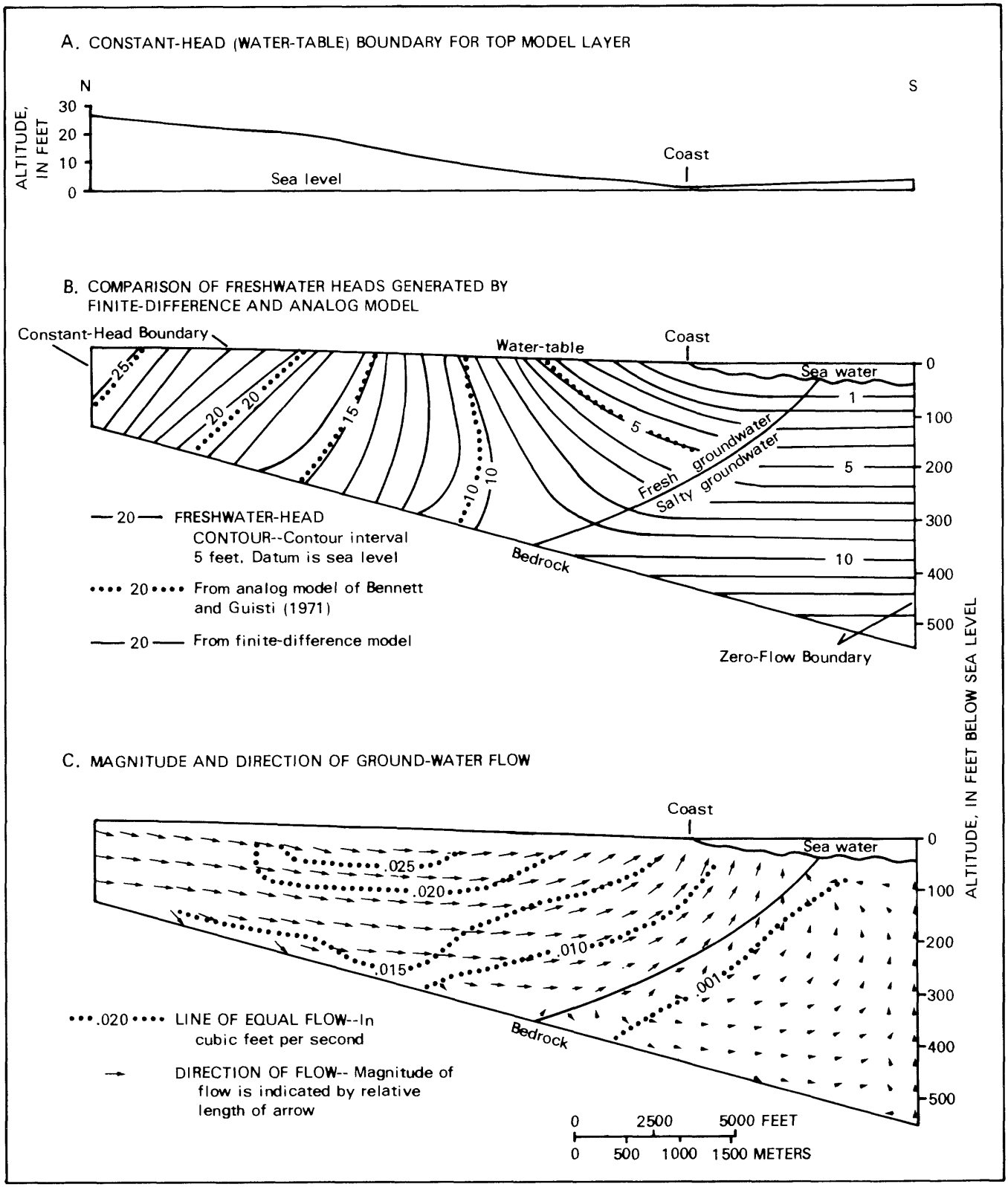


Figure 3.--Results of variable-density finite-difference cross-sectional model of an area near Ponce, Puerto Rico.

tivity, pure-water and ground-water density, viscosity, aquifer and confining-layer thickness, and elevation gradient between the node center and the adjacent node centers according to equations 3a-f and equations 2a-f. Consequently, whether or not the total-source term for a particular node represents a source or sink is difficult to determine from cursory inspection of the data.

The disposition of the component source terms A, B, and Γ of equations 2a-f used to compute flow in the α,β, and γ directions is more readily apparent, however. In areas where the relative ground-water density is greater than unity, $(\rho'-1)$ is positive, so that the sign of the individual source terms A and B governing flow within a layer is given by the elevation gradient in the appropriate direction. For example, if the elevation of node i,j+1,k is less than that of node i,j,k, then $A_{i,j,k}$ is always positive. In addition, as shown in equation 2f, areas where relative ground-water density is greater than unity will always have a negative Γ term.

To illustrate the nature of the total and component-source terms, their values for a simple example consisting of a cross section through an idealized, v-shaped, two-dimensional basin are shown in figures 4 and 5. The basin consists of three aquifer layers separated by confining layers. The layers have a constant elevation gradient of 1000 ft per model node spacing (fig. 4A). Layers 1 and 2 contain ground water whose relative density is unity at and near the basin margins but increases linearly toward the basin trough. The relative density of the topmost layer (layer 3) is that of freshwater. The hydraulic conductance (TR) in the α direction (perpendicular to the trough) is set at 0.001 (slug/ft)/s for all nodes of layers 1 and 2. With these values, the A source terms for layer 1 and 2 have the form shown in figure 4B. The A source term for both layers is positive along the descending limb (negative elevation gradient) and negative along the ascending limb (positive elevation gradient).

For the Γ term, the product of all terms in equations 2e and 2f, except for the $(\rho'-1)$ term, is assumed to be 1 slug/s between the nodes of layers 1 and 2 and layers 2 and 3. For these values and the density distribution of figure 4A, the Γ source terms for layers 1 and 2 have the form shown in figure 4B. Because layer 3 contains freshwater, $(\rho'-1)$ is zero, making the A term zero for all nodes. In addition, layer 3 has no Γ term because it is the topmost layer.

The components of the total-source terms (PSA and PS3) in the α and γ directions according to equation 5c are given in figure 5A and 5B. In a simulation of the system, the actual input to the model for each node would consist of the total-source term, TS, obtained by summing these components over each node by equation 5B. As shown in figure 5A, the total-source terms for layer 1 consist primarily of positive contributions with a relatively large positive source at the trough of the basin. The total-source terms for layer 2 (fig. 5B) are negative except for the node at the trough. The total sources for layer 3, equivalent to the Γ terms of layer 2 (fig. 4B) (and not shown on fig. 5), are negative. In a simulation, the relatively large positive total sources of layers 1 and 2 at node 11 (fig. 5) would cause a buildup of freshwater head at the center of the basin. Note that the sum of the total-source terms over all layers (TS of fig. 5A (+0.28 slug/s) + TS of fig. 5B (-0.20 slug/s) + Γ of layer 2 in fig. 4B (-0.08 slug/s)) is zero, as required by conservation of mass.

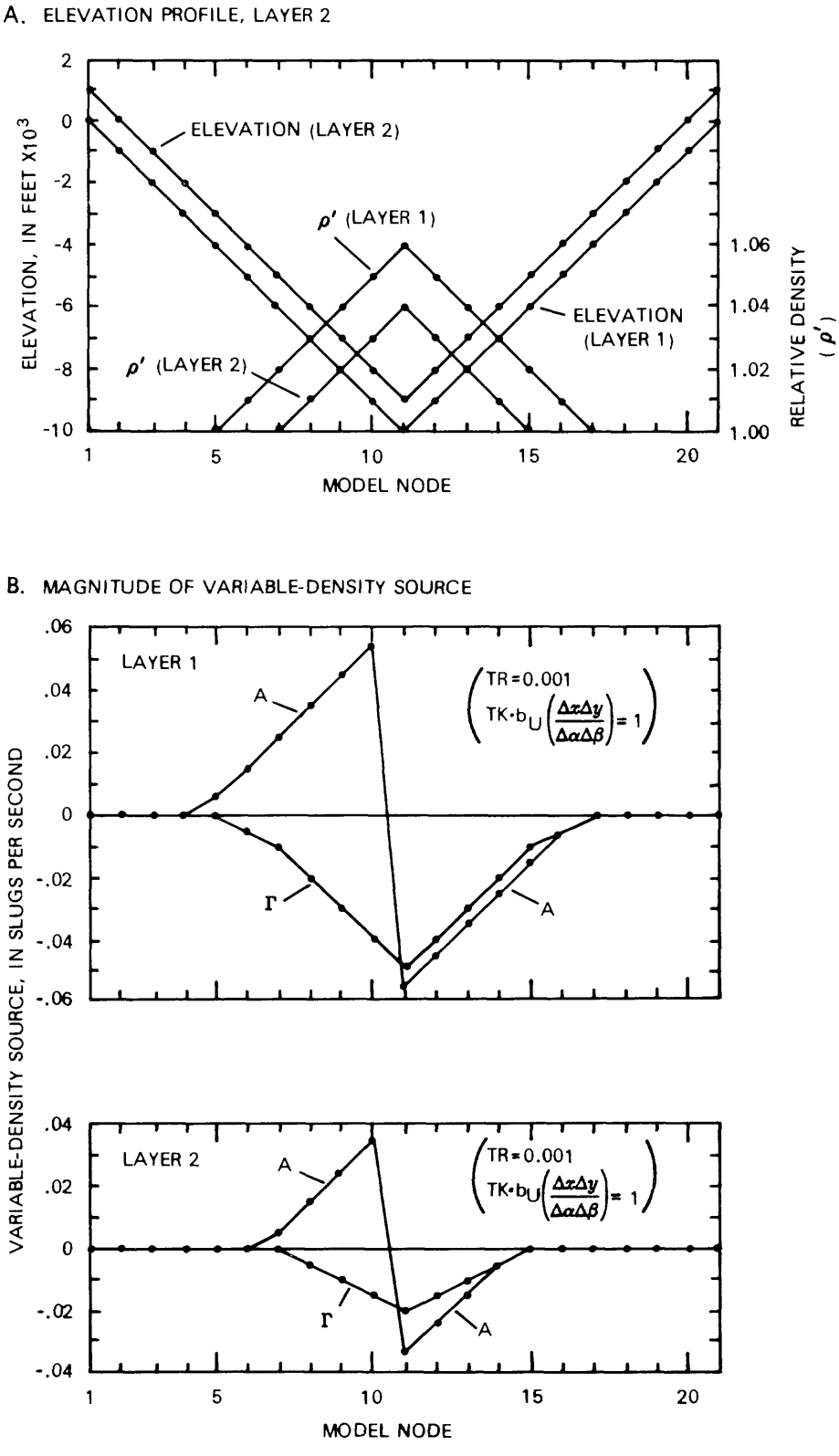


Figure 4.--Profiles of (A) elevation and relative ground-water density, and (B) magnitude of component source terms (A and Γ) for idealized two-dimensional basin.

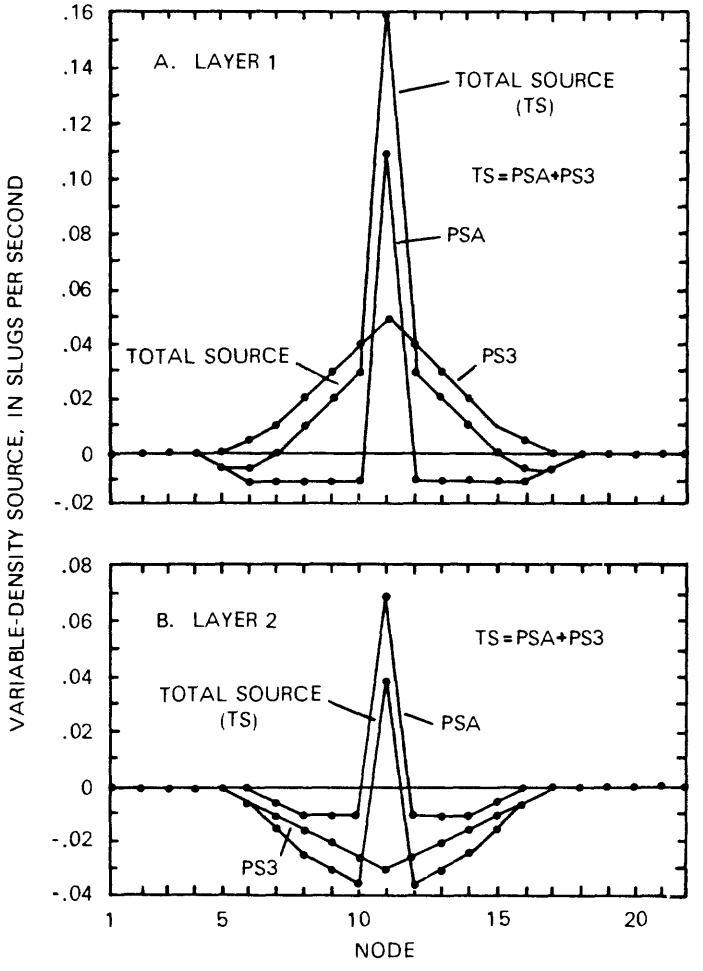


Figure 5.

Magnitude of variable water-density source terms (PSA, PS3, and TS) for idealized two-dimensional basin.

Description of Program VARDEN

To obtain the variable-density source terms and conductances required for input to the Trescott-Larson finite-difference model, in terms of either freshwater head or pressure, a Fortran computer program, VARDEN (VARiable DENsity), was developed from a program originally written by Weiss (1982a). VARDEN computes and stores exterior to the Trescott-Larson model, the TR, TC, TK conductances (eq. 3a-f) and the modified storage coefficient S* (eq. 4). In addition, the individual density terms A, B, and Γ (eqs. 2a-f) for use in postmodel flow calculations (eq. 6a-c) and a modified transmissivity for use in the simulation of the effects of multiaquifer wells, discussed in a later section are computed and stored. For isotropic conditions, the modified transmissivity is:

$$T^*_{i,j,k} = \left(\frac{K_{\alpha\alpha} b \rho_\mu}{\mu'}\right)_{i,j,k} = \left(\frac{K_{\beta\beta} b \rho_\mu}{\mu'}\right)_{i,j,k}, \qquad (7)$$

The input to the program consists of a single control record containing the size of the model and options to be used, followed by a sequence of arrays (table 1A). For these input arrays, table 1 gives the program names, the corresponding names of equations 1-7, and the number of arrays required for a model with KO layers. All arrays, except model-grid spacing, are presumed to reside in a single file (unit LIN). The input transmissivity and confining

Table 1.--Program VARDEN (VARiable DENsity) Input and Output Arrays

CONTROL RECORD IN 6I5 FORMAT

Name in program	Description
I0	Number of model rows
J0	Number of model columns
K0	Number of model layers
LIN	Fortran unit number for input arrays
IPRESS =	{ 0 for freshwater formulation 1 for pressure formulation
IDELTA =	{ 1 if confining layer conductivity is to be modified 0 otherwise

A. INPUT ARRAYS

Name in program	Description	Name in Eq. 1-7	Number of arrays	Fortran unit number	Format	Units
S	Storage coefficient	S	K0	LIN	10E8.2	--
T	Transmissivity	$K_{\alpha\alpha}{}^{*}b$, $K_{\beta\beta}{}^{*}b$	K0	LIN	10E8.2	ft^2/s
ELEV	Elevation of nodes	Z	K0	LIN	10F8.0	ft
DEN	Ground water density	ρ_s	K0	LIN	10F8.1	kg/m^3
DENP	Pure water density	ρ_p	K0	LIN	10F8.1	kg/m^3
TKV	Confining layer leakance	TK	K0-1	LIN	10E8.2	1/s
THK	Confining layer thickness	b_U, b_L	K0-1	LIN	10F8.0	ft
VISC	Viscosity	μ	K0	LIN	10F8.3	cp
DELTA	Factors to modify T and TK	--	K0	LIN	10F8.3	--
DELX	Node spacing along rows	Δx	1	5	8G10.0	ft
DELY	Node spacing along columns	Δy	1	5	8G10.0	ft

B. OUTPUT ARRAYS

Name in program	Description	Name in Eq. 1-7	Number of arrays	Fortran unit number	Format	Units
	Arrays for input to Trescott model					
SOUT	Modified storage coefficient	S*	K0	19	10F8.0	slug/ft
TOUT	Modified transmissivity	$K^{*}b\rho_p/\mu'$	K0	18	8E10.4	(slug/s)/ft
TK	Conductance between layers	TK	K0-1	17	8E10.4	(slug/s)/ft
PST	Total source term	TS	K0	16	8E10.4	slug/s
TR	Conductance along rows	TR	K0	11	8E10.4	(slug/s)/ft
TC	Conductance along columns	TC	K0	12	8E10.4	(slug/s)/ft
	Arrays for postmodel computation of flow					
ALPHA	Source term for rows	A	K0	13	8E10.4	slug/s
BETA	Source term for columns	B	K0	14	8E10.4	slug/s
GAMMA	Source term for layers	Γ	K0-1	15	8E10.4	slug/s

layer leakance arrays are estimates of the hydraulic properties for freshwater conditions. The ground-water and pure-water density and viscosity arrays are the output from a procedure (program STEAM) described on p. 18. The model grid spacing is that of a conventional discretization of the aquifer system for a horizontal plane at the surface.

In table 1, the DELTA array is a set of numerical factors for each node that may be used to modify model conductivities to account for possible decrease in conductivity as a function of overburden thickness. A discussion of this procedure, as used in the northern Midwest RASA simulation, is given in Mandle and Kontis (in press). As noted in table 1 and in program VARDEN (appendix B), if the constant IDELTA is set to zero and the DELTA arrays are set to unity, the conductivities will not be modified by the program.

The program output consists of the arrays given in table 1B and, if desired, a printed copy of all input and output data as well as the summation of source terms over each layer and over all layers. Information concerning the format of input and output arrays and details concerning the functioning of the program and its use are given in comment statements within the program listing (appendix B).

The modified Trescott-Larson program is based on equations with units of mass, length, and time in slugs, feet, and seconds, respectively. The procedure that computes density and viscosity (program STEAM) generates density in kilograms per cubic meter and viscosity in centipoise. In program VARDEN, ground-water density is read as kilograms per cubic meter and is converted first to grams per cubic centimeter and then to a relative density (ρ'= DEN) by division by the ambient density (ρ_f) of relatively fresh water in the study area as computed by program STEAM. For the northern Midwest RASA, ρ_f = 0.9977 g/cm^3. Pure-water density (ρ_p) is likewise converted to a relative density (DENP) when it is read and then is converted to density, in units of slugs per cubic foot. This is done by multiplying by the factor UNIT1, where:

$$\text{UNIT1} = 0.9977\ \text{g/cm}^3\ \frac{(6.852 \times 10^{-5}\ \text{slug/g})}{(3.531 \times 10^{-5}\ \text{ft}^3/\text{cm}^3)} = 0.9977 \times 1.94\ \text{slug/ft}^3$$

Viscosity in centipoise is converted to a relative viscosity (μ' = VISC) by division by the ambient viscosity of relatively fresh ground water of the study area, as computed by program STEAM. The value for the northern Midwest RASA was 1.127 centipoise. If the model is to be executed in terms of freshwater head, the above quantities, as used in VARDEN, produce conductances in slugs per second per foot and source terms in slugs per second.

If the model is to be executed in terms of pressure (in slugs per foot second squared, the source terms of equations 2a-f are multiplied by a factor termed UNIT2, where:

$$\text{UNIT2} = \rho_f g = (1.94 \times .9977)\ \text{slugs/ft}^3 \times 32.174\ \text{ft/s}^2,$$

and all conductances in equations 3a-f are divided by the same factor. Because pure-water density appears in the program as a relative density, the actual division is by g = 32.174 ft/s^2. For the pressure case, VARDEN produces conductance in units of second-feet and source terms in units of slugs per second. The values of UNIT1 and UNIT2 factors are set in the program (appendix B).

To obtain the geometric terms required for equations 2 and 3, a local coordinate system α, β, γ is developed at each node from specified horizontal grid spacings Δx_j and Δy_i in the x,y plane and elevations $Z_{i,j,k}$ of the midpoints of the aquifer layers. At each node location in the x,y plane, the projection of the x axis, within the x,z plane, onto the plane of an aquifer layer defines the orientation of the α axis. Thus, as shown in figure 2C, the α axis is constrained to lie in the x,z plane. Similarly, the projection of the y axis onto the aquifer plane defines the β axis constrained to lie in the y, z plane. In figure 2C, let $\overline{A}$ be a vector in the α direction. The components of $\overline{A}$ in the x and z directions are A_x and A_z, that is, for unit vectors $\hat{i}$ and $\hat{k}$ in the x and z directions.

$$\overline{A} = A_x\hat{i} + A_z\hat{k} = A \cos a'\hat{i} + A \sin a'\hat{k}.$$

The unit vector $\hat{\alpha}$ in the α direction is then:

$$\hat{\alpha} = \frac{\overline{A}}{|\overline{A}|} = \frac{A(\cos a'\hat{i} + \sin a'\hat{k})}{A(\cos^2 a' + \sin^2 a')^{1/2}} = \cos a'\hat{i} + \sin a'\hat{k}$$

In similar manner, a unit vector $\hat{\beta}$ in the β direction (fig. 2B) in terms of unit vectors $\hat{j}$ and $\hat{k}$ in the y and z directions is given by

$$\hat{\beta} = \cos b'\hat{j} + \sin b'\hat{k}.$$

The vectors $\hat{\alpha}$ and $\hat{\beta}$ lie in the α , β plane so that a vector $\overline{\gamma}$ normal to the α, β plane (fig. 2A) is the vector product of $\hat{\alpha}$ and $\hat{\beta}$, or

$$\overline{\gamma} = \hat{\alpha} \times \hat{\beta} = - \cos b' \sin a'\hat{i} - \cos a' \sin b'\hat{j} + \cos a' \cos b'\hat{k}.$$

If $\hat{\gamma}$ is a unit vector in the γ direction, and if c' is the angle between the γ and z axes then for small angles a', b':

$$\cos c' = \hat{\gamma}\cdot\hat{k} = \frac{\overline{\gamma}\cdot\hat{k}}{|\overline{\gamma}|} = \frac{\cos a' \cos b'}{(1-\sin^2 a' \sin^2 b')^{1/2}} = \sim \cos a' \cos b' = \frac{\Delta x \Delta y}{\Delta\alpha\Delta\beta}$$

This relation is used to approximate the gradient of elevation in the γ direction. As pointed out by Kuiper (1983), the α and β directions defined in this manner are not, in general, normal to each other. For the small dips (generally less than several degrees) that usually occur in nature, the angle between α and β is almost 90°, and the α, β, γ system is essentially orthogonal.

For aquifers that dip uniformly, the geometric relations given below are exact, whereas for aquifers with small dip and where changes in dip are gradual the relations are a good approximation. Conversely, for strata characterized by large and rapidly changing dips, the computations of inclined block widths may be erroneous. With regard to the general problem of correctly simulating complex aquifer geometry, regardless of the density distribution, Weiss (1985)

developed the ground-water-flow equation for uniform-density conditions in curvilinear coordinates that follow the bedding of strata and that rigorously take into account changes in dip. By comparing the simulated flow system obtained from finite-difference approximations based on conventional x,y,z cartesian coordinates and on curvilinear coordinates for several test cases, Weiss concluded that the hydraulic effects of variable dip was relatively small, especially for dips less than several degrees.

The geometric terms consisting of inclined distances and inclined block widths $\Delta\alpha$ and $\Delta\beta$ are approximated in the program (appendix A) as follows. Figure 2C shows three adjacent nodes (i,j-1,k), (i,j,k) and (i,j+1,k) for a layer dipping in the α direction with horizontal node spacing of Δx_{j-1}, Δx_j, and Δx_{j+1}, respectively. The variables used in the program (appendix A) consist of the terms listed below.

DJM = $(\Delta x_{j-1})/2$,
DJ = $(\Delta x_j)/2$,
DJP = $(\Delta x_{j+1})/2$,
DAM = inclined distance between node i,j-1,k and node i,j,k $(\alpha_{i,j,k} - \alpha_{i,j-1,k})$,
DAP = inclined distance between node i,j,k and node i,j+1,k $(\alpha_{i,j+1,k} - \alpha_{i,j,k})$,
DA = width of block i,j,k in α direction $(\Delta\alpha_{i,j,k})$,
DZAM = difference in elevation of nodes i,j-1,k and i,j,k,
DZAP = difference in elevation of nodes i,j,k and i,j+1,k.

From figure 2,

$$DAM = ((DJM+DJ)^2 + DZAM^2)^{1/2}$$
$$DAP = ((DJ+DJP)^2 + DZAP^2)^{1/2}$$

If Θ is the angle between the vertical (z) and α directions (fig. 2C),

$$SIN\Theta = DJ/(DA/2) = (DJM+DJ)/DAM,$$

so that the block width is

DA = 2*DJ*DAM/(DJ+DJM).

The block width for node i,j+1,k is

DA2 = 2*DJP*DAP/(DJ+DJP).

The block width DB = $\Delta\beta$, in the β direction, is developed in a similar manner from the terms of figure 2B. In the northern Midwest, the maximum dip of strata over model-node distances is less than 1.5 degrees. In general, for most aquifer systems, maximum dip of strata is probably less than several degrees, and the angles a',b',c' (fig. 2A) are quite small, so that slant distances between nodes and inclined block areas ($\Delta\alpha$ and $\Delta\beta$) and, in turn, model conductances TR, TC, and TK, will be close in magnitude to their corresponding values for an

x,y,z coordinate system. Consequently, elevation changes are significant only with respect to computation of the variable density terms A and B and less so for the Γ component (eq. 2f).

Description of Program STEAM

Ground-water density, pure-water density, and viscosity arrays for use in VARDEN may be obtained from a modified version (program STEAM) of a method developed and documented in detail by Weiss (1982a). The method is based on the assumption that the density of ground water is equivalent to the density of a sodium chloride solution of the same molality, temperature, and pressure. Under this assumption, the density of ground water at node i,j,k as a function of temperature at constant composition and pressure, can be computed through a relation from Potter and Brown (1977):

$$\rho_{s_{i,j,k}} = [C_a + C_b T_g + C_c T_g^2]_{i,j,k} , \tag{8}$$

where:

T_g = temperature of ground water, in degrees Celsius, and

C_a, C_b, C_c = empirical coefficients for sodium chloride solutions given in table 29 of Potter and Brown (1977).

The coefficients are given for values of molality ranging from 0.5 to 6.0 moles per kilogram (mol/kg) in increments of 0.5 mol/kg and of pressure ranging from standard atmospheric to 2,000 bars (29,000 lb/in^2) in increments of 100 bars (1,450 lb/in^2) up to 1,000 bars and 250 bars thereafter. Part of table 29 from Potter and Brown (1977) is reproduced in table 2. In addition, table 2 contains C_a, C_b, and C_c coefficients for zero molality as determined by R. J. Sun (U.S. Geological Survey, written commun., 1981).

The molality for node i,j,k is computed iteratively by the relation (Weiss, 1982a).

$$XMOL_{i,j,k} = \frac{DS}{\left[wt\left(DEN1 - \frac{DS}{1000}\right)\right]_{i,j,k}} , \tag{9}$$

where:

XMOL = approximate molality of ground water, in mol/kg,

DS = dissolved-solids concentration of ground water, in g/L at laboratory conditions (20 °C and 1 bar),

wt = gram molecular weight of sodium chloride (58.4428 g/mol), and

DEN1 = density of ground water at 20°C and 1 bar, in g/cm^3.

Table 2.--NaCl C_a, C_b, C_c interpolation coefficients

[From table 29 of Potter and Brown, 1977 and R. J. Sun, U.S. Geological Survey, written commun., 1981. SAT = standard atmospheric pressure.]

A. C_a INTERPOLATION COEFFICIENTS

		Pressure (bars)				
		SAT	100	200	300	400
I	Molality	J=1	J=2	J=3	J=4	J=5
1	0.0	1.00184	1.00667	1.01121	1.01548	1.01948
2	0.5	1.00590	1.02100	1.02600	1.03100	1.03400
3	1.0	1.03120	1.04300	1.04700	1.05100	1.05500
4	1.5	1.04900	1.06300	1.06800	1.07100	1.07500
5	2.0	1.07700	1.08100	1.08500	1.08800	1.09300
6	2.5	1.09900	1.09800	1.10200	1.10700	1.10900
7	3.0	1.11700	1.11600	1.11900	1.12300	1.12500
8	3.5	1.13700	1.13500	1.13500	1.13800	1.14000
9	4.0	1.15500	1.15000	1.15200	1.15400	1.15600
10	4.5	1.16800	1.16600	1.16800	1.17000	1.17200
11	5.0	1.18400	1.18300	1.18300	1.18500	1.18700
12	5.5	1.19800	1.19400	1.19700	1.19900	1.20100
13	6.0	1.21300	1.21000	1.21200	1.21500	1.21700

B. C_b INTERPOLATION COEFFICIENTS TIMES -0.0001

I	Molality	J=1	J=2	J=3	J=4	J=5
1	0.0	1.50194	1.29778	1.48770	1.65600	1.80564
2	0.5	- .31800	.23900	.86800	1.07700	.30500
3	1.0	.74300	1.20000	1.39900	1.41700	.84300
4	1.5	1.02800	1.61300	1.86900	1.87800	1.12100
5	2.0	2.76900	1.99400	1.60200	2.22200	1.41800
6	2.5	3.60200	1.66600	1.93900	2.63100	1.62200
7	3.0	4.13600	2.54800	2.23900	2.89400	1.79800
8	3.5	4.74600	3.43900	2.60800	2.85500	1.91400
9	4.0	5.10600	3.74100	3.35400	3.16100	2.23300
10	4.5	5.08400	4.10300	3.88100	3.68800	2.64400
11	5.0	5.33400	5.58800	3.96700	3.77200	2.79300
12	5.5	5.29200	4.26600	4.06600	3.87000	2.93200
13	6.0	5.42800	4.64200	4.46700	4.29400	3.54300

C. C_c INTERPOLATION COEFFICIENTS TIMES -0.000001

I	Molality	J=1	J=2	J=3	J=4	J=5
1	0.0	2.73006	3.14091	2.97626	2.83139	2.70357
2	0.5	3.09900	2.97400	2.59200	2.42700	2.56900
3	1.0	2.66100	2.52000	2.33600	2.21300	2.32600
4	1.5	2.41300	2.31400	2.10200	2.00100	2.43000
5	2.0	1.88800	2.27600	2.36800	1.80300	1.96700
6	2.5	1.60100	2.41100	2.17300	1.63500	1.84700
7	3.0	1.37800	1.95500	1.99400	1.49600	1.75200
8	3.5	1.17400	1.59300	1.82000	1.46400	1.60800
9	4.0	1.03300	1.31800	1.34300	1.33800	1.52500
10	4.5	.98000	1.16500	1.13700	1.13800	1.36900
11	5.0	.88700	.71700	1.09100	1.09000	1.30300
12	5.5	.85800	1.02800	1.03800	1.04300	1.24200
13	6.0	.80500	.88500	.89200	.89100	1.04700

In the program STEAM (appendix C) the iteration process is implemented by using an initial value of unity for DEN1 and computing molality from equation 9. The resultant value of molality is used in a relation applicable for constant temperature and pressure and for temperature less than 100 °C (eq. 1 of Potter and Brown, 1977). That is:

$$DEN1_{i,j,k} = \left.\frac{(1000+wt\cdot XMOL)DEN0}{1000+(A_0\cdot XMOL+B_0\cdot XMOL^{3/2}+C_0\cdot XMOL^2)DEN0}\right|_{i,j,k}, \qquad (10)$$

where:

DEN0 = density of ground water at 20 °C, 1 bar

A_0, B_0, C_0 = empirical constants (see table 28, Potter and Brown, 1977).

For the next iteration, the value of density computed from equation 10 is used in equation 9 to recompute molality. This process is repeated until the percentage of molality difference between successive iterations is less than some specified closure criteria (CDIFF of Program STEAM).

For the northern Midwest study, estimates of the total dissolved-solids concentration were obtained from regional data compiled by D. I. Siegel (U.S. Geological Survey, written commun., 1981).

The pressure in bars at node i,j,k is assumed to be hydrostatic and computed from:

$$P_{i,j,k} = 980.665 \times 10^{-6}\ g/cm^2s^2 \times 30.48\ cm/ft \times Z'_{i,j,k} \qquad (11)$$

where:

$Z'_{i,j,k}$ = distance in feet from the surface to the midpoint of node i,j,k

The procedure used by Weiss (1982a) to obtain the C_a, C_b, and C_c coefficients required for equation 8 is based on linear interpolation of table values for the computed molality and pressure values. An alternative method is used in the program STEAM. If each set of table values (tables 2A, 2B, and 2C) are considered to be a group of uniformly spaced discrete points from a continuous two-dimensional surface, then the C_a, C_b, and C_c values for use in equation 8, which correspond to computed values of molality and pressure, may be obtained by bicubic interpolation. The procedure consists of fitting a bicubic polynomial (Davis and Kontis, 1970) for each cell defined by the four points (I,J), (I+1,J), (I,J+1), and (I+1, J+1), where I = 1.,...12, and J = 1,..4 of tables 2A, 2B, and 2C. This results in a set of 16 bicubic coefficients for each cell of each table. The interpolation is effected by determining molality and pressure according to equations 9 and 11 and solving a bicubic polynomial. For example, the C_a value for a particular value of pressure (x) and molality (y) is determined from the relation:

$$C_a(x,y)_{j,i} = \sum_{m=0}^{3}\sum_{n=0}^{3} BC_{m,n}^{j,i}\,\frac{(x-x_j)^m}{100}\,\frac{(y-y_i)^n}{0.5}, \qquad (12)$$

where:

x_j = tabulated pressure for column J (table 2A)

y_i = tabulated molality for row I (table 2A) and

BC_{mn} = 16 bicubic coefficients associated with cell J, I (table 2A).

For the same molality and pressure, C_b and C_c are obtained in the same manner from their respective sets of bicubic coefficients. The bicubic coefficients are not given in this report but are available upon request.

After determination of the C_a, C_b, and C_c values, ground-water density is obtained from equation 8. For the northern Midwest, the temperature values Tg for use in equation 8 were obtained from a straight-line fit of temperature versus depth data for the Illinois basin (Bond, 1972) given by:

$$Tg_{i,j,k} = 0.0083Z'_{i,j,k} + 17.09 \ ^\circ C$$

Given the density of ground water, ρ_s, the pure-water component is then obtained from:

$$\rho_{p_{i,j,k}} = \rho_{s_{i,j,k}} \left(1 - \frac{1}{1 + \frac{1000}{XMOL \cdot wt}_{i,j,k}} \right) . \qquad (13)$$

The dynamic viscosity required in the variable-density formulation (eq. 3) is obtained from an empirical relation developed by Weiss (1982b) based on curves of viscosity versus temperature for various values of dissolved-solids concentration (Matthews and Russell, 1967). The relation is:

$$\mu_{i,j,k} = (38.3432/T_g^{1/2} - 14.621/T_g^{1/1} + 1.481)(1 + DS/300)|_{i,j,k} , \qquad (14)$$

where:

$\mu_{i,j,k}$ = dynamic viscosity at node i,j,k, in centipoise, (M/L)/T,

T_g = temperature, in °F, and

DS = dissolved-solids concentration, as previously defined.

The procedure to compute equations 8-14 (program STEAM) is given in appendix C. Comment statements within the program explain the general functions of the program. Table 3 summarizes the required input data and the type of output generated by the program.

Table 3.--Program STEAM input and output arrays

CONTROL RECORD IN (F10.6,3I10) FORMAT		
Name in program	Description	Fortran unit number
CDIFF	Molality iteration closure criterion	5
IPRINT	1 if print of computations at each iteration is desired. 0 otherwise	
IO	Number of model rows	
JO	Number of model columns	

A. INPUT ARRAYS

Name in program	Description	Fortran unit number	Format	Units
TDS	Dissolved solids concentration	10	10F8.3	g/l
H	Freshwater heads	11	10F8.0	ft
ELEV	Elevation of nodes	12	10F8.0	ft
BC	Bicubic coefficients for C_a table values	13	10F8.4	--
BC	Bicubic coefficients for C_b table values	14	10F8.4	--
BC	Bicubic coefficients for C_c table values	15	10F8.4	--

B. OUTPUT ARRAYS

Name in program	Description	Fortran unit number	Format	Units
ZMU	Viscosity array	20	10F8.3	cp
XDEN	Groundwater-density array	21	10F8.1	kg/m^3
XDENP	Pure-water-density array	22	10F8.1	kg/m^3
	Program printout	6		

Water-Density Modifications to Trescott-Larson Program

Modifications of the Trescott-Larson model to implement equation 5a consist of insertion of statements to (1) directly read the TR and TC conductances computed by VARDEN, (2) bypass Entry TCOF in subroutine COEFF, wherein the values of TR and TC were formerly computed, and (3) increase the two-dimensional recharge array, QRE, to three dimensions to provide space for the variable-density source terms, $TS_{i,j,k}$. The specific computer-code modifications are given in appendix D. The statement numbers of appendix D follow the statement-numbering system of the original Trescott-Larson documentation (Trescott, 1975). Modifications were made to the main program and to subroutines DATAI, CHECKI, COEFF, and SOLVE as follows:

Program MAIN.--Modifications given in appendix D consist of statements to label the new input arrays (TR and TC), expand the size of the QRE array by the number of model layers to provide space for the total source arrays (TS), and change the call to Entry ARRAY of subroutine DATAI to read all input arrays from unit LIN. To read the TR and TC conductances, two additional calls to Entry ARRAY are performed.

Subroutine DATAI.--Modifications given in appendix D consist of statements to expand the QRE array to three dimensions and to read input arrays from unit LIN in the correct order.

Subroutine CHECKI.--Modifications given in appendix D consist of statements to expand the QRE array to three dimensions.

Subroutine COEFF.--Modifications given in appendix D consist of statements to expand the QRE array to three dimensions and to bypass computation of TR and TC conductances.

Subroutine SOLVE.--Modifications given in appendix F consist of statements to expand the numbering of the QRE array to three dimensions. Variable water-density modifications of subroutine SOLVE are indicated by dollar signs in columns 73-75.

To execute the Trescott-Larson program modified according to appendix D, all documented procedures given in Trescott (1975) and Trescott and Larson (1976) apply, with the following exceptions.

(1) To invoke the variable-density option, specify the option formerly used to apply areal recharge to the top layer (IQRE). If simulation of areal recharge is desired, the areal recharge can be converted to a mass rate and added to the total-source array for the top layer.

(2) Place starting heads (or pressure) and the following arrays from program VARDEN in a single file (LIN = 11): modified storage coefficient, modified transmissivity, TK conductance, total source term, TR conductance, and TC conductance. The order of the arrays following the starting heads (or pressure) are as given in table 1B.

(3) Because all conductances are precomputed, model node-spacing arrays (DELX and DELY) are set to unity.

With these procedures, the model generates either freshwater head or pressure (depending on what option is selected in VARDEN). Given the computed heads (or pressure), and the conductances (TR, TC, and TK) and individual sources terms (A, B, and Γ) from VARDEN, flow may be computed from the appropriate form of equations 6a-c.

Effects of Multiaquifer Wells

Finite-Difference Equation

Bennett and others (1982) outline a method for accounting, approximately, for the effects of multiaquifer wells in three-dimensional ground-water-flow simulation. Their analysis is based on the assumptions that the area represented by any one model node contains only one multiaquifer well, that the well is exactly at the nodal point, and that the node spacing is uniform and radial symmetry prevails in the head distribution surrounding the well in each aquifer.

A graphic representation of flow and head around a multiaquifer well, with discharge Q_{w_t} and open to four layers, is shown in figure 6 (p. 26). In the figure, h_w is the water level in the multiaquifer well, and $Q_{i,j,k}$ is the flow between layer k and the well. For each layer, the head $h_{i,j,k}$ computed in a simulation for node i,j,k is assumed to prevail in the aquifer along a radius of r_{a_k} around the well and thus around the node i,j,k itself.

Mandle and Kontis (in press) show how the effects of multiaquifer wells may be generalized for the case where a model node has numerous wells and where subsets of the wells may be open to differing combinations of aquifers. In their development, all wells within a finite-difference block area $\Delta x_j . \Delta y_i$ are assumed to be uniformly distributed and are segregated into categories that include those wells open to a unique combination of aquifers. Each well in a given category is assumed to have the same discharge, radius, and water level. The equation, in finite-difference form, to describe total freshwater mass flow, $\rho_p Q^N_{i,j,k}$, between node i,j,k and the wells of all categories (eq. 15) is in a form compatible with the variable-density formulation (eq. 5a). In equation 15:

ρ_p = density of pure water, which can be set at 1.94 slugs/ft^3 if wells are producing relatively fresh water;

$T^*_{i,j,k}$ = modified transmissivity of equation 7;

r_w = radius of the wells;

$hf_{i,j,k}$ = freshwater head in node i,j,k;

u_m = number of wells in category m, (m = 1, 2, 3, ... N) for N categories;

Q_{T_m} = total well discharge from all wells of category m in the block area $\Delta x_j . \Delta y_i$,

kmax = total number of model aquifer layers;

W(k,m) = k if a category m well is open to aquifer layer k, or
0 if a category m well is not open to aquifer layer k; and

$\delta[W(k,m)]$ = 1 if W (k,m) = k or 0 if W (k,m) = 0.

The δ function provides that, in summations over layers n (n = 1,2,. . .kmax), only layers open to the well are included, and, in summations over categories m (m = 1,2,. . N), only those categories with wells open to layer k are included.

$$\rho_\mu Q^N_{i,j,k} = \frac{2\pi T^*_{i,j,k} h_{f,j,k}}{\ln\left(\frac{r_{a_k}}{r_w}\right)} \sum_{m=1}^{N} u_m \delta[W(k,m)] - \frac{2\pi T^*_{i,j,k}}{\ln\left(\frac{r_{a_k}}{r_w}\right)} \sum_{m=1}^{N} \left\{ u_m \delta[W(k,m)] \frac{\sum_{n=1}^{k_{max}} \left[\frac{T^*_{i,j,n} h_{f,j,n}}{\ln\left(\frac{r_{a_n}}{r_w}\right)} \right] \delta[W(n,m)]}{\sum_{n=1}^{k_{max}} \left[\frac{T^*_{i,j,n}}{\ln\left(\frac{r_{a_n}}{r_w}\right)} \right] \delta[W(n,m)]} \right\}$$

$$+ \frac{T^*_{i,j,k}}{\ln\left(\frac{r_{a_k}}{r_w}\right)} \sum_{m=1}^{N} \left\{ \frac{\rho_\mu Q_{T_m} \delta[W(k,m)]}{\sum_{n=1}^{k_{max}} \left[\frac{T^*_{i,j,n}}{\ln\left(\frac{r_{a_n}}{r_w}\right)} \right] \delta[W(n,m)]} \right\}, \qquad (15)$$

In equation 15, r_{a_k} (effective radius) is the radius of a circle around a well along which the head computed for node i,j,k is assumed to prevail and is given by:

$$r_{a_k} = \frac{\sqrt{\frac{s^2}{N_k}}}{4.81} = \frac{s}{4.81\sqrt{N_k}}. \qquad (16)$$

where:

s = minimum of the finite-difference block dimensions Δx_j, Δy_i, and

N_k = number of wells in block area $\Delta x_j \cdot \Delta y_i$.

As discussed in Mandle and Kontis (in press) estimation of the radius, r_{a_k} (eq. 16), for each of the N_k wells is an oversimplification of actual field conditions. It is, however, a useful assumption when the purpose of a simulation is to approximate the effects of numerous multiaquifer wells on a flow system, as opposed to the detailed analysis of individual wells. The radius term is obtained by specifying that the sum of the assumed areas ($N_k \pi r_{a_k}^2$) surrounding each of the N_k is equivalent to the area ($\pi s^2/4.81^2$) for a single well of effective radius s/4.81 (Prickett, 1967).

The water level in category m wells is given by:

$$h_{fw_m} = \frac{\sum_{n=1}^{k_{max}} \left[\frac{T^*_{i,j,n} h_{f_{i,j,n}}}{\ln\left(\frac{r_{a_n}}{r_w}\right)} \right] \delta[W(n,m)]}{\sum_{n=1}^{k_{max}} \left[\frac{T^*_{i,j,n}}{\ln\left(\frac{r_{a_n}}{r_w}\right)} \right] \delta[W(n,m)]} - \frac{\rho_\mu Q_{T_m}}{2\pi u_m \sum_{n=1}^{k_{max}} \left[\frac{T^*_{i,j,n}}{\ln\left(\frac{r_{a_n}}{r_w}\right)} \right] \delta[W(n,m)]}, \qquad (17)$$

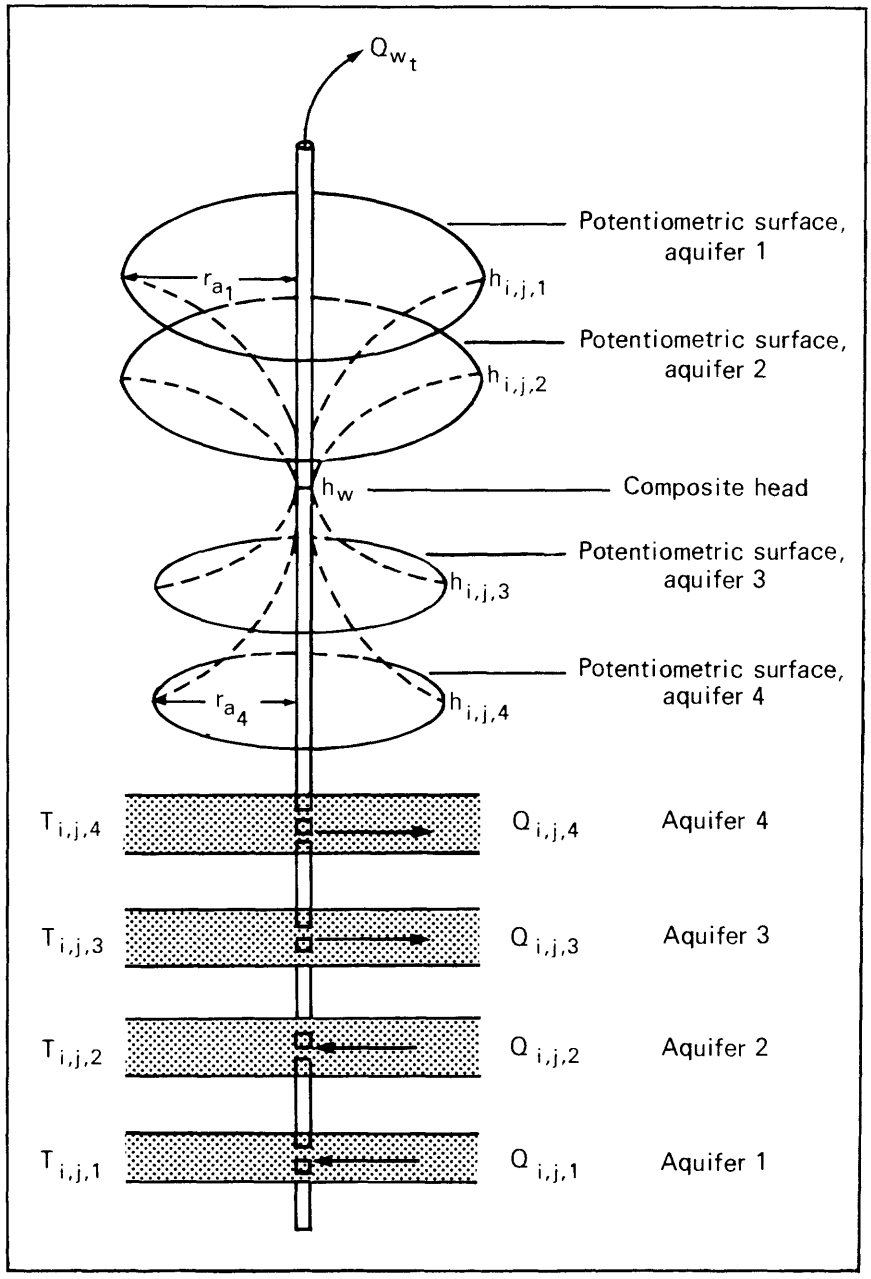


Figure 6.

Conditions of flow and head around a typical multiaquifer well. (From Bennett and others, 1982).

The generalized finite difference form of equation 5a, can be expressed as:

$$L^*(h_{f_{i,j,k}}) - \rho_p Q^N_{i,j,k} = 0,$$

where: $L^*(h_{f_{i,j,k}})$ includes all terms of equation 5a except the product of density and flow to wells, which is now represented by the term $\rho_p Q^N_{i,j,k}$ of equation 15.

The finite-difference formulation of equation 5a allows the pure-water density (ρ_p) to be treated as a variable in space to accomodate areas of highly saline brine, where the concentration of dissolved salts may be so large that the mass of pure water per unit fluid volume is reduced. If most of the wells in a study area produce from zones of relatively fresh water, in which spatial variation of ρ_p is minimal, then a constant ρ_p value of 1.94 slugs/ft^3 may be used in all well-flow terms. For uniform-density applications, equation 15 applies directly if ρ_p is set to unity and T* is set to the product of hydraulic conductivity and layer thickness.

A major advantage of using equation 15 in simulation of systems containing multiaquifer wells is that water will be extracted from each aquifer open to a well in proportion to its relative transmissivity. Otherwise, the contribution from each aquifer to total well discharge must be estimated and altered with each change in model transmissivity. Implementation of equation 15 eliminates the necessity of this step and allows rapid calculation of composite head with equation 17.

In equation 15, the first and second terms consist of various coefficients that multiply the heads of nodes that are linked to each other by the presence of multiaquifer wells. Because these links occur only at nodes above and below a given node, the slice-successive-over-relaxation (SSOR) method was chosen to solve the finite-difference equations to obtain the model heads. The SSOR method readily accommodates the multiaquifer terms because a solution coefficient matrix is constructed for vertical slices along model rows. The nonzero terms of the matrix represent conductances of the nodes within the slice. Consequently, to implement equation 15, the Trescott-Larson program was modified to compute the multiaquifer coefficients of the equation and to insert them in the proper location of the SSOR coefficient matrix. The last term in equation 15 is independent of head and, inserted into the well array of the Trescott-Larson program code, serves as a source/sink term in the finite-difference equation.

Modifications to Trescott-Larson Program

The correspondence between Fortran variables in the modified Trescott-Larson program and terms of the multiaquifer-well formulation (eq. 15) is given in table 4. The indices of the Fortran variables in the table are symbolic and represent: layer number K, (K = 1,2,...KO) for KO layers; node-set number L, (L = 1,2,...NWEL) for NWEL node sets (a node set consists of a vertical stack of nodes at a particular row and column intersection (I,J) for which a well or wells taps at least one aquifer); and well category number N, (N = 1,2,...JA) for JA categories of a given node set.

Table 4.--Correspondence between Fortran variables used in Trescott-Larson model program and terms of multiaquifer-well formulation

Multiaquifer-well term (equation 15)	Fortran variable (equation 18)
kmax	KO
W(k,m)	KLL(K,L,N)
Q_{T_m}	QWL(N,L)
r_{a_k}	REL(K,L)
r_w	RWL(L)
ρ_p	1.94
2π	PI2
$T^*_{i,j,k}$	T(I,J,K)
$h_{f_{i,j,k}}$	PHI(I,J,K)
u_m	NU(N,L)
N	NWL(L)=JA
$1/\ln(r_{a_k}/r_w)$	C1(K,L)
$1 \Big/ \sum_{n=1}^{k_{max}} \left[\frac{T^*_{i,j,n}}{\ln\left(\frac{r_{a_n}}{r_w}\right)} \right] \delta[W(n,m)]$	C2(N,L)
$\frac{T^*_{i,j,k}}{\ln\left(\frac{r_{a_k}}{r_w}\right)} \sum_{m=1}^{N} \left\{ \frac{\rho_p Q_{T_m} \delta[W(k,m)]}{\sum_{n=1}^{k_{max}} \left[\frac{T^*_{i,j,n}}{\ln\left(\frac{r_{a_n}}{r_w}\right)} \right] \delta[W(n,m)]} \right\}$	WELL(I,J,K)

Through the Fortran variable notation given in table 4, the multiaquifer-well formulation (eq. 15) for node (I,J,K) corresponding to node set L may be written as:

$$\rho_p Q(I,J,K)^N = \sum_{N=1}^{JA} NU(N,L)\cdot\delta[KLL(K,L,N)]\{PI2\cdot C1(K,L)\cdot T^*(I,J,K)\cdot PHI(I,J,K) - PI2\cdot C1(K,L)\cdot T^*(I,J,K)\sum_{M=1}^{KO} PHI(I,J,M)\cdot T^*(I,J,M)\cdot C1(M,L)\cdot C2(N,L)\cdot\delta[KLL(M,L,N)]\} + WELL(I,J,K). \quad (18)$$

Equation 18 is the working formula for the multiaquifer-well terms that was incorporated in the Trescott model. The input data required by equation 18 are read in subroutine DATAI near the location where pumping rates were formerly read. A multiaquifer-well record contains the following information for each node set L:

IL(L) = row number,
JL(L) = column number,
NWL(L) = JA = total number of categories of multiaquifer wells,
KLL(K,L,N) = A KO digit code for each category N (N = 1,2...JA) of wells containing the layer number K if the layer is open to a well and zero otherwise, and
QWL(N,L) = total pumping rate, in million gallons per day, of all wells in category N.

For example, table 5 shows the input record for a node set L at model row 20 and column 27 that contains three categories of wells (NWL). The first category is open to layers 1, 2, 3, and 4; the second to layers 1 and 2; and the third to layer 1 only. The pumping rates for each category are 2, 1 and 0.5 million gallon per day, respectively. The multiaquifer-well input file consists of a record similar to the table below for each node set and is ordered by column within each row starting with the topmost row.

Table 5.--Example of multiaquifer-well input record for node set L at row 20, column 27.

INPUT-VARIABLE NAME								
IL(L)	JL(L)	NWL(L)	KLL(K,L,1)	KLL(K,L,2)	KLL(K,L,3)	QWL(1,L)	QWL(2,L)	QWL(3,L)
20	27	3	1234	1200	1000	2.0	1.0	0.5

The multiaquifer well formulation was incorporated in the Trescott-Larson model as an additional option so that the program can be executed in the conventional manner if the multiaquifer-well option is not desired. The following routines of the program were changed:

(1) MAIN program and subroutine PRNTA to include the new multiaquifer-well option,

(2) subroutine DATAI to read the multiaquifer-well input data (table 5) and to compute some of the terms of table 4,

(3) subroutine CHECKI to compute and output composite heads, and

(4) subroutine SOLVE (SSOR solution method) to call two new subroutines, MULT and RESI.

The new subroutine MULT computes modified and additional coefficients resulting from the presence of multiaquifer wells and places them in the proper location of the SSOR solution coefficient matrix. Subroutine RESI computes the multiaquifer-well contributions for the "right-hand" side of the solution matrix equation. In addition, a third new subroutine ALOCAT computes all flows into and out of each node of a multiaquifer node set and a balance for each node.

A brief discussion of multiaquifer-well modifications of the Trescott-Larson routines follows. A discussion of the SSOR method (subroutines SOLVE and BAND), and a discussion of the new subroutines (MULT, RESI, and ALOCAT) is given in appendix A; and a listing of these routines is given in appendix F and G. A listing of the multiaquifer-well modifications is given in appendix E; the statement numbers of appendix E follow the numbering system of the original Trescott documentation (Trescott, 1975).

Program MAIN.--Modifications, given in appendix E, consist of adding a common block termed WIS containing the multiaquifer-well input variables of table 4, dimensioning new variables with a parameter statement, adding two new options (IOPEN and IALOCA) to the model options list, and specifying the value of PI2. In addition, several format changes and changes to the call to entry ARRAY of subroutine DATAI to specify a logical unit number (LIN) for input data were made. To invoke the multiaquifer-well option and subroutine ALOCAT, the words "OPEN" and "ALOC" are specified in the 12th and 13th fields of the Trescott-Larson options record.

Subroutine DATAI.--Modifications, given in appendix E, consist of including the new options (IOPEN and IALOCA), common-block WIS, the logical unit number LIN, code to read the multiaquifer-well records (as in table 5), and statements to compute or specify the terms REL, RWL, NU, C1, C2 and WELL of table 4. Because the northern Midwest pumping-rate file contains positive values for discharge (table 5), the pumping rate input (QWL), in million gallons per day, is multiplied by -1 and converted to cubic feet per second within the subroutine. If a simulation includes the variable water-density option, the pumping rates are multiplied by the density of pure water (1.94 slugs/ft^3) to convert to units of mass/time required by the variable water-density formulation. The number of wells (NU) in a given category was estimated by dividing the pumping rate (QWL) by an assumed pumping rate per well of 0.1 ft^3/s, and the well radius (RWL) was taken to be 0.5 ft.

The effective radius (r_{a_k} or REL of table 4) of wells tapping layer k is estimated in the subroutine as follows:

From equation 16,

$$r_{a_k} = \frac{s}{4.81\sqrt{\frac{1}{N_k}}}$$

N_k, the total number of wells open to layer K for a particular node set L, is computed by

$$N_k = NR(K,L) = \sum_{JJ=1}^{JA} NU(JJ,L) \cdot \delta[KLL(K,L,JJ)],$$

where NU and JA are as previously defined (table 4). Consequently,

$$r_{a_k} = REL(K,L) = \sqrt{\frac{s^2}{4.81^2 NR(K,L)}} = \sqrt{\frac{0.04322 s^2}{NR(K,L)}}.$$

The multiaquifer-well coefficients are not particularly sensitive to variations in the value of r_{a_k} and r_w because the effective and well radii appear in equation 15 as the natural logarithm of their ratio (r_{a_k}/rw).

Subroutine CHECKI.--Modifications, given in appendix E, were made to include the new options (IOPEN and IALOCA) and common block WIS, dimension arrays HW and HZ used to compute composite head; output the composite head and call subroutine ALOCAT if the IALOCA option is invoked. In subroutine CHECKI, HW is the composite head given by equation 17, whereas HZ is the composite head with no pumping ($Q_{T_m} = 0$) and represents model results that may be compared with measured data from observation wells. For the latter case, the output consists of records similar in structure to the multiaquifer-well input records (table 5). For each node set the record contains the row and column, number of multiaquifer-well categories, the layer code, and the observation-well composite-head for each category. These records are written to Fortran unit 13.

Subroutine PRNTA.--Modifications, given in appendix E, consist of adding the new options IOPEN and IALOCA.

SUMMARY

The ground-water flow equation used in the Northern Midwest Regional Aquifer-System Analysis was modified to simulate the effects of time invariant, spatially varying ground-water density, and wells open to more than a single aquifer. Corresponding modifications of the Trescott-Larson three-dimensional ground-water flow program to allow simulation of these effects are presented. In addition, the functioning of the slice-successive-overrelaxation method to solve for head distribution is summarized and illustrated. Two new programs are presented: program STEAM computes ground-water density, pure-water density, and dynamic viscosity; program VARDEN computes model-conductance and variable-density input data for the Trescott-Larson model.

The following steps are required to simulate variable-density effects with the modified Trescott-Larson model (appendix D):

(1) For each model layer, execute program STEAM (appendix C) with the input data given in table 3 to generate arrays of ground-water density, pure-water density, and dynamic viscosity.

(2) From the output of program STEAM (table 3) and the additional input given in table 1, execute program VARDEN (appendix B) to generate input data for the Trescott-Larson model (table 1) for the terms of equation 5a and individual variable-density terms (table 1) required to compute flow (eq. 6).

(3) Place model starting heads and output arrays of program VARDEN (table 1) in a single file (Fortran unit-number LIN of appendix D).

(4) Set model grid spacing (DELX and DELY) to unity, specify the IQRE option of Trescott-Larson model and execute the model in the conventional manner (as given in Trescott, 1975) to generate freshwater head or pressure.

The following procedures are required to simulate the effects of multi-aquifer wells with the modified Trescott-Larson (appendix E) and SSOR subroutines (appendix F), and the new subroutines MULT and RESI (appendix G).

(1) If variable density is not being simulated or if model grid spacing is nonuniform modify statements DAT1790 and Dat1797 according to comments in subroutine DATAI (statements DAT1786 and DAT1793 of appendix E) pertaining to multiplication of pumping rates by density of pure water and computation of area of block (AREA).

(2) Create the multiaquifer pumping-rate file to incorporate records similar to those in table 5 for each model node set containing a well(s). Records are ordered columnwise starting at top row.

(3) Specify an output file (Fortran unit 13) to save resultant multiaquifer composite heads (eq. 17).

(4) Invoke IOPEN option in the 12th field of the Trescott-Larson options record and execute the Trescott-Larson model in the conventional manner (as given in Trescott, 1975).

Files of programs discussed in this report are available from the authors upon request.

REFERENCES CITED

Bennett, G. D., 1979, Regional ground-water systems analyses: Water Spectrum, v. 11, no. 4, p. 36-42.

_____ 1980, Research and model development related to the Regional Aquifer System Analysis Program: American Geophysical Union Transactions, EOS, v. 61, no. 46, p. 951.

Bennett, G. D., and Giusti, E. V., 1971, Coastal ground-water flow near Ponce, Puerto Rico: U.S. Geological Survey Professional paper 750-D, p. D200-D211.

Bennett, G. D., Kontis, A. L., and Larson, S. P., 1982, Representation of multiaquifer-well effects in three-dimensional ground-water flow simulation: Ground Water, v. 20, no. 3, p. 334-341.

Bond, D. C., 1972, Hydrodynamics in deep aquifers of the Illinois basin: Illinois State Geological Survey Circular 470, 72 p.

Davis, T. M., and Kontis, A. L., 1970, Spline interpolatlion algorithms for track-type survey data with application to the computation of mean gravity anomalies: U.S. Naval Oceanographic Office Technical Report TR-226, 50 p.

Kuiper, L. K., 1983, A numerical procedure for the solution of the steady state variable density groundwater flow equation: Water Resources Research, v. 19, no. 1, p. 234-240.

_____ 1985, Documentation of a numerical code for the simulation of variable density ground-water flow in three dimensions: U.S. Geological Survey Water-Resources Investigations Report 84-4302, 98 p.

Matthews, G. S., and Russell, D. G., 1967, Pressure buildup and flow tests in wells, appendix G: Dallas, Texas, Society of Petroleum Engineers of the American Institute of Mining, Metallurgical and Petroleum Engineers Monograph HI, p. 158.

Mandle, R. J., and Kontis, A. L., Simulation of regional ground-water flow in the Cambrian-Ordovician aquifer system in the northern Midwest, United States: U.S. Geological Survey Professional Paper 1405-C (in press).

Potter, R. W., and Brown, D. L., 1977, The volumetric properties of aqueous sodium chloride solutions from 0° to 500°C at pressures up to 2,000 bars based on a regression of available data in literature: U. S. Geological Survey Bulletin 1421-C, 36 p.

Prickett, T. A., 1967, Designing pumped well characteristics into electric analog models: Groundwater, v. 5, P.O. Box 1669. 38-46.

Steinhilber, W. L., and Young, H. L., 1979, Plan of study for the Northern Midwest Regional Aquifer-System Analysis: U.S. Geological Survey Water-Resources Investigations Report 79-44, 20 p.

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., 1976, Documentation of finite-difference model for simulation of three-dimensional ground-water flow, supplement to Open-File Report 75-438: U.S. Geological Survey Open-File Report 76-591, 21 p.

Weiss, E. J., 1982a, A model for the simulation of flow of variable density ground water in three dimensions under steady-state conditions: U.S. Geological Survey Open-File Report 82-352, 60 p.

_____ 1982b, A computer program for calculating relative-transmissivity arrays to aid in model calibration: U.S. Geological Survey Open-File Report 82-447, 18 p.

_____ 1985, Evaluating the hydraulic effects of changes in aquifer elevation using curvilinear coordinates: Journal of Hydrology, v. 81, p. 253-275.

Young, H. L., Siegel, D. I., Mandle, R. J., and Kontis, A. L., 1986, Northern Midwest regional aquifer-system study, in Sun, R. J., Regional aquifer system analysis program of the U.S. Geological Survey, summary of projects, 1978-84: U.S. Geological Survey Circular 1002, p. 72-87.

APPENDIX A.--SLICE-SUCCESSIVE-OVER-RELAXATION (SSOR) SOLUTION METHOD

A narrative of the major elements of the SSOR procedure used in this study follows, and a listing of subroutine SOLVE, which implements the SSOR method, and subroutine BAND, which solves the set of resultant linear equations by Gaussian elimination, is given in appendix F. The description of the SSOR method is presented to illustrate how the Trescott-Larson model was modified to incorporate the effects of multiaquifer wells. In subroutine SOLVE (appendix F), original program statements as written by S. P. Larson and C. R. Faust (U.S. Geological Survey written commun., 1979), multiaquifer-well statements, and comments describing the functioning of the subroutine are indicated by "SP3," asterisks, and zeros in columns 73-75, respectively. Subroutine SOLVE contains two entry points--entry ITER and entry NEWITA. Entry ITER, entered once per simulation, contains statements to (1) identify active model nodes, (2) number nodes, and (3) order and count the solution equations for each slice and for the entire model. Entry NEWITA, entered each time step, sets up matrices required to obtain solutions for head and calls subroutine BAND to implement the solution.

Entry ITER

In the SSOR routine, model nodes are numbered according to a one-dimensional numbering convention. The convention is illustrated for a single slice through a cross-sectional model consisting of 3 rows, 5 columns, and 5 layers in figure A-1. The node numbering begins at the upper left corner of the bottom layer (K = 1) and proceeds in sequence along model rows. In Entry ITER, given in appendix F, the node number is designated by the variable N. Of the 75 model nodes in the example (fig. A-1), 15 are active (circled) because the border nodes of each layer are inactive. The number of active nodes and, therefore, the number of equations for each slice are counted and placed in array NB. For the example (fig. A-1), NB(I) for I = 2 is 15. The equations for each slice are ordered in array NP by indexing on layer (K) and then on column (J). If a node is inactive the corresponding value of NP is zero. Table A-1 illustrates the equation ordering system for the 15 active nodes of the slice through I = 2. In addition to numbering nodes and ordering and counting equations for each slice, the maximum half-band width (MBW) of the solution coefficient matrix is computed (MBW = number of layers +1). The SSOR acceleration parameter (WO), usually chosen to be some value between 1 and 2, and the number of iterations between application of the so-called Watt's correction (LENGTH), are read in Entry ITER.

Entry NEWITA

For the conventional finite-difference computation stencil shown in figure A-2A (p.38), the finite-difference equation of node i,j,k at the start of a time step and with no multiaquifer well is:

$$Eh_{i,j,k} + Dh_{i,j-1,k} + Fh_{i,j+1,k} + Bh_{i-1,j,k} + Hh_{i+1,j,k} + Zh_{i,j,k-1} + SUh_{i,j,k+1} = \frac{RHO}{\Delta t} h^*_{i,j,k} + Q_{i,j,k}, \qquad \text{(A-1)}$$

Table A-1

Slice-successive-over-relaxation equation numbering scheme for cross-sectional model on right.

Node number (N)	Node (I,J,K)	Equation number (NP)
5	2,2,1	1
20	2,2,2	2
35	2,2,3	3
50	2,2,4	4
65	2,2,5	5
8	2,3,1	6
23	2,3,2	7
38	2,3,3	8
53	2,3,4	9
68	2,3,5	10
11	2,4,1	11
26	2,4,2	12
41	2,4,3	13
56	2,4,4	14
71	2,4,5	15

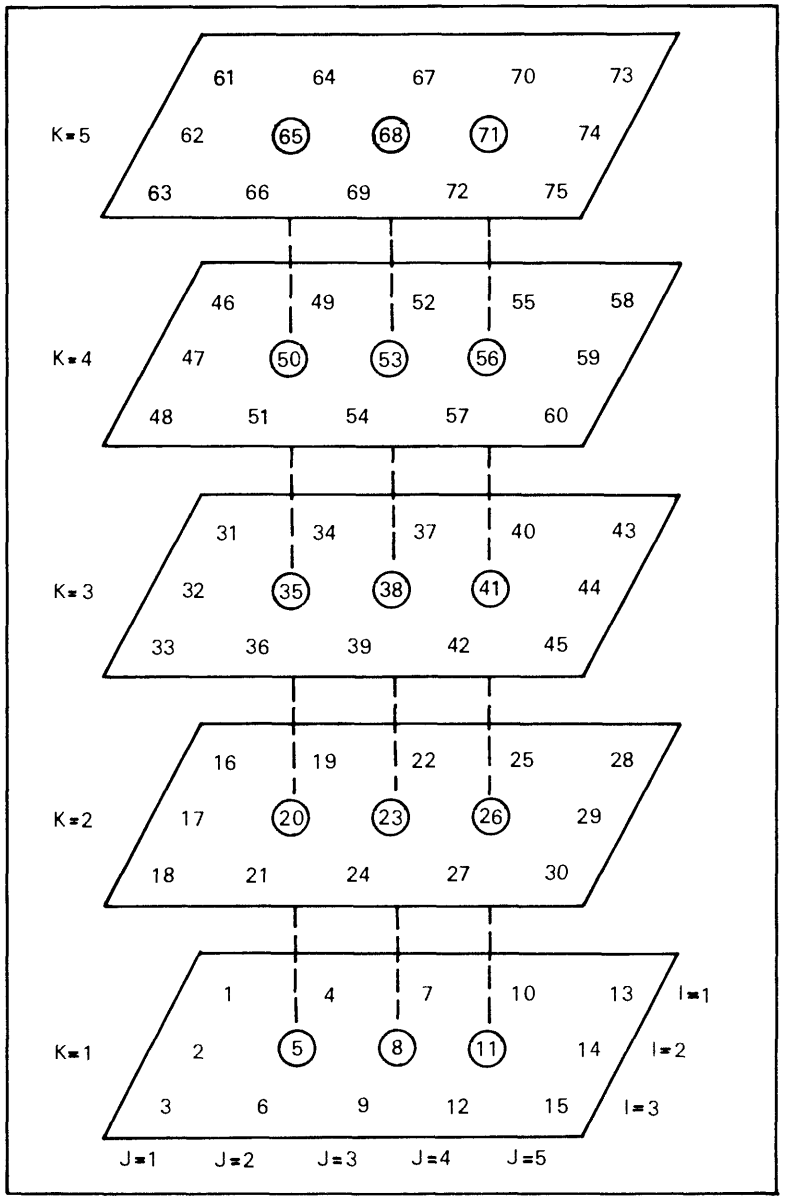


Figure A-1.

Example of slice-successive-over-relaxation node numbering scheme (values of N) for cross-sectional model consisting of 3 rows, 5 columns, and 5 layers.

where:

coefficients D, F, B, H, Z, and Su are the conductances between node i,j,k and the six adjacent nodes (fig. A-2),

RHO/Δt is the storage coefficient times the area of the block i,j,k divided by the length of the time step,

Q represents sources and sinks, and

h* is the head at the beginning of the time step.

The coefficient of the head at i,j,k (E) is the sum of all conductances plus RHO/Δt.

In the SSOR routine, the solution is posed in terms of head differences rather than absolute head so that if $\Delta h_{i,j,k}$ is defined to be the difference in head between time steps τ+1 and τ, the finite-difference equation for node i,j,k at time step τ+1 may be written as

$$E\Delta h^{\tau+1}_{i,j,k} + D\Delta h^{\tau+1}_{i,j-1,k} + F\Delta h^{\tau+1}_{i,j+1,k} + B\Delta h^{\tau+1}_{i-1,j,k} + H\Delta h^{\tau+1}_{i+1,j,k} + Z\Delta h^{\tau+1}_{i,j,k-1} + SU\Delta h^{\tau+1}_{i,j,k+1}$$

$$= Q_{i,j,k} - Eh^{\tau}_{i,j,k} - Dh^{\tau}_{i,j-1,k} - Fh^{\tau}_{i,j+1,k} - Bh^{\tau}_{i-1,j,k} - Hh^{\tau}_{i+1,j,k} - Zh^{\tau}_{i,j,k-1} - SUh^{\tau}_{i,j,k+1} - \frac{RHO}{\Delta t} h^{*}_{i,j,k}$$

$$= RHS^{\tau}_{i,j,k}. \qquad \text{(A-2)}$$

The right hand side of equation A-2 consists of the product of coefficients and heads of the previous time step (τ) plus the source/sink terms and is called $RHS_{i,j,k}$. Equation A-2 is implemented according to the matrix relation

$$[A_s]^{\tau}\,[\Delta h]^{\tau+1} = [RHS]^{\tau}, \qquad \text{(A-3)}$$

Where:

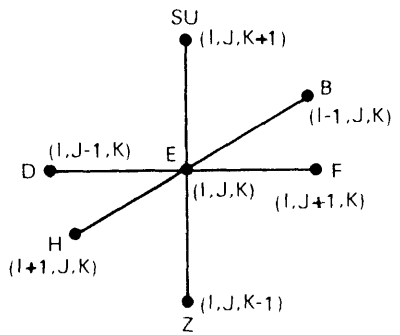
$[A_S]$ (array 'A') = symmetric compressed coefficient matrix of dimension NEQ x MBW for NEQ active model nodes and half-bandwith MBW,

[Δh] = vector of head differences of dimension NEQ, and

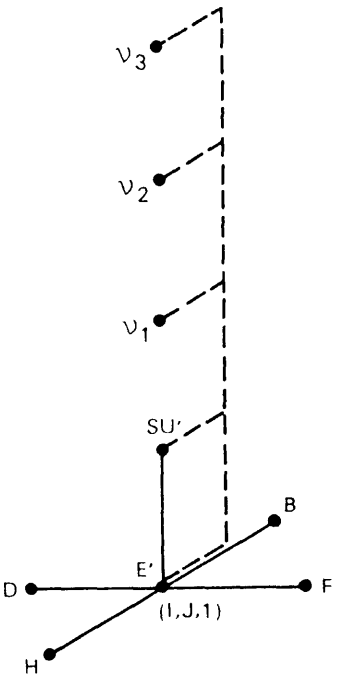
[RHS] = vector of right-hand side terms of equation A-2 and is of dimension NEQ.

For the example (fig. A-1), MBW = 6 and NEQ = 15.

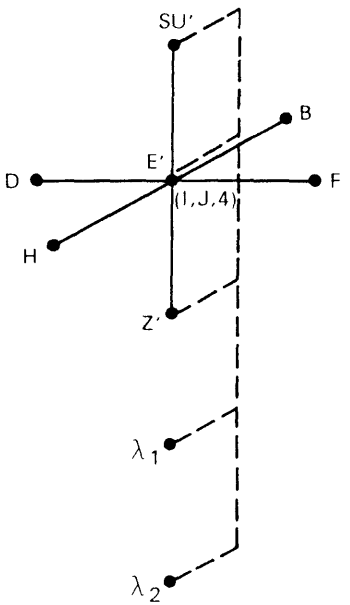
In entry NEWITA, given in appendix F, the coefficients for the 'A' array and the RHS term are built according to the numbering scheme of table A-1 for all nodes containing a nonzero value in array NP. These terms are developed for each slice, once per time step, in statements SP30780 through SP31300 of subroutine SOLVE (appendix F).



A. Conventional finite-difference stencil at node i,j,k



B. Finite-difference stencil at node i,j,1 and multi-aquifer-well coefficients for well open to five consecutive model layers



C. Finite-difference stencil at node i,j,4 and multi-aquifer-well coefficients for well open to five consecutive model layers

Figure A-2. Conventional finite-difference computation stencil and example of multiaquifer-well stencil for five-layer case positioned at nodes in layers 1 and 4.

The solution for head differences at time step τ+1 is obtained by way of an iteration process. Because the SSOR procedure solves for the head differences within a slice, terms involving head differences for slice i-1 ($B\Delta h_{i-1,j,k}$) and slice i+1 ($H\Delta h_{i+1,j,k}$) (fig. A-2A) may be moved to the right side of the equation. Thus, for iteration m+1 of time step τ+1 the equation for node i,j,k is:

$$E\Delta h^{m+1}_{i,j,k} + D\Delta h^{m+1}_{i,j-1,k} + F\Delta h^{m+1}_{i,j+1,k} + Z\Delta h^{m+1}_{i,j,k-1} + SU\Delta h^{m+1}_{i,j,k+1} =$$

$$RHS^{\tau}_{i,j,k} - (B\Delta h^{m+1}_{i-1,j,k} + H\Delta h^{m}_{i+1,j,k}) = RSMAL^{m+1}_{i,j,k}\,, \qquad \text{(A-4)}$$

where $RSMAL_{i,j,k}$ is the new right side of the equation incorporating the RHS term of equation A-2 and the coefficient terms involving B and H. Because the head differences for slice i-1 are available at iteration m + 1, they are used in the development of RSMAL. In matrix form, the solution equation for time step τ+1 and iteration m + 1 is

$$[A_s]^{\tau}\,[\Delta h]^{\tau+1}_{m+1} = [RSMAL]_{m+1} = [RHS]^{\tau} - [B\Delta h]_{m+1} - [H\Delta h]_{m}\,. \qquad \text{(A-5)}$$

Application of equation A-4 and the nature of the compressed coefficient matrix A_S (array 'A') for the cross sectional slice through I = 2 is illustrated in figure A-3A. In the figure, all elements having the same symbol will, in general, have different values. If the coefficient matrix were written in its full form, it would be of size NEQ x NEQ, and the E coefficient associated with node i,j,k would lie along the main diagonal of the full matrix. Because the SU conductance of node i,j,k is equivalent to the Z conductance of node i,j,k+1, and the F conductance of node i,j,k is equivalent to the D conductance of node i,j+1,k, the full coefficient matrix is symmetric so that only coefficients for the upper diagonal part (E,SU and F) are required.

This symmetry allows development of the matrix in compressed form such that the main diagonal coefficients are located in the first column of each matrix row (fig. A-3A). The vector of head differences [Δh], contains all the active nodes of the model and is ordered as to table A-1. Multiplication of components of [Δh] by the compressed coefficient matrix according to the relation

$$\sum_{i=1}^{MBW} A_{j,i}\Delta h_{j+i-1} = RSMAL_j\,, \quad j=1,2...NEQS$$

generates the set of equations to be solved for head difference for each iteration.

For each iteration, the RSMAL term (eq. A-4) is updated and subroutine BAND is entered to solve for an interim set of head differences, for each slice, by Gaussian elimination. The final head differences for iteration m + 1 are obtained by applying an over-relaxation factor (acceleration parameter WO) to speed the rate of convergence--that is, for node i,j,k

$$\Delta h^{m+1}_{i,j,k} = \Delta h^{m}_{i,j,k} + wo(\Delta h^{*m+1}_{i,j,k} - \Delta h^{m}_{i,j,k}), \qquad \text{(A-6)}$$

where:

$\Delta h^*_{i,j,k}$ = interim head-difference solution, and

Δh^m = solution from the previous iteration.

In subroutine SOLVE (appendix F), the iteration loop extends from statement SP31310 to SP32330. The section where the RSMAL terms are developed for each slice and where the interim solutions for head differences within the slice are obtained extends from statement SP31400 through SP31570. Application of the over-relaxation factor to obtain the head difference solutions extends from SP31580 to SP31690.

After iteration m + 1 is completed for all slices, the maximum change in the head difference for all nodes, relative to the previous iteration, is compared with the specified closure criteria (ERR). If closure is achieved, final heads for the time step are formed by adding the head differences to the heads of the previous time step. This is performed from statements SP32350 to SP32400 (appendix F).

If closure is not achieved, a process termed the Watts correction, designed to speed the convergence, may be applied. In this process, a correction factor is computed for each slice, such that when the factor is added uniformly to the head-difference solution, the sum of the residuals over each slice is reduced to zero. If, however, a slice contains at least one constant-head node, the correction factor is zero, so that nothing is gained by applying the process to the slice. The Watts correction was not implemented for the northern Midwest study because the top layer consists of constant-head nodes.

Modifications of SSOR for Effects of Multiaquifer Wells

The multiaquifer-well modifications of the SSOR procedure in subroutine SOLVE consist of determining the set of coefficients as prescribed by equations 15 or 18 and placing them in their proper location in the solution coefficient matrix (array 'A') and the right-hand side vector RHS of equation A-2. No changes are required in the development of the RSMAL vector (eq. A-4) because the presence of multiaquifer wells does not alter the B and H coefficients.

The coefficients in question are those of equation 15 or 18 that multiply the heads of layers open to a well. For example, the multiaquifer-well stencil-position for a node set at layer 1 of a five-layer model where all layers are open to a well is shown in figure A-2B. The coefficients associated with layers 1 and 2 will consist of the E and SU terms of the standard stencil (fig. A-2A) plus a contribution from equation 15 or 18 forming coefficients E' and SU' (fig. A-2B). As shown in figure A-2B, new coefficients associated with layers 3, 4, and 5 (ν_1, ν_2, and ν_3) are required. The set of coefficients for the computation stencil at a node in layer 4 is shown in figure A-2C. Because the multiaquifer-well coefficients are symmetric, the Z', λ_1 and λ_2 terms need not be determined explicitly. In the example, the Z', λ_1 and λ_2 coefficients of figure A-2C are equivalent to the SU', ν_1 and ν_2 coefficients when the com-

' A' ARRAY							ΔH(N),(I,J,K)		RSMAL(N)
E	SU	0	0	0	F		ΔH(5),(2,2,1)		RSMAL(5)
E	SU	0	0	0	F		ΔH(20),(2,2,2)		RSMAL(20)
E	SU	0	0	0	F		ΔH(35),(2,2,3)		RSMAL(35)
E	SU	0	0	0	F		ΔH(50),(2,2,4)		RSMAL(50)
E	0	0	0	0	F		ΔH(65),(2,2,5)		RSMAL(65)
E	SU	0	0	0	F		ΔH(8),(2,3,1)		RSMAL(8)
E	SU	0	0	0	F		ΔH(23),(2,3,2)		RSMAL(23)
E	SU	0	0	0	F	×	ΔH(38),(2,3,3)	=	RSMAL(38)
E	SU	0	0	0	F		ΔH(53),(2,3,4)		RSMAL(53)
E	0	0	0	0	F		ΔH(68),(2,3,5)		RSMAL(68)
E	SU	0	0	0	0		ΔH(11),(2,4,1)		RSMAL(11)
E	SU	0	0	0	0		ΔH(26),(2,4,2)		RSMAL(26)
E	SU	0	0	0	0		ΔH(41),(2,4,3)		RSMAL(41)
E	SU	0	0	0	0		ΔH(56),(2,4,4)		RSMAL(56)
E	0	0	0	0	0		ΔH(71),(2,4,5)		RSMAL(71)

(A) SSOR SOLUTION MATRICES -- NO MULTIAQUIFER WELLS.

' A' ARRAY							ΔH(N),(I,J,K)		RSMAL(N)
E'	SU'	ν_1	ν_2	ν_3	F		ΔH(5),(2,2,1)		RSMAL(5)
E'	SU'	ν_1	ν_2	0	F		ΔH(20),(2,2,2)		RSMAL(20)
E'	SU'	ν_1	0	0	F		ΔH(35),(2,2,3)		RSMAL(35)
E'	SU'	0	0	0	F		ΔH(50),(2,2,4)		RSMAL(50)
E'	0	0	0	0	F		ΔH(65),(2,2,5)		RSMAL(65)
E	SU	0	0	0	F		ΔH(8),(2,3,1)		RSMAL(8)
E	SU	0	0	0	F		ΔH(23),(2,3,2)		RSMAL(23)
E	SU	0	0	0	F	×	ΔH(38),(2,3,3)	=	RSMAL(38)
E	SU	0	0	0	F		ΔH(53),(2,3,4)		RSMAL(53)
E	0	0	0	0	F		ΔH(68),(2,3,5)		RSMAL(68)
E	SU	0	0	0	0		ΔH(11),(2,4,1)		RSMAL(11)
E	SU	0	0	0	0		ΔH(26),(2,4,2)		RSMAL(26)
E	SU	0	0	0	0		ΔH(41),(2,4,3)		RSMAL(41)
E	SU	0	0	0	0		ΔH(56),(2,4,4)		RSMAL(56)
E	0	0	0	0	0		ΔH(71),(2,4,5)		RSMAL(71)

(B) SSOR SOLUTION MATRICES -- WITH WELL AT NODE SET I=2,J=2 AND OPEN TO ALL 5 LAYERS.

Figure A-3. Example of slice-successive-over-relaxation solution matrices without (A) and with (B) multiaquifer-well coefficients for cross-sectional model consisting of 3 rows, 5 columns, and 5 layers.

putation stencil is positioned at layers 3, 2, and 1, respectively. For the cross-sectional example of figure A-1, the compressed coefficient matrix (array 'A') would have the structure shown in figure A-3B if a well is located in node set (I=2,J=2) and open to all layers.

The computation and loading of multiaquifer-well coefficients for array 'A' are performed in subroutine MULT (appendix G), which is called from subroutine SOLVE prior to development of the standard 'A' array. In subroutine MULT, the coefficients are developed for the entire model before returning to subroutine SOLVE. Because of the symmetry discussed above and illustrated in figures A-2B and C, only those coefficients above any particular location of the computation stencil needs to be explicitly computed. As the E and SU coefficients are computed for the conventional stencil in subroutine SOLVE, the corresponding nonzero elements of the 'A' array generated by subroutine MULT are added to the E and SU terms to form the E' and SU' coefficients.

The multiaquifer-well terms required for the RHS vector (eqs. A-2 and A-3) consisting of coefficients of layers open to a well times the corresponding head prevailing at the end of the previous time step is computed in subroutine RESI (appendix G). This subroutine is called from subroutine SOLVE each time the conventional RHS term is computed. The multiaquifer-well contribution termed (RESO) is added to the RHS after each return to subroutine SOLVE.

Sections of subroutine SOLVE (appendix F) that contain modifications for the multiaquifer-well effects and that call subroutines MULT and RESI are indicated by asterisks in columns 73-75.

Subroutine ALOCAT (appendix G) computes all individual flow components of each node of a multiaquifer node set, including layer flow to the wells. This subroutine is not required for the simulation of multiaquifer-well effects and was written primarily to aid in debugging the multiaquifer-well modification of the SSOR procedure. If the IALOCA option is invoked, subroutine ALOCAT is called from subroutine CHECKI (statement CHKI178).

APPENDIX B.--PROGRAM VARDEN

```
C ***************************************************************VAR0010
C *                          PROGRAM VARDEN                       *VAR0020
C *                          --------------                       *VAR0030
C * PROGRAM TO COMPUTE CONDUCTANCES AND SOURCE TERMS FOR SIMULATION OF *VAR0040
C * STEADY STATE VARIABLE DENSITY EFFECTS WITH TRESCOTT 3-D FLOW MODEL.*VAR0050
C * PROGRAM IS BASED ON R.J.SUN QUASI-3D DEVELOPMENT (WRITTEN COMMUN., *VAR0060
C * 1981) OF G.D.BENNETT VARIABLE-DENSITY FORMULATION (WRITTEN COMMUN. *VAR0070
C * ,1979).                                                          *VAR0080
C * THE GENERAL STRUCTURE OF THE PROGRAM IS PATTERNED IN PART AFTER   *VAR0090
C * PROGRAM WRITTEN BY E.WEISS,1982,(' A MODEL FOR THE SIMULATION OF  *VAR0100
C * FLOW OF VARIABLE-DENSITY GROUND WATER IN THREE DIMENSIONS UNDER   *VAR0110
C * STEADY STATE CONDITIONS',USGS OPEN FILE REPORT 82-352).          *VAR0120
C * THE PROGRAM MAY BE EXECUTED TO GENERATE DATA FOR TRESCOTT MODEL IN *VAR0130
C * TERMS OF FRESH WATER HEADS OR IN TERMS OF PRESSURE. OUTPUT CONSISTS*VAR0140
C * OF ARRAYS WHICH SERVE AS INPUT TO TRESCOTT MODEL AND ARRAYS REQUIR-*VAR0150
C * ED TO COMPUTE FLOW DIRECTION AND MAGNITUDE.                      *VAR0160
C *                              INPUT                               *VAR0170
C *                              -----                               *VAR0180
C * INPUT CONSISTS OF A SINGLE PROGRAM CONTROL RECORD (ON UNIT 5) CON- *VAR0190
C * TAINING THE NUMBER OF MODEL ROWS(IO),COLUMNS(JO), LAYERS(KO); THE  *VAR0200
C * LOGICAL UNIT NUMBER OF INPUT ARRAYS(LIN); AND 2 OPTION CODES ----- *VAR0210
C * (IPRESS AND IDELTA). FOLLOWING THIS RECORD THE PROGRAM READS A SER-*VAR0220
C * IES OF INPUT ARRAYS AND THE MODEL GRID SPACING.                  *VAR0230
C * FOR FRESH WATER HEAD SIMULATION IPRESS=0 AND FOR PRESSURE SIMULA-  *VAR0240
C * TION IPRESS=1.                                                   *VAR0250
C * THE PROGRAM HAS CODE TO MODIFY TRANSMISSIVITY OF EACH LAYER ,AND   *VAR0260
C * THE VERTICAL LEAKANCE BETWEEN LAYERS BY ARRAYS OF MULTIPLICATIVE   *VAR0270
C * FACTORS (DELTA ARRAYS). INPUT TRANSMISSIVITY AND VERTICAL LEAKANCE *VAR0280
C * ARE MULTIPLIED BY THE DELTA ARRAYS IF THE OPTION IS INVOKED.      *VAR0290
C * IF MODIFICATION OF VERTICAL LEAKANCE IS DESIRED IDELTA=1 AND IF    *VAR0300
C * NOT, IDELTA=0.                                                   *VAR0310
C * TO BYPASS MODIFICATION OF TRANSMISSIVITY AND VERTICAL LEAKANCE SET *VAR0320
C * DELTA ARRAYS TO UNITY AND SET IDELTA=0.                          *VAR0330
C *                                                                  *VAR0340
C * INPUT ARRAYS READ FROM UNIT LIN CONSIST OF STORAGE COEFFICIENTS    *VAR0350
C *        , TRANSMISSIVITY, NODE ELEVATIONS, GROUND WATER DENSITY,    *VAR0360
C * PURE WATER DENSITY, VERTICAL LEAKANCE, CONFINING LAYER THICKNESS,  *VAR0370
C * VISCOSITY, AND DELTA FACTORS. ALL ARRAYS ARE READ BY SUBROUTINE    *VAR0380
C * READS SIMILAR TO THE ARRAY INPUT PROCEEDURE OF THE TRESCOTT MODEL. *VAR0390
C * (SEE COMMENTS IN SUBROUTINE READS).                              *VAR0400
C * IT IS TO BE NOTED THAT THE (X,Y) ORIGIN OF THE VARDEN PROGRAM IS   *VAR0410
C * AT THE LOWER LEFT CORNER OF EACH LAYER. CONSEQUENTLY INPUT ARRAYS  *VAR0420
C * ARE PRESUMED TO RUN FROM BOTTOM ROW TO TOP. LIKEWISE OUTPUT ARRAYS *VAR0430
C * ARE WRITTEN FROM BOTTOM ROW TO TOP. IN THE PROGRAM THE (I+1) INDEX *VAR0440
C * CORRESPONDS TO THE (I-1) INDEX OF THE VARIABLE DENSITY FORMULATION.*VAR0450
C * TO CONFORM TO THE TRESCOTT MODEL ROW CONVENTION, MODEL INPUT ARRAYS*VAR0460
C * ARE READ IN REVERSE ROW ORDER IN THE TRESCOTT MODEL.             *VAR0470
C * FORMATS OF INPUT DATA AND PRINT OF INPUT DATA ARE CONTROLLED BY    *VAR0480
C * INFT AND IOFT ARRAYS PASSED INTO SUBROUTINE READS. SEE DESCRIPTION *VAR0490
C * OF INFT AND IOFT ARRAYS IN DATA STATEMENT.                       *VAR0500
C * IF THE ARGUMENT OF INFT IS (1,1) THE INPUT FORMAT IS 10F8.0; IF THE*VAR0501
```

```
C * ARGUMENT IS (1,2) THE FORMAT IS 10E8.2; AND IF THE ARGUMENT IS     *VAR0502
C * (1,3) THE FORMAT IS 20F4.0. FOR PRINTED OUTPUT THE ARGUMENT OF(1,4)*VAR0503
C * OF ARRAY IOFT PASSED TO SUBROUTINE READS WILL RESULT IN PRINTED    *VAR0504
C * ARRAYS IN 10E11.3 FORMAT.                                          *VAR0505
C *                                                                    *VAR0510
C * IT IS ASSUMED THAT INPUT DENSITIES ARE IN KILOGRAMS/CUBIC METER.   *VAR0520
C * WITHIN PROGRAM, WHEN PURE WATER DENSITY IS READ IT IS CONVERTED TO *VAR0530
C * GRAMS/CUBIC CENTIMETER, AND THEN CONVERTED TO A RELATIVE DENSITY BY*VAR0540
C * DIVIDING BY THE DENSITY OF FRESHWATER (TAKEN TO BE 0.9977).        *VAR0550
C * THUS, PURE WATER DENSITY IS REPLACED BY 0.9977*RELATIVE DENSITY.   *VAR0560
C * IF TERMS FOR PRESSURE ARE TO BE DEVELOPED, CONDUCTANCES ARE DIVIDED*VAR0570
C * BY ACCELERATION OF GRAVITY (32.174) RATHER THAN 1.94*.9977*32.174. *VAR0580
C * ALSO RELATIVE GROUND WATER DENSITY IS MULTIPLIED BY 1.94*.9977*G   *VAR0590
C * FOR PRESSURE FORMULATION.                                          *VAR0600
C * TO CONVERT DENSITIES AND VISCOSITY TO RELATIVE DENSITY AND RELATIVE*VAR0610
C * VISCOSITY USE FAC MULTIPLIER OF ARRAY CONTROL RECORD.              *VAR0620
C * FOR NORTHERN MIDWEST RASA MODEL USED FAC=1/(.9977) FOR DENSITY AND *VAR0630
C * FAC=1/(1.127 CENTIPOISE)) FOR VISCOSITY.                           *VAR0640
C * NOTE THAT THE DENSITY OF 0.9977 and VISCOSITY OF 1.127 REPRESENT   *VAR0650
C * AMBIENT VALUES FOR FRESHWATER AREAS OF THE NORTHERN MIDWEST        *VAR0655
C * FOLLOWING INPUT OF THE ABOVE ARRAYS, THE MODEL GRID SPACING ALONG  *VAR0660
C * ROWS (DELX), AND COLUMNS (DELY) IS READ FROM UNIT 5 IN SUBROUTINE  *VAR0670
C * DEL (SEE COMMENTS IN SUBROUTINE DEL).                              *VAR0680
C *                          OUTPUT                                    *VAR0690
C *                          ------                                    *VAR0700
C * OUTPUT OF VARDEN PROGRAM ARRAYS IS IMPLEMENTED IN SUBROUTINE OUTP. *VAR0710
C * OUTPUT ARRAYS AND THEIR PROGRAM UNIT NUMBERS ARE AS FOLLOWS:       *VAR0720
C * MODIFIED TRANSMISSIVITY (TOUT)-- UNIT 18                           *VAR0730
C * MODIFIED STORAGE COEFFICIENTS (SOUT)-- UNIT 19                     *VAR0740
C * CONDUCTANCE ALONG ROWS (TR)--UNIT 11                               *VAR0750
C * CONDUCTANCE ALONG COLUMNS (TC)--UNIT 12                            *VAR0760
C * INDIVIDUAL SOURCE TERM FOR FLOW ALONG ROWS (ALPHA)--UNIT 13        *VAR0770
C * INDIVIDUAL SOURCE TERM FOR FLOW ALONG  COLUMNS (BETA)--UNIT 14     *VAR0780
C * TOTAL SOURCE TERM (PST)--UNIT 16                                   *VAR0790
C * INDIVIDUAL SOURCE TERM FOR FLOW BETWEEN LAYERS (GAMMA)--UNIT 17    *VAR0800
C * CONDUCTANCE BETWEEN LAYERS (TK)--UNIT 15                           *VAR0810
C * (SEE SUBROUTINE OUTP FOR FORMAT OF OUTPUT ARRAYS).                 *VAR0820
C * PRINT OUTPUT OF PROGRAM IS WRITTEN TO UNIT 6.                      *VAR0830
C *                                                                    *VAR0840
C * IN ADDITION TO THE ABOVE INPUT AND OUTPUT ARRAYS, INTERNAL PROGRAM *VAR0850
C * ARRAYS INCLUDE PSA,PSB,AND PS3 TERMS OF EQ. 5C.                    *VAR0860
C *                                                                    *VAR0865
C * DIMENSIONING OF PROGRAM IS SET BY PARAMETER STATEMENT, WHERE IR IS *VAR0870
C * NUMBER OF MODEL ROWS, IC IS NUMBER OF MODEL COLUMNS, AND KL IS NUM-*VAR0880
C * BER OF MODEL LAYERS.                                               *VAR0890
C * SUBROUTINES REQUIRED BY VARDEN ARE:                                *VAR0900
C * SUBROUTINE READS TO READ INPUT ARRAYS (OTHER THAN GRID SPACING)    *VAR0910
C * SUBROUTINE DEL TO READ GRID SPACING                                *VAR0920
C * SUBROUTINE ZERO TO SET ARRAYS TO ZERO AFTER COMPUTATIONS FOR LAYER *VAR0930
C * K ARE COMPLETED                                                    *VAR0940
C * SUBROUTINE OUTP TO WRITE OUTPUT ARRAYS IN TO SPECIFIED FILES.      *VAR0950
C * DIMENSIONING OF SUBROUTINES IS SET BY PROGRAM.                     *VAR0960
```

```
C * ****************************************************************VAR0970
C                                                                     VAR0980
      PARAMETER IR=37,IC=40,KL=5,KV=KL+1,KE=KL+2                      VAR0990
      DIMENSION                                                       VAR1000
     +    ALPHA(IR,IC),        BETA(IR,IC),        DELTA(IR,IC,KL),  VAR1010
     +    DELX(IC),            DELY(IR),           DEN(IR,IC,KE),    VAR1020
     +    DENP(IR,IC,KE),      ELEV(IR,IC,KL),     GAMMA(IR,IC),     VAR1030
     +    INFT(2,3),           IOFT(9,4),          PS3(IR,IC),       VAR1040
     +    PSA(IR,IC),          PSB(IR,IC),         PST(IR,IC),       VAR1050
     +    S(IR,IC,KL),         T(IR,IC,KL),        TC(IR,IC),        VAR1060
     +    THK(IR,IC,KV),       TK(IR,IC),          TKV(IR,IC,KV),    VAR1070
     +    TOUT(IR,IC),         TR(IR,IC),          VISC(IR,IC,KE)    VAR1080
     +   ,SOUT(IR,IC)                                                 VAR1090
      COMMON /MISC/                                                   VAR1100
     +    IO,                  JO                                    VAR1110
      DATA INFT/4H(10F,4H8.0),4H(10E,4H8.2),4H(20F,4H4.0)/            VAR1120
      DATA IOFT/4H(1HO,4H,I2,,4H2X,2,4HOF6.,4H1/(5,4HX,20,4HF6.1,4H)) ,VAR1130
     +4H    ,4H(1HO,4H,I5,,4H14F9,4H.5/(,4H1H ,,4H5X,1,4H4F9.,4H5)) ,4H VAR1140
     +   ,4H(1HO,4H,I5,,4H10E1,4H2.5/,4H(1H ,4H,5X,,4H10E1,4H2.5),4H) VAR1150
     +,4H(1HO,4H,I5,,4H10E1,4H1.3/,4H(1H ,4H,5X,,4H10E1,4H1.3),4H)   / VAR1160
C                                                                     VAR1170
C READ IN VARDEN PROGRAM CONTROL RECORD.                              VAR1180
C                                                                     VAR1190
      READ (5,1) IO,JO,KO,LIN,IPRESS,IDELTA                           VAR1200
    1 FORMAT(6I5)                                                     VAR1210
      WRITE (6,2) IO,JO,KO,IPRESS                                     VAR1220
    2 FORMAT (1HO,4I5)                                                VAR1230
      IF(IDELTA.NE.0) WRITE(6,986)                                    VAR1240
  986 FORMAT(1HO,' ********** DELTA FACTOR APPLIED TO TK *****')      VAR1250
      IF (IPRESS.EQ.0) WRITE (6,3)                                    VAR1260
    3 FORMAT (1HO,'******************** COEFFICIENTS FOR FRESHWATER  HEAVAR1270
     +D********************')                                         VAR1280
C                                                                     VAR1290
      DDELTU=1.0                                                      VAR1300
      DDELTL=1.0                                                      VAR1310
      KVV=KO+1                                                        VAR1320
      KEE=KO+2                                                        VAR1330
      K1=KO-1                                                         VAR1340
      J2=JO-2                                                         VAR1350
      I2=IO-2                                                         VAR1360
C                                                                     VAR1370
C             NOTE IN CALL TO READS PASS APPROPRIATE INDEX OF LAYERS. VAR1380
C             READ STORAGE COEFFICIENTS                               VAR1390
C                                                                     VAR1400
      DO 4 K=1,KO                                                     VAR1410
    4 CALL READS (S(1,1,K),INFT(1,2),IOFT(1,4),'S ARRAYS              'VAR1420
     +,LIN,K)                                                         VAR1430
C                                                                     VAR1440
C             READ  TRANSMISSIVITY                                    VAR1450
C                                                                     VAR1460
```

```
      DO 5 K=1,KO                                                       VAR1470
    5 CALL READS (T(1,1,K),INFT(1,2),IOFT(1,4),'T ARRAYS               'VAR1480
     +,LIN,K)                                                           VAR1490
C                                                                       VAR1500
C             READ IN NODE ELEVATIONS                                   VAR1510
C                                                                       VAR1520
      DO 6 K=1,KO                                                       VAR1530
        CALL READS (ELEV(1,1,K),INFT(1,1),IOFT(1,4),'ELEVATION ARRAYS   VAR1540
     +    ',LIN,K)                                                     VAR1550
    6 CONTINUE                                                          VAR1560
C                                                                       VAR1570
C             READ  DENSITY OF SOLUTION                                 VAR1580
C                                                                       VAR1590
      DO 7 K=2,KVV                                                      VAR1600
    7 CALL READS (DEN(1,1,K),INFT(1,1),IOFT(1,4),'RHO-S ARRAYS         'VAR1610
     +,LIN,K)                                                           VAR1620
C                                                                       VAR1630
C             READ DENSITY OF PURE WATER                                VAR1640
C                                                                       VAR1650
      DO 8 K=2,KVV                                                      VAR1660
    8 CALL READS (DENP(1,1,K),INFT(1,1),IOFT(1,4),'DEN-P ARRAYS         VAR1670
     +',LIN,K)                                                          VAR1680
C                                                                       VAR1690
C             READ VERTICAL LEAKANCE                                    VAR1700
C                                                                       VAR1710
      DO 9 K=2,KO                                                       VAR1720
    9 CALL READS (TKV(1,1,K),INFT(1,2),IOFT(1,4),'LEAKANCE ARRAYS       VAR1730
     +',LIN,K)                                                          VAR1740
C                                                                       VAR1750
C             READ THICKNESS OF CONFINING LAYERS                        VAR1760
C                                                                       VAR1770
      DO 10 K=2,KO                                                      VAR1780
   10 CALL READS (THK(1,1,K),INFT(1,1),IOFT(1,4),'CONFINING BED THICKNESVAR1790
     +S',LIN,K)                                                         VAR1800
C                                                                       VAR1810
C             READ IN VISCOSITY                                         VAR1820
C                                                                       VAR1830
      DO 11 K=2,KVV                                                     VAR1840
   11 CALL READS (VISC(1,1,K),INFT(1,1),IOFT(1,4),'VISCOSITY ARRAY      VAR1850
     +  ',LIN,K)                                                       VAR1860
C                                                                       VAR1870
C             READ IN PERMEABILITY/OVERBURDEN FACTOR                    VAR1880
C                                                                       VAR1890
      DO 12 K=1,KO                                                      VAR1900
   12 CALL READS (DELTA(1,1,K),INFT(1,1),IOFT(1,4),'K VARIATION WITH DEPVAR1910
     +TH ',LIN,K)                                                       VAR1920
      I1=I0-1                                                           VAR1930
      J1=J0-1                                                           VAR1940
C                                                                       VAR1950
C             FOR CROSS-SECTIONAL MODEL IN J DIRECTION(X DIRECTION) SET VAR1960
C             BOUNDARY ELEVATIONS ALONG I=1,AND I=3 TO BE THE SAME AS   VAR1970
C             ELEVATIONS OF NODES ALONG CROSS-SECTION.                  VAR1980
```

```
C                                                                       VAR1990
      IF (IO.GT.3) GO TO 16                                             VAR2000
      DO 15 K=1,KO                                                      VAR2010
        DO 14 J=1,JO                                                    VAR2020
          ELEV(1,J,K)=ELEV(2,J,K)                                       VAR2030
          ELEV(3,J,K)=ELEV(2,J,K)                                       VAR2040
   14   CONTINUE                                                        VAR2050
   15 CONTINUE                                                          VAR2060
   16 CONTINUE                                                          VAR2070
C                                                                       VAR2080
C ***********************************************************************VAR2090
C  THE FOLLOWING COMMENTED STATEMENTS PERTAIN TO CODE APPLICABLE TO     VAR2100
C  NORTHERN-MIDWEST MODEL. FOR NODES IN "PINCHOUT" AREAS T,DELTA,VISC,  VAR2110
C  DENP,AND DEN ARE SET TO CORRESPONDING VALUES OF TOP LAYER. ALSO SET  VAR2120
C  TKV OF CONFINING BEDS BOUNDED BY INACTIVE NODES TO ZERO.             VAR2130
C                                                                       VAR2140
C     DO 19 K=1,K1                                                      VAR2150
C       DO 18 I=1,IO                                                    VAR2160
C         DO 17 J=1,JO                                                  VAR2170
C           TT=T(I,J,K)                                                 VAR2180
C           SS=S(I,J,K)                                                 VAR2190
C     IF(SS.LT.0.0.AND.S(I,J,K+1).LT.0.0.AND.TT.EQ.0.0) TKV(I,J,K+1)=0.0VAR2200
C           IF ((TT.EQ.0.0).AND.(SS.LT.0.0)) DELTA(I,J,K)=DELTA(I,J,KO) VAR2210
C           IF ((TT.EQ.0.0).AND.(SS.LT.0.0)) VISC(I,J,K+1)=VISC(I,J,KV) VAR2220
C           IF ((TT.EQ.0.0).AND.(SS.LT.0.0)) DENP(I,J,K+1)=DENP(I,J,KV) VAR2230
C           IF ((TT.EQ.0.0.AND.(SS.LT.0.0)) DEN(I,J,K+1)=DENP(I,J,KV)   VAR2240
C  17     IF ((T(I,J,K).EQ.0.0).AND.(S(I,J,K).LT.0.0)) T(I,J,K)=T(I,J,KOVAR2250
C    +    )                                                            VAR2260
C  18   CONTINUE                                                        VAR2270
C  19 CONTINUE                                                          VAR2280
C ***********************************************************************VAR2290
C                                                                       VAR2300
C             ZERO BORDER NODES                                         VAR2310
C                                                                       VAR2320
      DO 22 K=1,KO                                                      VAR2330
        DO 20 I=1,IO                                                    VAR2340
          S(I,1,K)=-1.0                                                 VAR2350
          T(I,1,K)=0.0                                                  VAR2360
          S(I,JO,K)=-1.0                                                VAR2370
          T(I,JO,K)=0.0                                                 VAR2380
   20   CONTINUE                                                        VAR2390
        DO 21 J=2,J1                                                    VAR2400
          S(1,J,K)=-1.0                                                 VAR2410
          T(1,J,K)=0.0                                                  VAR2420
          S(IO,J,K)=-1.0                                                VAR2430
          T(IO,J,K)=0.0                                                 VAR2440
   21   CONTINUE                                                        VAR2450
   22 CONTINUE                                                          VAR2460
C                                                                       VAR2470
C             ZERO DUMMY ARRAYS AT TOP AND BOTTOM OF MODEL.             VAR2480
```

```
C                                                                       VAR2490
      KA=KVV                                                            VAR2500
      KB=KEE                                                            VAR2510
      DO 24 I=1,IO                                                      VAR2520
        DO 23 J=1,JO                                                    VAR2530
          TKV(I,J,1)=0.0                                                VAR2540
          TKV(I,J,KA)=0.0                                               VAR2550
          THK(I,J,1)=0.0                                                VAR2560
          THK(I,J,KA)=0.0                                               VAR2570
          DEN(I,J,1)=0.0                                                VAR2580
          DEN(I,J,KB)=0.0                                               VAR2590
          DENP(I,J,1)=0.0                                               VAR2600
          DENP(I,J,KB)=0.0                                              VAR2610
          VISC(I,J,1)=0.0                                               VAR2620
          VISC(I,J,KB)=0.0                                              VAR2630
   23   CONTINUE                                                         VAR2640
   24 CONTINUE                                                          VAR2650
C                                                                       VAR2660
C             READ IN DELX,DELY                                         VAR2670
C                                                                       VAR2680
      CALL DEL (DELX,1,JO)                                              VAR2690
      CALL DEL (DELY,2,IO)                                              VAR2700
C                                                                       VAR2710
C           SET UP UNIT FACTORS FOR FRESHWATER OR PRESSURE FORMULATION  VAR2720
C           IN BRITISH SYSTEM OF UNITS                                  VAR2730
C                                                                       VAR2740
      UNIT1=1.94*.9977                                                  VAR2750
      UNIT2=1.0                                                         VAR2760
      IF (IPRESS.NE.0) UNIT1=1.0/32.174                                 VAR2770
      IF (IPRESS.NE.0) UNIT2=32.174*1.94*.9977                          VAR2780
      SUMA=0.0                                                          VAR2790
      SUMB=0.0                                                          VAR2800
      SUMC=0.0                                                          VAR2810
      SUMT=0.0                                                          VAR2820
C                                                                       VAR2830
C             MODIFY TRANSMISSIVITY TO INCORPORATE DELTA, DENSITY      *VAR2840
C                         AND VISCOSITY.                               *VAR2850
C                                                                        VAR2860
      DO 27 K=1,KO                                                      VAR2870
        DO 26 I=2,I1                                                    VAR2880
          DO 25 J=2,J1                                                  VAR2890
   25     T(I,J,K)=T(I,J,K)*DELTA(I,J,K)*DENP(I,J,K+1)/VISC(I,J,K+1)    VAR2900
   26   CONTINUE                                                        VAR2910
   27 CONTINUE                                                          VAR2920
C                                                                       VAR2930
C ************************ START OF MAJOR LOOP ************************VAR2940
C             ZERO ARRAYS AFTER COMPUTATION FOR LAYER K                 VAR2950
C                                                                       VAR2960
      DO 42 K=1,KO                                                      VAR2970
        CALL ZERO (TR)                                                  VAR2980
        CALL ZERO (TC)                                                  VAR2990
        CALL ZERO (PSA)                                                 VAR3000
```

```
        CALL ZERO (PSB)                                                    VAR3010
        CALL ZERO (PS3)                                                    VAR3020
        CALL ZERO (PST)                                                    VAR3030
        CALL ZERO (ALPHA)                                                  VAR3040
        CALL ZERO (BETA)                                                   VAR3050
        CALL ZERO (GAMMA)                                                  VAR3060
      CALL ZERO (TOUT)                                                     VAR3070
      CALL ZERO(SOUT)                                                      VAR3080
      CALL ZERO(TK)                                                        VAR3090
        SMA=0.0                                                            VAR3100
        SMB=0.0                                                            VAR3110
        SMG=0.0                                                            VAR3120
        SMT=0.0                                                            VAR3130
        DO 30 I=I1,2,-1                                                    VAR3140
          DO 29 J=2,J1                                                     VAR3150
            IF (T(I,J,K).EQ.0.0) GO TO 29                                  VAR3160
C                                                                          VAR3170
C             COMPUTE GEOMETRY OF MODEL                                    VAR3180
C                                                                          VAR3190
            E=ELEV(I,J,K)                                                  VAR3200
            DZAP=ELEV(I,J+1,K)-E                                           VAR3210
            DZAM=-ELEV(I,J-1,K)+E                                          VAR3220
            DZBP=-ELEV(I+1,J,K)+E                                          VAR3230
            DZBM=ELEV(I-1,J,K)-E                                           VAR3240
            DJP=DELX(J+1)/2.                                               VAR3250
            DJ=DELX(J)/2.                                                  VAR3260
            DJM=DELX(J-1)/2.                                               VAR3270
            DIP=DELY(I+1)/2.                                               VAR3280
            DI=DELY(I)/2.                                                  VAR3290
            DIM=DELY(I-1)/2.                                               VAR3300
            DAP=SQRT((DJP+DJ)**2.+DZAP**2.)                                VAR3310
            DAM=SQRT((DJM+DJ)**2.+DZAM**2.)                                VAR3320
            DBP=SQRT((DIP+DI)**2.+DZBP**2.)                                VAR3330
            DBM=SQRT((DIM+DI)**2.+DZBM**2.)                                VAR3340
            DA=2.0*DJ*DAM/(DJM+DJ)                                         VAR3350
            DA2=2.0*DJP*DAP/(DJ+DJP)                                       VAR3360
            DB=2.0*DI*DBM/(DIM+DI)                                         VAR3370
            DB1=2.0*(DBM-DB/2.)                                            VAR3380
            DB2=2.0*DIP*DBP/(DI+DIP)                                       VAR3390
C                                                                          VAR3400
C             DEVELOP HARMONIC MEANS FOR PRESSURE OR HEAD FORMULATION      VAR3410
C                                                                          VAR3420
C         COMPUTE AND STORE HORIZONTAL CONDUCTANCE FOR OUTPUT              VAR3430
            T1=T(I,J,K)                                                    VAR3440
        T2=T(I,J+1,K)                                                      VAR3450
        T3=T(I-1,J,K)                                                      VAR3460
        TR(I,J)=((2.*T1*DB)*(T2*UNIT1))/(T2*DA+T1*DA2)                     VAR3470
        TC(I,J)=((2.*T1*DA)*(T3*UNIT1))/(T3*DB+T1*DB1)                     VAR3480
C                                                                          VAR3490
C                                                                          VAR3500
```

```
C                                                                       VAR3510
C             COMPUTE AND STORE STORAGE TERMS FOR SUBSEQUENT OUTPUT     VAR3520
C                                                                       VAR3530
      IF(S(I,J,K).EQ.-1.) SOUT(I,J)=-1.0                                VAR3540
      IF(S(I,J,K).EQ.-1.0) GO TO 222                                    VAR3550
            SOUT(I,J)=S(I,J,K)*DA*DB*UNIT1                              VAR3560
  222 CONTINUE                                                          VAR3570
C                                                                       VAR3580
C             DEVELOPE VERTICAL CONDUCTANCE COEFFICIENTS                VAR3590
C             NOTE THAT M+1 INDEX CORRESPONDS TO LAYER K INDEX          VAR3600
C                                                                       VAR3610
            M=K                                                         VAR3620
            DDENPL=(DENP(I,J,M+1)+DENP(I,J,M))/2.0                      VAR3630
            DDENPU=(DENP(I,J,M+1)+DENP(I,J,M+2))/2.0                    VAR3640
            DVISCL=(VISC(I,J,M+1)+VISC(I,J,M))/2.0                      VAR3650
            DVISCU=(VISC(I,J,M+1)+VISC(I,J,M+2))/2.0                    VAR3660
C                                                                       VAR3670
C             MODIFY VERTICAL CONDUCTANCE BY DELTA FACTOR IF IDELTA IS  VAR3680
C                    NOT ZERO.                                         *VAR3690
C                                                                       VAR3700
            IF (IDELTA.EQ.0) GO TO 28                                   VAR3710
            IF (K.NE.K0) DDELTU=(DELTA(I,J,K)+DELTA(I,J,K+1))*0.5       VAR3720
            IF (K.NE.1) DDELTL=(DELTA(I,J,K)+DELTA(I,J,K-1))*0.5        VAR3730
            IF (K.EQ.1) DDELTL=0.0                                      VAR3740
            IF (K.EQ.K0) DDELTU=0.0                                     VAR3750
   28       CONTINUE                                                    VAR3760
            TZL=TKV(I,J,M)*DA*DB*DDENPL*UNIT1*DDELTL/DVISCL             VAR3770
            TZU=TKV(I,J,M+1)*DA*DB*DDENPU*UNIT1*DDELTU/DVISCU           VAR3780
C                                                                       VAR3790
C             STORE COEFFICIENTS FOR MODEL INPUT                        VAR3800
C                                                                       VAR3810
            IF (K.LT.K0) TK(I,J)=TZU                                    VAR3820
C                                                                       VAR3830
C             DEVELOP HORIZONTAL SOURCE TERMS                           VAR3840
C                                                                       VAR3850
            M=K+1                                                       VAR3860
            D=DEN(I,J,M)                                                VAR3870
            DNAP=((DEN(I,J+1,M)+D)/2.0)*UNIT2                           VAR3880
            DNAM=((DEN(I,J-1,M)+D)/2.0)*UNIT2                           VAR3890
            DNBP=((DEN(I+1,J,M)+D)/2.0)*UNIT2                           VAR3900
            DNBM=((DEN(I-1,J,M)+D)/2.0)*UNIT2                           VAR3910
C                                                                       VAR3920
C             ALTER DENSITY TERM IF FRESH WATER FORMULATION             VAR3930
C                                                                       VAR3940
            IF (IPRESS.EQ.0) DNAP=(DNAP-1.0)                           VAR3950
            IF (IPRESS.EQ.0) DNAM=(DNAM-1.0)                           VAR3960
            IF (IPRESS.EQ.0) DNBP=(DNBP-1.0)                           VAR3970
            IF (IPRESS.EQ.0) DNBM=(DNBM-1.0)                           VAR3980
            ALPHA(I,J)=-TR(I,J)*DNAP*DZAP                              VAR3990
            BETA(I,J)=-TC(I,J)*DNBM*DZBM                               VAR4000
            PSA(I,J)=-TR(I,J)*DNAP*DZAP+TR(I,J-1)*DNAM*DZAM            VAR4010
            PSB(I,J)=-TC(I,J)*DNBM*DZBM+TC(I+1,J)*DNBP*DZBP            VAR4020
```

```
            SUMA=PSA(I,J)+SUMA                                              VAR4030
            SUMB=PSB(I,J)+SUMB                                              VAR4040
C                                                                           VAR4050
C             DEVELOP VERTICAL SOURCE TERM                                  VAR4060
C                                                                           VAR4070
            M=K                                                             VAR4080
            DDENL=(DEN(I,J,M+1)+DEN(I,J,M))/2.0*UNIT2                       VAR4090
            DDENU=(DEN(I,J,M+1)+DEN(I,J,M+2))/2.0*UNIT2                     VAR4100
C                                                                           VAR4110
C             ALTER DENSITY TERM IF FRESHWATER FORMULATION                  VAR4120
C                                                                           VAR4130
            IF (IPRESS.EQ.0) DDENL=DDENL-1.0                                VAR4140
            IF (IPRESS.EQ.0) DDENU=DDENU-1.0                                VAR4150
            IF (THK(I,J,K+1).LT.1.0) THK(I,J,K+1)=1.0                       VAR4160
            IF (THK(I,J,K).LT.1.0) THK(I,J,K)=1.0                           VAR4170
            GAMMA(I,J)=-TZU*DDENU*THK(I,J,K+1)*(DELX(J)*DELY(I)/(DA*DB))VAR4180
            PS3(I,J)=(+TZL*DDENL*THK(I,J,K)-TZU*DDENU*THK(I,J,K+1))*(DELVAR4190
     +      X(J)*DELY(I)/(DA*DB))                                           VAR4200
            PST(I,J)=-PSA(I,J)-PSB(I,J)-PS3(I,J)                            VAR4210
            SUMC=PS3(I,J)+SUMC                                              VAR4220
            SUMT=PST(I,J)+SUMT                                              VAR4230
            SMA=PSA(I,J)+SMA                                                VAR4240
            SMB=PSB(I,J)+SMB                                                VAR4250
            SMG=PS3(I,J)+SMG                                                VAR4260
            SMT=PST(I,J)+SMT                                                VAR4270
   29     CONTINUE                                                          VAR4280
   30   CONTINUE                                                            VAR4290
C                                                                           VAR4300
C ***************************************************************************VAR4310
C  THE FOLLOWING COMMENTED STATEMENTS PERTAIN TO CODE APPLICABLE TO         VAR4320
C  NORTHERN-MIDWEST MODEL. FOR INACTIVE NODES CONDUCTANCES AND SOURCE       VAR4330
C  TERMS ARE SET TO ZERO.                                                   VAR4340
C                                                                           VAR4350
C       IF (K.EQ.K0) GO TO 35                                               VAR4360
C       DO 32 I=I1,2,-1                                                     VAR4370
C         DO 31 J=2,J2                                                      VAR4380
C           T5=T(I,J,K0)                                                    VAR4390
C           IF (T(I,J,K).NE.T5) GO TO 31                                    VAR4400
C           IF ((T(I,J+1,K).NE.T5).AND.(S(I,J+1,K).GE.0.0)) GO TO 31        VAR4410
C           ALPHA(I,J)=0.0                                                  VAR4420
C           PST(I,J)=PST(I,J)+PSA(I,J)                                      VAR4430
C           TR(I,J)=0.0                                                     VAR4440
C  31     CONTINUE                                                          VAR4450
C  32   CONTINUE                                                           VAR4460
C       DO 34 J=2,J1                                                        VAR4470
C         DO 33 I=I1,3,-1                                                   VAR4480
C           T5=T(I,J,K0)                                                    VAR4490
C           IF (T(I,J,K).NE.T5) GO TO 33                                    VAR4500
C           IF ((T(I-1,J,K).NE.T5).AND.(S(I-1,J,K).GE.0.0)) GO TO 33        VAR4510
C           BETA(I,J)=0.0                                                   VAR4520
C           PST(I,J)=PST(I,J)+PSB(I,J)                                      VAR4530
C           TC(I,J)=0.0                                                     VAR4540
```

```
C  33     CONTINUE                                                      VAR4550
C  34   CONTINUE                                                        VAR4560
C  35   CONTINUE                                                        VAR4570
C ***************************************************************************VAR4580
C                                                                       VAR4590
C            SET TRANSMISSIVITY TO CONFORM TO MULTIAQUIFER FORMULATION. VAR4600
C                                                                       VAR4610
        DO 37 I=2,I1                                                    VAR4620
          DO 36 J=2,J1                                                  VAR4630
   36     TOUT(I,J)=T(I,J,K)*UNIT1                                      VAR4640
   37   CONTINUE                                                        VAR4650
C     PRINT SUMMATION OF SOURCE TERMS FOR LAYER K                       VAR4660
        WRITE (6,38)                                                    VAR4670
   38   FORMAT (1H0,125('*'))                                           VAR4680
        WRITE (6,39) K                                                  VAR4690
   39   FORMAT (1X,'LAYER',1X,I1,5X,'SUMA,SUMB,SUMG,SUMT')              VAR4700
        WRITE (6,40) SMA,SMB,SMG,SMT                                    VAR4710
   40   FORMAT (1X,4(E12.5))                                            VAR4720
C            OUTPUT ARRAYS FOR LAYER K                                  VAR4730
        CALL OUTP (TOUT,K,18,'MULTIAQUIFER T         ',2)               VAR4740
        CALL OUTP (SOUT,K,19,'STORAGE COEFFICIENTS   ',1)               VAR4750
        CALL OUTP (TR,K,11,'TR COEFFICIENTS        ',2)                 VAR4760
        CALL OUTP (TC,K,12,'TC COEFFICIENTS        ',2)                 VAR4770
        CALL OUTP (ALPHA,K,13,'ALPHA SOURCE TERM     ',2)               VAR4780
        CALL OUTP (BETA,K,14,'BETA SOURCE TERM       ',2)               VAR4790
        CALL OUTP (PST,K,16,'TOTAL SOURCE TERM     ',2)                 VAR4800
        IF (K.EQ.K0) GO TO 41                                           VAR4810
        CALL OUTP (GAMMA,K,15,'GAMMA SOURCE TERM     ',2)               VAR4820
        CALL OUTP (TK,K,17,'TK COEFFICIENT         ',2)                 VAR4830
   41   CONTINUE                                                        VAR4840
   42 CONTINUE                                                          VAR4850
C     PRINT SUMMATION OF SOURCE TERMS OVER ALL LAYERS                   VAR4860
      WRITE (6,43)                                                      VAR4870
   43 FORMAT (1H0,' SOURCE TERM TOTALS: ALPHA,BETA,GAMMA,TOTAL')        VAR4880
      WRITE (6,44) SUMA,SUMB,SUMC,SUMT                                  VAR4890
   44 FORMAT (1H0,4E12.3)                                               VAR4900
      WRITE (6,45)                                                      VAR4910
   45 FORMAT (1H0,' PROGRAM COMPLETE')                                  VAR4920
      STOP                                                              VAR4930
      END                                                               VAR4940
C***************************************************************************VAR4950
      SUBROUTINE DEL (D,IM,LL)                                          VAR4960
C ***************************************************************************VAR4970
C * SUBROUTINE TO READ MODEL NODE SPACING ARRAY IN EITHER ROW OR COLUMN*VAR4980
C * DIRECTION. EACH ARRAY REQUIRES A CONTROL RECORD CONTAINING VALUES  *VAR4990
C * OF FAC,IVAR,IPRN. CONTROL RECORD AND NODE SPACING ARRAY ARE READ   *VAR5000
C * FROM UNIT 5. SEE COMMENTS IN SUBROUTINE READS CONCERNING FORM OF   *VAR5010
C * CONTROL RECORD.                                                    *VAR5020
C ***************************************************************************VAR5030
```

```
      DIMENSION                                                          VAR5040
     +    D(100)                                                        VAR5050
C                                                                        VAR5060
C                                                                        VAR5070
      IC=IM                                                              VAR5080
      READ (5,1) FAC,IVAR,IPRN                                           VAR5090
    1 FORMAT (F10.0,2I10)                                                VAR5100
      IF (IVAR.EQ.1) READ (5,2) (D(J),J=1,LL)                            VAR5110
    2 FORMAT (8G10.0)                                                    VAR5120
      DO 4 J=1,LL                                                        VAR5130
        IF (IVAR.NE.1) GO TO 3                                           VAR5140
        D(J)=D(J)*FAC                                                    VAR5150
        GO TO 4                                                          VAR5160
    3   D(J)=FAC                                                         VAR5170
    4 CONTINUE                                                           VAR5180
      IF (IC.NE.1) GO TO 7                                               VAR5190
      IF (IVAR.EQ.1.AND.IPRN.NE.1) WRITE (6,5) (D(J),J=1,LL)             VAR5200
    5 FORMAT (1H1,'GRID SPACING IN X DIRECTION'//12F10.0)                VAR5210
      IF (IVAR.EQ.0) WRITE (6,6) FAC                                     VAR5220
    6 FORMAT (1H0,72X,'DELX =',G15.7)                                    VAR5230
      RETURN                                                             VAR5240
    7 IF (IVAR.EQ.1.AND.IPRN.NE.1) WRITE (6,8) (D(J),J=1,LL)             VAR5250
    8 FORMAT (1H1,'GRID SPACING IN Y DIRECTION'//12F10.0)                VAR5260
      IF (IVAR.EQ.0) WRITE (6,9) FAC                                     VAR5270
    9 FORMAT (1H0,72X,'DELY =',G15.7)                                    VAR5280
      RETURN                                                             VAR5290
      END                                                                VAR5300
C************************************************************************VAR5310
      SUBROUTINE OUTP (A,K,LOUT,IN,KOUT)                                 VAR5320
C ***********************************************************************VAR5330
C * SUBROUTINE TO OUTPUT VARDEN PROGRAM ARRAYS IN FORMAT OF TRESCOTT   *VAR5340
C * MODEL INPUT.                                                       *VAR5350
C * OUTPUT FORMAT GOVERNED BY VALUE OF KOUT IN SUBROUTINE ARGUMENT     *VAR5360
C * LIST. IF KOUT =1 FORMAT IS (8F10.0), IF KOUT=2 FORMAT IS (8E10.4), *VAR5370
C * IF KOUT=3 FORMAT IS (8F8.4).                                       *VAR5380
C ***********************************************************************VAR5390
      CHARACTER*4                                                        VAR5400
     +    IN(6)                                                         VAR5410
      COMMON /MISC/                                                      VAR5420
     +    IO,            JO                                            VAR5430
      DIMENSION                                                          VAR5440
     +    V(2)                                                          VAR5450
      DIMENSION                                                          VAR5460
     +    A(IO,JO)                                                      VAR5470
      DATA (V(I),I=1,2)/4H(10F,4H8.5)/                                   VAR5480
C                                                                        VAR5490
C                                                                        VAR5500
      WRITE (6,1) IN,K                                                   VAR5510
    1 FORMAT (1H0,2X,6A4,' FOR LAYER',I3/2X,39('-'))                     VAR5520
      V(1)=4H(8F1                                                        VAR5530
```

```
      IF (KOUT.EQ.2) V(1)=4H(8E1                                        VAR5540
      V(2)=4H0.4)                                                       VAR5550
      IF (KOUT.EQ.1) V(2)=4H0.0)                                        VAR5560
      IF (KOUT.EQ.3) V(2)=4H8.4)                                        VAR5570
      DO 2 I=1,I0                                                       VAR5580
    2 WRITE (LOUT,V) (A(I,J),J=1,J0)                                    VAR5590
      RETURN                                                            VAR5600
      END                                                               VAR5610
C*********************************************************************VAR5620
      SUBROUTINE READS (A,INFT,IOFT,IN,LIN,K)                           VAR5630
C **********************************************************************VAR5640
C * SUBROUTINE READS IS A MODIFICATION OF ENTRY ARRAY OF SUBROUTINE    *VAR5650
C * DATAI OF TRESCOTT MODEL. EACH INPUT ARRAY REQUIRES A CONTROL RECORD*VAR5660
C * CONTAINING VALUES OF FAC, IVAR, AND IPRN SIMILAR TO THE CONTROL    *VAR5670
C * RECORD FOR ARRAY INPUT OF TRESCOTT MODEL. IF ARRAY IN QUESTION     *VAR5680
C * CONSISTS OF A CONSTANT VALUE FOR EACH NODE, SET FAC= THE CONSTANT  *VAR5690
C * AND SET IVAR=0. IN THIS CASE NO ARRAY IS READ.                     *VAR5700
C * IF THE ARRAY VALUES ARE NOT CONSTANT SET FAC TO A CONSTANT AND SET *VAR5710
C * IVAR=1. IN THIS CASE AN ARRAY WILL BE READ AND ALL VALUES ARE MUL- *VAR5720
C * TIPLIED BY FAC. IN EITHER CASE SET IPRN=0 TO PRINT INPUT ARRAY OR  *VAR5730
C * SET IPRN=1 IF PRINT OF INPUT ARRAY IS NOT DESIRED.                 *VAR5740
C **********************************************************************VAR5750
      CHARACTER*4                                                       VAR5760
     +    IN(6)                                                        VAR5770
      COMMON /MISC/                                                     VAR5780
     +    I0,            J0                                            VAR5790
      DIMENSION                                                         VAR5800
     +    A(I0,J0),      INFT(2,2),     IOFT(9,4)                     VAR5810
C                                                                       VAR5820
C                                                                       VAR5830
      READ (5,1) FAC,IVAR,IPRN                                          VAR5840
    1 FORMAT (G10.0,2I10,3F10.0,2I10)                                   VAR5850
      IRECS=0                                                           VAR5860
      IC=4*IRECS+2*IVAR+IPRN+1                                          VAR5870
      GO TO (2,2,5,5,9,9), IC                                           VAR5880
    2 DO 3 I=1,I0                                                       VAR5890
        DO 3 J=1,J0                                                     VAR5900
    3 A(I,J)=FAC                                                        VAR5910
      WRITE (6,4) IN,FAC,K                                              VAR5920
    4 FORMAT (1H0,52X,6A4,' =',G15.7,' FOR LAYER',I3)                   VAR5930
      GO TO 11                                                          VAR5940
    5 IF (IC.EQ.3) WRITE (6,6) IN,K                                     VAR5950
    6 FORMAT (1H0,45X,6A4,' MATRIX, LAYER',I3/46X,41('-'))              VAR5960
      DO 8 I=1,I0                                                       VAR5970
        READ (LIN,INFT) (A(I,J),J=1,J0)                                 VAR5980
        DO 7 J=1,J0                                                     VAR5990
          A(I,J)=A(I,J)*FAC                                             VAR6000
    7   CONTINUE                                                        VAR6010
    8 IF (IC.EQ.3) WRITE (6,IOFT) I,(A(I,J),J=1,J0)                     VAR6020
      GO TO 11                                                          VAR6030
    9 CONTINUE                                                          VAR6040
```

```
      IF (IC.EQ.6) GO TO 11                                              VAR6050
      WRITE (6,6) IN,K                                                   VAR6060
      DO 10 I=1,I0                                                       VAR6070
   10 WRITE (6,IOFT) I,(A(I,J),J=1,J0)                                   VAR6080
   11 CONTINUE                                                           VAR6090
      RETURN                                                             VAR6100
      END                                                                VAR6110
C*********************************************************************VAR6120
      SUBROUTINE ZERO (A)                                                VAR6130
C **********************************************************************VAR6140
C * SUBROUTINE TO ZERO ARRAYS AT THE START OF PROCESSING FOR EACH LAYER*VAR6150
C **********************************************************************VAR6160
      COMMON /MISC/                                                      VAR6170
     +    I0,           J0                                               VAR6180
      DIMENSION                                                          VAR6190
     +    A(I0,J0)                                                       VAR6200
C                                                                        VAR6210
C                                                                        VAR6220
      DO 2 I=1,I0                                                        VAR6230
        DO 1 J=1,J0                                                      VAR6240
    1   A(I,J)=0.0                                                       VAR6250
    2 CONTINUE                                                           VAR6260
      RETURN                                                             VAR6270
      END                                                                VAR6280
```

APPENDIX C.--PROGRAM STEAM

```
C ***************************************************************************
C *                          PROGRAM STEAM                                  *STM0010
C *                          -------------                                  *STM0015
C *PROGRAM TO COMPUTE DENSITY OF GROUND WATER AND DENSITY OF PURE WATER     *STM0020
C * COMPONENT USING PRELIMINARY STEAM TABLES OF POTTER,R.W., AND D.L.       *STM0030
C * BROWN,1977,"THE VOLUMETRIC PROPERTIES OF AQUEOUS SODIUM CHLORIDE       *STM0040
C * SOLUTIONS FROM 0-500 DEGREES C. AT PRESSURES UP TO 2000 BARS BASED     *STM0050
C * ON A REGRESSION OF AVAILABLE DATA IN THE LITERATURE", GEOLOGICAL       *STM0060
C * SURVEY BULLETIN 1421-C.                                                *STM0070
C * PROGRAM ALSO COMPUTES VISCOSITY OF GROUND WATER FROM FIT TO DATA OF    *STM0080
C * MATTHEWS,C.S.,AND D.G. RUSSEL,1967,"PRESSURE BUILDUP AND FLOW TESTS    *STM0090
C * IN WELLS",APPENDIX G: DALLAS,TEXAS, SOC. OF PETROLEUM ENGS. OF THE     *STM0100
C * AM.INST. OF MINING,METALLURGICAL, AND PETROLEUM ENGS. MONOGRAPH #1,    *STM0110
C * P. 158.                                                                *STM0120
C * ORIGINAL PROGRAM WRITTEN AND DOCUMENTED BY EMANUEL WEISS,1982,         *STM0130
C * "A MODEL FOR THE SIMULATION OF VARIABLE-DENSITY GROUND WATER IN        *STM0140
C * THREE DIMENSIONS UNDER STEADY-STATE CONDITIONS", U.S.GEOLOGICAL        *STM0150
C * SURVEY OPEN-FILE REPORT 82-352.                                        *STM0160
C * THIS VERSION OF PROGRAM MODIFIED BY R.J. MANDLE AND A.L. KONTIS.       *STM0170
C * MODIFICATIONS CONSIST OF COMPUTING TEMPERATURE DISTRIBUTION FROM       *STM0180
C * ASSUMED GEOTHERMAL GRADIENT AND TO USE BICUBIC INTERPOLATION OF        *STM0190
C * POTTER AND BROWN, TABLE 29 TO OBTAIN Ca,Cb,Cc COEFFICIENTS.           *STM0200
C * PROCEEDURE TO OBTAIN INTERPOLATION COEFFICIENTS GIVEN IN DAVIS,T.M.    *STM0210
C * AND KONTIS,A.L.,1970," SPLINE INTERPOLATION ALGORITHMS FOR TRACK-      *STM0220
C * TYPE DATA WITH APPLICATION TO THE COMPUTATION OF MEAN GRAVITY ANO-      *STM0230
C * MALIES", U.S. NAVAL OCEANOGRAPHIC OFFICE TECHNICAL REPORT-226.         *STM0240
C *                        INPUT DATA                                      *STM0250
C *                        ----------                                      *STM0260
C * READ INPUT DATA FROM FORTRAN UNIT NUMBERS AS FOLLOWS:                  *STM0270
C * CONTROL RECORD (CDIFF,IPRINT,IO,JO) FROM UNIT 5                        *STM0280
C * DISSOLVED SOLIDS (TDS) IN GRAMS PER LITER FROM UNIT 10                 *STM0290
C * FRESHWATER HEADS (H) IN FEET FROM UNIT 11                              *STM0300
C * NODE ELEVATIONS (ELEV) IN FEET FROM UNIT 12                            *STM0310
C * POTTER TABLE 29 -Ca BICUBIC COEFFICIENTS (BC) FROM UNIT 13             *STM0320
C * POTTER TABLE 29 -Cb BICUBIC COEFFICIENTS (BC) FROM UNIT 14             *STM0330
C * POTTER TABLE 29 -Cc BICUBIC COEFFICIENTS (BC) FROM UNIT 15             *STM0340
C *                        OUTPUT DATA                                     *STM0350
C *                        -----------                                     *STM0360
C * WRITE OUTPUT DATA ON FORTRAN UNIT NUMBERS AS FOLLOWS:                  *STM0370
C * PROGRAM PRINTOUT ON UNIT 6                                             *STM0380
C * VISCOSITY (ZMU) IN CENTIPOISE ON UNIT 20                               *STM0390
C * GROUND WATER DENSITY (XDEN) IN KG. PER CU.METER ON UNIT 21             *STM0400
C * PURE WATER DENSITY (XDENP) IN KG. PER CU.METER ON UNIT 22              *STM0410
C *                    PROGRAM INTERNAL ARRAY NAMES                        *STM0420
C *                    ----------------------------                        *STM0430
C * OTHER ARRAY NAMES USED IN PROGRAM ARE AS FOLLOWS:                      *STM0440
C * P IS PRESSURE IN BARS                                                  *STM0450
C * TEMP IS TEMPERATURE IN DEGREES CELSIUS FOR DENSITY CALCULATIONS        *STM0460
C * AND IN DEGREES FARENHEIT FOR VISCOSITY CALCULATIONS                    *STM0470
C * DEN IS GROUND WATER DENSITY IN GRAMS PER CUBIC CM.                     *STM0480
C * DENP IS PURE WATER DENSITY IN GRAMS PER CUBIC CM.                      *STM0490
C *                                                                        *STM0500
```

```
C * SUBROUTINES CONSIST OF: SUBROUTINE ARRAYS TO READ INPUT ARRAYS AND *STM0510
C * SUBROUTINE BICUBE TO READ BICUBIC COEFFICIENTS AND TO PERFORM      *STM0520
C * INTERPOLATION OF POTTER TABLE 29 VALUES.                          *STM0530
C * DIMENSIONING IS SET BY PARAMETER STATEMENTS IN MAIN PROGRAM AND    *STM0540
C * IN SUBROUTINE ARRAYS: NR=NUMBER OF ROWS, NC=NUMBER OF COLUMNS      *STM0550
C * NOTE THAT PROGRAM IS WRITTEN TO COMPUTE  ONE LAYER AT A TIME.      *STM0560
C * IF THERE ARE KO LAYERS, RUN PROGRAM KO TIMES.                      *STM0565
C ***************************************************************************STM0570
      PARAMETER NR=37,NC=40                                                STM0580
      DIMENSION H(NR,NC),INFT(2,2),IOFT(9,4),ZMU(NR,NC)                    STM0590
      DIMENSION TEMP(NR,NC), ELEV(NR,NC), TDS(NR,NC), P(NR,NC)             STM0600
      DIMENSION DEN(NR,NC), DENP(NR,NC), XDEN(NR,NC), XDENP(NR,NC)         STM0610
      DIMENSION E(3)                                                       STM0620
      COMMON/MISC/IO,JO                                                    STM0630
C --- E(I) ARE CONSTANTS IN POTTER'S EQ 1 FOR 20 DEGREES CENTIGRADE        STM0640
      DATA E/15.782,2.0324,0.0744/                                         STM0650
      DATA INFT/4H(20F,4H4.0),4H(10F,4H8.0)/                               STM0660
      DATA IOFT/'(1HO',',I2,','2X,2','0F6.','1/(5','X,20','F6.1',')) ',STM0670
     1'    ','(1HO',',I5,','14F9','.5/(','6X,1','4F9.','5)) ','    ','  STM0680
     2  ','(1HO',',I5,','10E1','2.5/','(6X,','10E1','2.5)',')   ','    'STM0690
     3,'(1HO',',I5,','10E1','1.3/','(6X,','10E1','1.3)',')   ','    '/  STM0700
C     SPECIFY THE MOLECULAR WEIGHT OF NACL (WT) AND FACTOR (CF) TO         STM0710
C     CONVERT FEET OF WATER TO BARS (PRESSURE)                             STM0720
      WT=58.4428                                                           STM0730
      CF=.02987712                                                         STM0740
C---- READ IN CLOSURE CRITERION(CDIFF) FOR MOLALITY ITERATION IN PERCENTSTM0750
C---- ,PRINT CODE(IPRINT), NUMBER OF MODEL ROWS (IO), AND COLUMNS (JO)  STM0760
C---- FOR PARTIAL PRINTOUT ON UNIT 6, SET IPRINT=0.                      STM0770
C     FOR FULL PRINTOUT OF CALCULATIONS AT EACH ITERATION SET IPRINT=1   STM0780
      READ (5,250) CDIFF,IPRINT,IO,JO                                      STM0790
      WRITE (6,260) CDIFF,IPRINT                                           STM0800
      WRITE (6,280) IO,JO                                                  STM0810
C---- READ IN DISSOLVED SOLIDS (IN GRAMS PER LITER)                        STM0820
      DO 97 I=1,IO                                                         STM0830
      READ(10,98)(TDS(I,J),J=1,JO)                                         STM0840
   98 FORMAT(10F8.3)                                                       STM0850
   97 CONTINUE                                                             STM0860
C---- READ IN 'FRESHWATER' HEADS (PURE WATER AT STP) IN FEET               STM0870
      CALL ARRAYS(H,INFT(1,2),IOFT(1,1),11)                                STM0880
C---- READ IN NODE ELEVATION (IN FEET RELATIVE TO MSL)                     STM0890
      CALL ARRAYS(ELEV,INFT(1,2),IOFT(1,4),12)                             STM0900
      IOM1=IO-1                                                            STM0910
      JOM1=JO-1                                                            STM0920
      DO 110 J=1,JO                                                        STM0930
      DO 110 I=1,IO                                                        STM0940
      DEN(I,J)=0.0                                                         STM0950
      DENP(I,J)=0.0                                                        STM0960
  110 P(I,J)=0.0                                                           STM0970
C                                                                          STM0980
C     COMPUTE TEMPERATURE AS FUNCTION OF DEPTH FROM GEOTHERMAL GRADIENT  STM0990
```

```
C     OF 0.00083 C/FT +17.09 OBTAINED BY FITTING STRAIGHT LINE TO DATA  STM1000
C     FROM ILLINOIS BASIN.                                              STM1010
C                                                                       STM1020
      DO 121 I=1,I0                                                     STM1030
      DO 125 J=1,J0                                                     STM1040
      TEMP(I,J)=0.0083*(H(I,J)-ELEV(I,J))+17.09                         STM1050
  125 CONTINUE                                                          STM1060
      WRITE(6,126)(TEMP(I,J),J=1,J0)                                    STM1070
  126 FORMAT(1X,10F8.2)                                                 STM1080
  121 CONTINUE                                                          STM1090
C                                                                       STM1100
C ****** START MAJOR LOOP TO COMPUTE MOLALITY AND DENSITIES *****       STM1110
      DO 220 I=2,I0M1                                                   STM1120
      DO 220 J=2,J0M1                                                   STM1130
      P(I,J)=CF*(H(I,J)-ELEV(I,J))                                      STM1140
      XMOL=TDS(I,J)*1000/((1000-TDS(I,J))*WT)                           STM1150
      N=0                                                               STM1160
  130 CONTINUE                                                          STM1170
C --- DEN0 IS DENSITY OF PURE WATER AT 20 DEG.C. AND 1BAR               STM1180
C     INTERPOLATED FROM TABLE 29 OF POTTER AND BROWN.                   STM1185
      DEN0 = 0.9976054                                                  STM1190
      DNM=1000+(E(1)*XMOL+E(2)*(SQRT(XMOL)**3)+E(3)*(XMOL**2))*DEN0     STM1200
      DEN1=(1000*DEN0+WT*XMOL*DEN0)/DNM                                 STM1210
      N=N+1                                                             STM1220
      XMOL1=TDS(I,J)*1000/((DEN1*1000-TDS(I,J))*WT)                     STM1230
      CRTN=ABS((XMOL-XMOL1)*100/XMOL)                                   STM1240
      XMOL=XMOL1                                                        STM1250
      IF (N.EQ.1.AND.IPRINT.EQ.1) WRITE (6,320)                         STM1260
      IF (PRINT.EQ.1) WRITE (6,330) I,J,N,XMOL1,DEN1,TDS(I,J),CRTN      STM1270
      IF(CRTN.GT.CDIFF.AND.N.LT.10)  GO TO 130                          STM1280
      IF(N.LE.9) GO TO 140                                              STM1290
      WRITE (6,340) I,J,N,DEN1,XMOL1,CRTN                               STM1300
  140 CONTINUE                                                          STM1310
C     FOR MOLALITY AND PRESSURE OF NODE I,J CALL SUBROUTINE BICUBE THREESTM1320
C     TIMES TO OBTAIN POTTER TABLE 29 VALUES (ACOEF= Ca, BCOEF= Cb,     STM1330
C     CCOEFF= Cc) AND REWIND FILES FOR NEXT NODE COMPUTATION.           STM1340
      LIN=13                                                            STM1350
      PP=P(I,J)                                                         STM1360
      CALL BICUBE(XMOL,PP,ACOEF,LIN)                                    STM1370
      REWIND 13                                                         STM1380
      LIN=14                                                            STM1390
      CALL BICUBE(XMOL,PP,BCOEF,LIN)                                    STM1400
      REWIND 14                                                         STM1410
      LIN=15                                                            STM1420
      CALL BICUBE(XMOL,PP,CCOEF,LIN)                                    STM1430
      REWIND 15                                                         STM1440
C                                                                       STM1450
C     COMPUTE GROUND WATER DENSITY                                      STM1460
C                                                                       STM1470
      RHO=ACOEF-(1.E-4)*BCOEF*TEMP(I,J)-(1.E-6)*CCOEF*(TEMP(I,J)**2.)   STM1480
  180 CONTINUE                                                          STM1490
C                                                                       STM1500
```

```
C     COMPUTE PURE WATER DENSITY                                          STM1510
C                                                                         STM1520
      DEN(I,J)=RHO                                                        STM1530
      DENP(I,J)=DEN(I,J)*(1-XMOL*WT/(XMOL*WT+1000))                       STM1540
  220 CONTINUE                                                            STM1550
C                                                                         STM1560
C     CONVERT DENSITIES TO KILOGRAMS PER CUBIC METER                      STM1570
C                                                                         STM1580
      DO 230 J=1,JO                                                       STM1590
      DO 230 I=1,IO                                                       STM1600
      XDEN(I,J)=(DEN(I,J)*1000)                                           STM1610
  230 XDENP(I,J)=(DENP(I,J)*1000)                                         STM1620
C                                                                         STM1630
C     CALCULATE VISCOSITY FROM GRAPH IN MATTHEWS AND RUSSELL,1967         STM1640
C                                                                         STM1650
      DO 232 I=1,IO                                                       STM1660
      DO 231 J=1,JO                                                       STM1670
      TEMP(I,J)=TEMP(I,J)*9/5+32                                          STM1680
      XMU=38.3432/SQRT(TEMP(I,J))-14.623/SQRT(SQRT(TEMP(I,J)))            STM1690
      YMU=XMU+1.481                                                       STM1700
      TDSMU=1.+(TDS(I,J)/300)                                             STM1710
      ZMU(I,J)=YMU*TDSMU                                                  STM1720
  231 CONTINUE                                                            STM1730
C                                                                         STM1740
C     WRITE OUTPUT ARRAYS                                                 STM1750
C                                                                         STM1760
      WRITE(6,391)                                                        STM1770
      WRITE(20,392)(ZMU(I,J),J=1,JO)                                      STM1780
  391 FORMAT(1X,'VISCOSITY DATA')                                         STM1790
  392 FORMAT(10F8.3)                                                      STM1800
  232 CONTINUE                                                            STM1810
      WRITE (6,360)                                                       STM1820
      DO 393 I=1,IO                                                       STM1830
      WRITE (6,390) (XDEN(I,J),J=1,JO)                                    STM1840
  393 CONTINUE                                                            STM1850
      WRITE (6,370)                                                       STM1860
      DO 394 I=1,IO                                                       STM1870
      WRITE (6,390) (XDENP(I,J),J=1,JO)                                   STM1880
      WRITE (21,390) (XDEN(I,J),J=1,JO)                                   STM1890
      WRITE (22,390) (XDENP(I,J),J=1,JO)                                  STM1900
  394 CONTINUE                                                            STM1910
      WRITE(6,789)                                                        STM1920
  789 FORMAT(1HO,' PRESSURE ARRAY')                                       STM1930
      DO 787 I=1,IO                                                       STM1940
  787 WRITE(6,390) (P(I,J),J=1,JO)                                        STM1950
      STOP                                                                STM1960
C                                                                         STM1970
  250 FORMAT (F10.6,3I10)                                                 STM1980
  260 FORMAT ('1',T35,'ITERATION CLOSURE CRITERION FOR DENSITY(IN %) =',  STM1990
     1F10.6,20X,'PRINT ='F4.1)                                            STM2000
  270 FORMAT (3I10)                                                       STM2010
  280 FORMAT (T45,'IO JO ',3I3)                                           STM2020
```

```
  320 FORMAT (/,T25,'XMOL1',6X,'DEN1',7X,'TDS',7X,'CRTN')                  STM2030
  330 FORMAT (T2,3I5,4F10.5)                                               STM2040
  340 FORMAT(T2,'GT 10 ITERATIONS',3I5,'DN1=',F10.6,'XMOL=',F10.6,'CRTN=STM2050
     1',F8.4)                                                              STM2060
  360 FORMAT ('1',T22,'GROUND WATER DENSITY (G/L)',/)                      STM2070
  370 FORMAT ('1',T22,'PURE WATER CONCENTRATION (G/L)',/)                  STM2080
  390 FORMAT(10F8.1)                                                       STM2090
      END                                                                  STM2100
C     **********                                                           STM2110
      SUBROUTINE ARRAYS(A,INFT,IOFT,LIN)                                   STM2120
C ***************************************************************************STM2130
      PARAMETER NR=37,NC=40                                                STM2140
      DIMENSION A(NR,NC),INFT(2,3),IOFT(9,4)                               STM2150
      COMMON /MISC/ I0,J0                                                  STM2160
      DO 50 I=1,I0                                                         STM2170
      READ (LIN,INFT) (A(I,J),J=1,J0)                                      STM2180
   50 CONTINUE                                                             STM2190
      DO 70 I=1,I0                                                         STM2200
   70 WRITE (6,IOFT) I,(A(I,J),J=1,J0)                                     STM2210
   80 CONTINUE                                                             STM2220
      RETURN                                                               STM2230
      END                                                                  STM2240
C **********                                                               STM2330
      SUBROUTINE BICUBE(XMOL,P,COEFF,LIN)                                  STM2340
C ***************************************************************************STM2350
C * SUBROUTINE TO OBTAIN VALUES OF Ca,Cb,Cc COEFFICIENTS OF TABLE 29,  *STM2360
C * (POTTER AND BROWN, GEOLOGICAL SURVEY BULLETIN 1421-C) FOR GIVEN    *STM2370
C * VALUES OF MOLALITY AND PRESSURE. Ca,Cb,Cc VALUES COMPUTED FROM     *STM2380
C * BICUBIC USING COEFFICIENTS GENERATED FROM TABULATED VALUES OF      *STM2390
C * TABLE 29. ESTIMATES OF Ca,Cb,Cc FOR MOLALITY OF 0, AND PRESSURES UP*STM2400
C * TO 400 BARS PROVIDED BY R.J.SUN (WRITTEN COMMUNICATION,1981).      *STM2410
c *                                                                    *STM2420
C * FOR EACH SET OF TABLE VALUES (TABLE 2 OF THIS REPORT),             *STM2430
C * THERE ARE 12x4 CELLS. FOR EACH CELL, THERE ARE 16 BICUBIC COEF-    *STM2440
C * FICIENTS SO THAT THE BICUBIC COEFFICIENT ARRAYS ARE OF SIZE 48x16. *STM2450
C * IN SUBROUTINE BICUBE THE 16 COEFFICIENTS FOR EACH OF THE 4 COLUMN  *STM2460
C * CELLS OF A PARTICULAR ROW ARE READ AND PLACED IN ARRAY BC OF SIZE  *STM2470
C * 4x4x4. IF THE ROW OF THE COMPUTED MOLALITY CORRESPONDS TO THE ROW  *STM2480
C * OF THE BICUBIC COEFFICIENTS THEN THE 16 COEFFICIENTS CORRESPONDING *STM2490
C * TO THE COMPUTED PRESSURE CELL ARE PLACED IN ARRAY C OF SIZE 4x4.   *STM2500
C * THESE COEFFICIENTS ARE THEN USED IN THE BICUBIC POLYNOMIAL TOGETHER*STM2510
C * WITH THE COMPUTED MOLALITY AND PRESSURE TO OBTAIN THE INTERPOLATED *STM2520
C * TABLE VALUE. UPON RETURNING TO THE MAIN PROGRAM THE BICUBIC COEF-  *STM2530
C * FICIENT FILE IN QUESTION IS REWOUND.                               *STM2540
C ***************************************************************************STM2550
      DOUBLE PRECISION C                                                   STM2560
      DIMENSION BC(4,4,4),C(4,4)                                           STM2570
      ICOL=12                                                              STM2580
      JCOL=4                                                               STM2590
      YMOLE=XMOL*2.0                                                       STM2600
      II=IFIX(YMOLE)+1                                                     STM2610
```

```
      JJ=IFIX(P/100.)+1                                                    STM2620
      DO 69 I=1,ICOL                                                       STM2630
      DO 70 J=1,JCOL                                                       STM2640
   70 READ(LIN,60)((BC(M,N,J),M=1,4),N=1,4)                                STM2650
   60 FORMAT(10F8.4)                                                       STM2660
      IF(II.NE.I) GO TO 69                                                 STM2670
      DO 79 N=1,4                                                          STM2680
      DO 78 M=1,4                                                          STM2690
   78 C(M,N)=BC(M,N,JJ)                                                    STM2700
   79 CONTINUE                                                             STM2710
      GO TO 300                                                            STM2720
   69 CONTINUE                                                             STM2730
  300 CONTINUE                                                             STM2740
      IP=(IFIX(P/100.))*100                                                STM2750
      X=(P-IP)/100.                                                        STM2760
      Y=YMOLE-II+1                                                         STM2770
      X2=X*X                                                               STM2780
      X3=X2*X                                                              STM2790
      Y2=Y*Y                                                               STM2800
      Y3=Y2*Y                                                              STM2810
      COEFF= C(1,1)+C(1,2)*X+C(1,3)*X2+C(1,4)*X3+Y*(C(2,1)+C(2,2)*X+       STM2820
     1C(2,3)*X2+C(2,4)*X3)+Y2*(C(3,1)+C(3,2)*X+C(3,3)*X2+C(3,4)*X3)+       STM2830
     2Y3*(C(4,1)+C(4,2)*X+C(4,3)*X2+C(4,4)*X3)                             STM2840
      RETURN                                                               STM2850
      END                                                                  STM2860
```

APPENDIX D.--WATER-DENSITY MODIFICATIONS TO TRESCOTT-LARSON MODEL PROGRAM

```
*************************************************************************
*                                                                       *
*                  TRESCOTT MAIN PROGRAM--MODIFICATIONS                 *
*                       FOR VARIABLE DENSITY                            *
*                  ------------------------------------                 *
*                                                                       *
*************************************************************************
      DIMENSION Y(120000),L(30),HEADNG(33),NAME(54),INFT(2,4),     IOFT(MAN0100
     3T,4HOM E,4HLEVA,4HTION,2*4H    ,4H   R,4HECHA,4HRGE ,4HRATE,2*4H  MAN0240
     4  ,4H TR ,4HCOEF,4HFICI,4HENT ,2*4H    ,4H TC ,4HCOEF,4HFICI,4HENTMAN0241
     5 /                                                               MAN0242
      DATA INFT/4H(20F,4H4.0),4H(10F,4H8.0),4H(8F1,4H0.0),4H(8E1,4H0.4)/MAN0250
      ISUM=ISUM+ISIZ                                                   MAN1050
C PLACE STARTING HEADS AND OUTPUT FROM VARDEN PROGRAM CONSISTING OF:   MAN1381
C STORAGE COEFF.,TRANSMISSIVITY,VERTICAL LEAKANCE(TK),VARIABLE DENSITY MAN1382
C SOURCE TERM(PST),TR CONDUCTANCE,AND TC CONDUCTANCE IN INPUT FILE LIN. MAN1383
      LIN=11                                                           MAN1384
   80 CALL ARRAY(Y(LOC),INFT(1,2),IOFT(1,1),NAME(1),IRN,DUM,LIN)       MAN1410
   90 CALL ARRAY(Y(LOC),INFT(1,1),IOFT(1,2),NAME(7),IRN,DUM,LIN)       MAN1440
      CALL ARRAY(Y(LOC),INFT(1,4),IOFT(1,2),NAME(13),IRN,DUM,LIN)      MAN1500
  110 CALL ARRAY(Y(LOC),INFT(1,4),IOFT(1,3),NAME(19),IRN,DUM,LIN)      MAN1580
C   FOR VARIABLE DENSITY REPLACE TRESCOTT STATEMENTS MAN1620-MAN1640   MAN1611
C   WITH THE FOLLOWING STATEMENTS.                                     MAN1612
C    FOR VARIABLE DENSITY SET IQRE OPTION ON OPTIONS RECORD.           MAN1613
  130 IF(IQRE.NE.ICHK(7)) GO TO 141                                    MAN1614
C    READ IN VARIABLE SOURCE TERM (PST)                                MAN1615
      DO 142 K=1,KO                                                    MAN1616
      LOC=L(25)+(K-1)*NIJ                                              MAN1617
  142 CALL ARRAY(Y(LOC),INFT(1,4),IOFT(1,3),NAME(37),IRN,DUM,LIN)      MAN1618
C                                                                      MAN1619
C     READ IN TR COEFFICIENTS                                          MAN1620
C                                                                      MAN1621
      DO 143 K=1,KO                                                    MAN1622
      LOC=L(6)+(K-1)*NIJ                                               MAN1623
  143 CALL ARRAY(Y(LOC),INFT(1,4),IOFT(1,3),NAME(43),IRN,DUM,LIN)      MAN1624
C                                                                      MAN1625
C     READ IN TC COEFFICIENTS                                          MAN1626
C                                                                      MAN1627
      DO 144 K=1,KO                                                    MAN1628
      LOC=L(7)+(K-1)*NIJ                                               MAN1629
  144 CALL ARRAY(Y(LOC),INFT(1,4),IOFT(1,3),NAME(49),IRN,DUM,LIN)      MAN1631
  141 CONTINUE                                                         MAN1632
      CALL MDAT                                                        MAN1640
*************************************************************************
*                                                                       *
*           TRESCOTT SUBROUTINE DATAI--MODIFICATIONS                    *
*                FOR VARIABLE DENSITY                                   *
*                                                                       *
*************************************************************************
```

```
C MX=NUMBER OF MODEL ROWS,NX=NUMBER OF MODEL COLUMNS.                    DAT  71
      PARAMETER MX=37,NX=40                                              DAT  73
     3TOM(IP,JP),QRE(IO,JO,KO), TF(3), A(MX,NX), IN(6), IOFT(9), INFT(2)DAT 140
      ENTRY ARRAY(A,INFT,IOFT,IN,IRN,TF,LIN)                             DAT 710
C  THE ORDER OF VARDEN PROGRAM OUTPUT ARRAYS IS FROM BOTTOM ROW TO TOP   DAT 815
C  ROW, SO REVERSE ROW ORDER OF READ STATEMENT TO CONFORM TO TRESCOTT     DAT 816
C  CONVENTION.                                                          DAT 817
      DO 110 I=IO,1,-1                                                   DAT 820
      READ (LIN,INFT) (A(I,J),J=1,JO)                                    DAT 830
*************************************************************************
*                                                                       *
*         TRESCOTT SUBROUTINE CHECKI--MODIFICATIONS                     *
*              FOR VARIABLE DENSITY                                     *
*                                                                       *
*************************************************************************
     3W(NCH), QRE(IO,JO,KO)                                             CHK 130
C  NOTE THAT FOR VARIABLE DENSITY OPTION, COMPUTATION OF FLOW FOR NODES
C  WITH RELATIVE DENSITY OTHER THAN UNITY REQUIRES ADDITION OF APPRO-
C  PRIATE SOURCE TERM (ALPHA,BETA,OR GAMMA FROM PROGRAM VARDEN).
C  SUBROUTINE CHECKI WAS NOT MODIFIED TO COMPUTE FLOW WITH VARIABLE DEN-
C  SITY SO THAT THE FLOW BETWEEN CONSTANT HEAD NODES AND ADJACENT ACTIVE
C  NODES MAY NOT BE CORRECT. BECAUSE OF THIS THE PORTION OF THE MODEL
C  MASS BALANCE OUTPUT ATTRIBUTED TO FLOW FROM/TO CONSTANT HEAD NODES
C  AND FLOW FROM/TO THE TOP LAYER WILL BE APPROXIMATE. ACTUAL FLOW MAY
C  BE COMPUTED WITH A POST-MODEL PROCESSOR BY SAVING MODEL HEADS AND
C  USING THE TR,TC,TK CONDUCTANCES AND ALPHA,BETA,GAMMA SOURCE TERMS
C  FROM PROGRAM VARDEN.
C  ADD TOTAL SOURCE TERM TO PUMP OR CFLUX                             CHK1019
C     (K.EQ.KO.AND.IQRE.EQ.ICHK(7)) QREFLX=QREFLX+QRE(I,J,K)*AREA     CHK1020
      IF(QRE(I,J,K)) 155,158,157                                       CHK1021
  155 PUMP=PUMP+QRE(I,J,K)                                             CHK1022
      GO TO 158                                                        CHK1023
  157 CFLUX=CFLUX+QRE(I,J,K)                                           CHK1024
  158 CONTINUE                                                         CHK1025
*************************************************************************
*                                                                       *
*              TRESCOTT SUBROUTINE COEFF MODIFICATIONS                  *
*                   FOR VARIABLE DENSITY                                *
*                                                                       *
*************************************************************************
     3TOM(IP,JP),QRE(IO,JO,KO)                                          COF 130
C  BYPASS TR AND TC CONDUCTANCE CALCULATIONS IF VARIABLE DENSITY OPTION COF 512
C  IS USED.                                                             COF 513
      IF(IQRE.EQ.ICHK(7)) GO TO 333                                     COF 514
  333 CONTINUE                                                          COF 741
```

APPENDIX E.--MULTIAQUIFER-WELL MODIFICATIONS TO TRESCOTT-LARSON MODEL PROGRAM

```
************************************************************************
*                                                                      *
*                TRESCOTT MAIN PROGRAM--MODIFICATIONS                  *
*                     FOR MULTIAQUIFER WELL                            *
*                                                                      *
*                                                                      *
*                                                                      *
************************************************************************
      PARAMETER MW=365,MJ=8,MQ=4                                      MAN0091
     2H,IDK1,IDK2,IWATER,IQRE,IP,JP,IQ,JQ,IK,JK,K5,IPU1,IPU2,ITK,IEQN MAN0170
     3,IOPEN,IALOCA                                                   MAN0175
      COMMON/WIS/IL(MW),JL(MW),NWL(MW),KLL(MQ,MW,MJ),RWL(MW),QWL(MJ,   MAN0201
     1MW),C2(MJ,MW),C1(MQ,MW),NU(MJ,MW),NR(MQ,MW),PI2                 MAN0202
      DATA PI2/6.28318/                                               MAN0291
     1,IEQN,IOPEN,IALOCA                                              MAN0395
     1,IEQN,IOPEN,IALOCA                                              MAN0405
************************************************************************
*                                                                      *
*            TRESCOTT SUBROUTINE DATAI--MODIFICATIONS                  *
*                    FOR MULTIAQUIFER WELL                             *
*                                                                      *
*                                                                      *
************************************************************************
C  MW=NUMBER OF NODE SETS                                               *
C  MJ=MAXIMUM NUMBER OF MULTIAQUIFER CATEGORIES.                    DAT  72
C  FOR NORTHERN MIDWEST TOP LAYER IS CONSTANT HEAD SO THAT MQ=KO-1, DAT  73
C  OTHERWISE MQ IS NUMBER OF ACTIVE LAYERS.                         DAT  74
      PARAMETER MW=365,MJ=8,MQ=4                                   DAT  73
      COMMON/WIS/IL(MW),JL(MW),NWL(MW),KLL(MQ,MW,MJ),RWL(MW),QWL(MJ,DAT 241
     1MW),C2(MJ,MW),C1(MQ,MW),NU(MJ,MW),NR(MQ,MW),PI2              DAT 242
C  FOR MULTIAQUIFER WELL MODIFACTIONS REPLACE TRESCOTT STATEMENTS
C  DAT1740--DAT1820 OF ORIGINAL PROGRAM WITH THE FOLLOWING.
C     --- READ AND WRITE PUMPING RATES ---                          DAT1740
      DO 245 K=1,KO                                                 DAT1741
      DO 245 I=1,IO                                                 DAT1742
      DO 245 J=1,JO                                                 DAT1743
  245 WELL(I,J,K)=0.0                                               DAT1744
C******** LOOP TO 333 IF OPEN WELL ********                         DAT1745
      IF(IOPEN.EQ.ICHK(12)) GO TO 333                               DAT1746
      WRITE(6,410) NWEL                                             DAT1747
      IF(NWEL.EQ.0) GO TO 260                                       DAT1748
      DO 270 II=1,NWEL                                              DAT1749
      READ(5,330) K,I,J,WELL(I,J,K)                                 DAT1750
  270 WELL(I,J,K)=WELL(I,J,K)/(DELX(J)*DELY(I))                     DAT1751
      GO TO 260                                                     DAT1758
  333 CONTINUE                                                      DAT1759
      KK=MQ                                                         DAT1760
      DO 250 I=1,NWEL                                               DAT1761
      READ(11,500) IL(I),JL(I),JA,((KLL(K,I,J),K=1,KK),QWL(J,I),J=1,JA) DAT1762
C     WRITE(6,501) IL(I),JL(I),JA,((KLL(K,I,J),K=1,KK),QWL(J,I),J=1,JA) DAT1763
  501 FORMAT(1X,2I2,4X,I2,8(4I1,F6.3))                              DAT1764
      NWL(I)=JA                                                     DAT1765
```

```
C   COMPUTE ESTIMATE OF NO. OF WELLS PER MULTIAQUIFER COMBINATION       DAT1766
      AVPUMP=0.1                                                        DAT1767
      DO 692 JJ=1,JA                                                    DAT1768
      IF(QWL(JJ,I).EQ.0.0) NU(JJ,I)=1                                   DAT1769
      IF(QWL(JJ,I).EQ.0.0) GO TO 692                                    DAT1770
      NU(JJ,I)=QWL(JJ,I)/AVPUMP                                         DAT1771
      IF(NU(JJ,I).EQ.0) NU(JJ,I)=1                                      DAT1772
  692 CONTINUE                                                          DAT1773
C                                                                       DAT1774
C   COMPUTE NO. OF WELLS TAPPING LAYER K(NR(K,I))                       DAT1775
C   I.E. NR(K,I) CONTAINS THE TOTAL NUMBER OF WELLS OPEN TO LAYER K     DAT1776
C   OF NODE SET I.                                                      DAT1777
C                                                                       DAT1778
      DO 688 K=1,KK                                                     DAT1779
      NRSUM=0                                                           DAT1780
      DO 687 JJ=1,JA                                                    DAT1781
  687 IF(KLL(K,I,JJ).NE.0) NRSUM=NRSUM+NU(JJ,I)                         DAT1782
      NR(K,I)=NRSUM                                                     DAT1783
  688 CONTINUE                                                          DAT1784
C                                                                       DAT1785
C     FOR THE NORTHERN MIDWEST THE PUMPING RATE FILE HAS POSITIVE VALUESDAT1786
C     FOR DISCHARGE. HENCE SIGN OF PUMPING RATES IS CHANGED ON INPUT.   DAT1787
C     IN ADDITION, BECAUSE VARIABLE DENSITY WAS SIMULATED PUMPING RATES DAT1788a
C     ARE MULTIPLIED BY DENSITY OF PURE WATER.                          DAT1788b
      DO 7356 J=1,JA                                                    DAT1789
 7356 QWL(J,I)=-1.5473*QWL(J,I)*1.94                                    DAT1790
      DX=DELX(JL(I))                                                    DAT1791
      DY=DELY(IL(I))                                                    DAT1792
C     IF VARIABLE DENSITY OPTION IS USED THEN DX=DY=1 FOR ALL NODES.    DAT1793
C     TO COMPUTE EFFECTIVE RADIUS (REL) THE AREA OF THE BLOCK IS NEEDED.DAT1794
C     BECAUSE A UNIFORM NODE SPACING WAS USED IN NORTHERN MIDWEST(16X16 DAT1795
C     MILES) AREA TERM BELOW IS EXPLICITLY DEFINED. IF A NON-UNIFORM    DAT1796a
C     SPACING IS USED AND VARIABLE DENSITY IS ALSO BEING SIMULATED,PRO- DAT1796b
C     GRAM MUST BE MODIFIED TO ALLOW COMPUTATION OF AREA.               DAT1796c
C     IN GENERAL, AREA=MIN(DX,DY)**2.                                   DAT1796d
      AREA=84480.*84480.                                                DAT1797
      RWL(I)=0.5                                                        DAT1798
C   COMPUTE EFFECTIVE RADIUS FOR EACH LAYER USING NR(K,I)               DAT1799
      DO 686 K=1,KK                                                     DAT1800
      IF(NR(K,I).EQ.0) GO TO 686                                        DAT1801
      REL=SQRT(0.04322*AREA/NR(K,I))                                    DAT1802
      C1(K,I)=1.0/(ALOG(REL/RWL(I)))                                    DAT1803
  686 CONTINUE                                                          DAT1804
      DO 620 J=1,JA                                                     DAT1805
      C2(J,I)=0.0                                                       DAT1806
      KK=MQ                                                             DAT1807
      DO 520 K=1,KK                                                     DAT1808
      KL=KLL(K,I,J)                                                     DAT1809
      IF(KL.EQ.0) GO TO 520                                             DAT1810
      C2(J,I)=C2(J,I)+T(IL(I),JL(I),KL)*C1(KL,I)                        DAT1811
```

```
  520 CONTINUE                                                          DAT1812
      C2(J,I)=1.0/C2(J,I)                                               DAT1813
  620 CONTINUE                                                          DAT1814
      DO 630 K=1,KK                                                     DAT1815
      SUMZ=0.0                                                          DAT1816
      DO 530 J=1,JA                                                     DAT1817
      KL=KLL(K,I,J)                                                     DAT1818
      IF(KL.EQ.0) GO TO 530                                             DAT1819
      SUMZ=SUMZ+QWL(J,I)*C2(J,I)                                        DAT1820
  530 CONTINUE                                                          DAT1821
      WELL(IL(I),JL(I),K) = T(IL(I),JL(I),K)*SUMZ*C1(K,I)               DAT1822
      WELL(IL(I),JL(I),K)= WELL(IL(I),JL(I),K)/(DX*DY)                  DAT1823
  630 CONTINUE                                                          DAT1824
  250 CONTINUE                                                          DAT1825
  500 FORMAT(1X,2I2,I6,8(4I1,F6.0))                                     DAT1826
  510 FORMAT(8F10.0)                                                    DAT1827
  260 RETURN                                                            DAT1828
*************************************************************************
*                                                                       *
*         TRESCOTT SUBROUTINE CHECKI--MODIFICATIONS                     *
*                  FOR MULTIAQUIFER WELL                                *
*                                                                       *
*************************************************************************
      PARAMETER MW=365,MJ=8,MQ=4                                        CHK  91
      DIMENSION  HW(MJ),HZ(MJ)                                          CHK  92
     3,IOPEN,IALOCA                                                     CHK 171
      COMMON/WIS/IL(MW),JL(MW),NWL(MW),KLL(MQ,MW,MJ),RWL(MW),QWL(MJ,    CHK 201
     1MW),C2(MJ,MW),C1(MQ,MW),NU(MJ,MW),NR(MQ,MW),PI2                   CHK 202
      IF(IOPEN.NE.ICHK(12)) GO TO 333                                   CHK1541
      WRITE(6,1111)                                                     CHK1543
      DO 300 I=1,NWEL                                                   CHK1544
      J=JL(I)                                                           CHK1545
      II=IL(I)                                                          CHK1546
C   COMPUTE HEAD IN WELL  FOR EACH OF THE "JA" AQUIFERS                 CHK1547
      JA=NWL(I)                                                         CHK1548
      DO 310 JJ=1,JA                                                    CHK1549
      SUMZ=0.0                                                          CHK1550
      KK=MQ                                                             CHK1550
      DO 320 L=1,KK                                                     CHK1552
      K=KLL(L,I,JJ)                                                     CHK1553
      IF(K.EQ.0) GO TO 320                                              CHK1554
      SUMZ=SUMZ+T(II,J,K)*PHI(II,J,K)*C1(K,I)                           CHK1555
  320 CONTINUE                                                          CHK1556
C  HW IS THE COMPOSITE HEAD WITH PUMPING FOR EACH CATEGORY WELL.        CHK1558
C  HZ IS THE COMPOSITE HEAD WITH NO PUMPING FOR EACH CATEGORY WELL.     CHK1559
C                                                                       CHK1561
      HW(JJ)=SUMZ*C2(JJ,I)+QWL(JJ,I)*C2(JJ,I)/(NU(JJ,I)*PI2)           CHK1562
      HZ(JJ)=SUMZ*C2(JJ,I)                                              CHK1563
  311 CONTINUE                                                          CHK1564
  310 CONTINUE                                                          CHK1565
```

```
C     WRITE(6,1118)I,II,J,((KLL(L,I,JJ),L=1,KK),QWL(JJ,I),JJ=1,JA)      CHK1566
 1118 FORMAT(1X,2('*'),' NODE SET',I4,' MODEL NODE(',I2,',',I2,')',2('*'CHK1567
     1),7(1X,4I1,F8.3)/3(1X,4I1,F8.3))                                  CHK1568
C     WRITE(6,1117)(HW(JJ),HZ(JJ),JJ=1,JA)                              CHK1569
 1117 FORMAT(36X,7(F7.0,F6.0)/1(F7.0,F6.0))                             CHK1570
C     WRITE COMPOSITE HEAD ON UNIT 13                                   CHK1570a
      WRITE(13,1115) II,J,JA,((KLL(L,I,JJ),L=1,KK),HZ(JJ),JJ=1,JA)      CHK1571
 1115 FORMAT(2I2,1X,I2,1X,8(4I1,F6.0))                                  CHK1572
 1113 FORMAT(3X,'PUMPING RATE (',I2,',',I2,')',10F8.4)                  CHK1573
 1112 FORMAT(3X,'HEAD IN WELL (',I2,',',I2,')',10F8.2)                  CHK1574
      IF(IALOCA.NE.ICHK(13)) GO TO 300                                  CHK1575
C     BYPASS ALOCAT EXCEPT FOR LAST PUMPING PERIOD                      CHK1576
      IF(KP.LT.NPER) GO TO 300                                          CHK1577
      CALL ALOCAT(PHI,OLD,T,S,TK,DELX,DELY,DELZ,IO,JO,KO,K5,HW,Q,NS1,I, CHK1578
     1II,J,TC,TR,QRE)                                                   CHK1579
  300 CONTINUE                                                          CHK1580
  333 CONTINUE                                                          CHK1581
 1111 FORMAT(1HO,'CALCULATED "WELL HEADS" FOR GIVEN PUMPING RATES')     CHK1582
*************************************************************************
*                                                                       *
*           TRESCOTT SUBROUTINE PRNTA--MODIFICATIONS                    *
*                    FOR MULTIAQUIFER-WELL                              *
*                                                                       *
*************************************************************************
     3,IOPEN,IALOCA                                                     PRN 151
     1UN2','ITKR','EQN3','OPEN','ALOC'/                                 BLK 140
```

APPENDIX F.--SUBROUTINES SOLVE AND BAND

```
      SUBROUTINE SOLVE(PHI,STRT,OLD,T,S,TR,TC,TK,WELL,DELX,DELY,DELZ,FACSP30010
     1T,NP,RHS,DH,BB,HH,TEST3,QRE)                                      SP30020
      REAL*8 SUM1,SUM2,SUM3                                             SP30030
      REAL *8PHI,RHO,B,D,F,H,Z,SU,RES,SUPH,ZPH,E                        SP30040
C     ------------------------------------------------------------------SP30050
C     SOLUTION BY THE SLICE-SUCCESSIVE-OVER-RELAXATION METHOD           SP30060
C     ------------------------------------------------------------------SP30070
C ******************************************************************0000000
C * ORIGINAL PROGRAM STATEMENTS AS WRITEN BY S.P.LARSON AND C.W.FAUST  ****0071
C * ARE THOSE THAT DO NOT CONTAIN ***, 000 OR $$$ IN COLUMNS 73-75.    ****0072
C * MODIFICATIONS TO IMPLEMENT THE EFFECTS OF MULTIAQUIFER WELLS       ****0073
C * CONSIST OF: STATEMENTS IN THIS SUBROUTINE THAT CONTAIN ASTERISKS   ****0074
C * IN COLUMNS 73-75, SUBROUTINE MULT TO DEVELOP ADDITIONAL TERMS IN   ****0075
C * THE COEFFICIENT MATRIX ARISING FROM THE CONNECTION BETWEEN LAYERS  ****0076
C * OPEN TO A WELL, AND SUBROUTINE RESI TO DEVELOP TERMS FOR THE RIGHT ****0077
C * HAND SIDE VECTOR OF THE SOLUTION EQUATION.                         ****0078
C * MODIFICATIONS FOR VARIABLE DENSITY CONSIST OF STATEMENTS WITH $$$  *$$$0079
C * IN COLUMNS 73-75                                                  *$$$0000
C ******************************************************************0000000
C                                                                       SP30080
C     SPECIFICATIONS:                                                   SP30090
C                                                                       SP30100
C                                                                       SP30105
C        DIMENSIONING OF ARRAYS IS AS FOLLOWS:                          ***0106
C     'A' ARRAY=NUMBER OF ACTIVE MODEL NODES*(KO+1)                     ***0107
C      RSMAL=(JO-2)*KO                                                  ***0108
C      JPOS=JO                                                          ***0109
C      KPOS=KO                                                          ***0110
C      NB=(IO-1)                                                        ***0111
C      ICH=IO                                                           ***0112
C      AA=(IO-2)                                                        ***0113
      PARAMETER MW=365,MJ=8,MQ=4                                        ***0114
      DIMENSION A(26000),RSMAL(190),NB(36),JPOS(40),KPOS(5)             SP30120
      DIMENSION AA(70),ICH(37)                                          SP30130
C                                                                       SP30140
      COMMON /INTEGR/ IO,JO,KO,I1,J1,K1,IX,JX,KX,NPER,KTH,ITMAX,LENGTH,KSP30150
     1P,N                                                               SP30160
     1WEL,NUMT,IFINAL,IT,KT,IHEAD,IDRAW,IFLO,IERR,I2,J2,K2,IMAX,ITMX1,NCSP30170
     2H,IDK1,IDK2,IWATER,IQRE,IP,JP,IQ,JQ,IK,J5,K5,IPU1,IPU2,ITK,IEQN   SP30180
     3,IOPEN,IALOCA                                                     ***0181
      COMMON /SPARAM/ TMAX,CDLT,DELT,ERR,TEST,SUM,SUMP,QR               SP30190
      COMMON /SARRAY/ ICHK(13),LEVEL1(9),LEVEL2(9)                      SP30200
      COMMON/WIS/IL(MW),JL(MW),NWL(MW),KLL(MQ,MW,MJ),RWL(MW),QWL(MJ,    ***0201
     1MW),C2(MJ,MW),C1(MQ,MW),NU(MJ,MW),NR(MQ,MW),PI2                   ***0202
      DIMENSION PHI(1), STRT(1), OLD(1), T(1), S(1), TR(1), TC(1), TK(1)SP30210
     1, WELL(1), DELX(1), DELY(1), DELZ(1), FACT(KO,3),           TEST3(SP30220
     21),NP(1),RHS(1),DH(1),BB(1),HH(1),QRE(1)                          SP30230
      RETURN                                                            SP30240
C     ..................................................................SP30250
CAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA0000000
```

```
C     *********************                                          SP30260
      ENTRY ITER                                                     SP30270
C     *********************                                          SP30280
C    ENTRY POINT ITER ENTERED ONCE PER SIMULATION                    0000000
      NIJ=I0*J0                                                      SP30290
      NIJK=NIJ*K0                                                    SP30300
      DO 5 I=1,NIJK                                                  SP30310
      DH(I)=0.                                                       SP30320
    5 NP(I)=0                                                        SP30330
C     SET UP PROCEEDURE TO NUMBER EQUATIONS                          0000000
      DO 6 K=1,K0                                                    SP30340
    6 KPOS(K)=(K-1)*NIJ                                              SP30350
      DO 7 J=1,J0                                                    SP30360
    7 JPOS(J)=(J-1)*I0                                               SP30370
      NB(1)=0                                                        SP30380
      MBW=K0+1                                                       SP30390
      NEQ=0                                                          SP30400
      ICH(I0)=0                                                      SP30410
C     IDENTIFY ACTIVE MODEL NODES, IDENTIFY SLICES WITH CONSTANT HEADS 0000000
C     (ARRAY ICH),AND COUNT THE TOTAL NUMBER OF EQUATIONS FOR ALL    0000000
C     SLICES(NEQ).                                                   0000000
      DO 11 I=2,I1                                                   SP30420
      ICH(I)=0                                                       SP30430
      NN=0                                                           SP30440
      DO 10 J=2,J1                                                   SP30450
      DO 10 K=1,K0                                                   SP30460
      N=I+JPOS(J)+KPOS(K)                                            SP30470
      IF(S(N).LT.0.) ICH(I)=1                                        SP30480
      IF(T(N).EQ.0..OR.S(N).LT.0.) GO TO 10                          SP30490
      NN=NN+1                                                        SP30500
      NP(N)=NN                                                       SP30510
      NEQ=NEQ+1                                                      SP30520
   10 CONTINUE                                                       SP30530
   11 NB(I)=NEQ                                                      SP30540
      NEQBW=NEQ*MBW                                                  SP30550
      PRINT 12,NEQBW                                                 SP30560
   12 FORMAT(49H0********* WARNING**ARRAY A MUST BE DIMENSIONED >,I10) SP30570
C     READ ACCELERATION PARAMETER WO.                                0000000
      READ 13,WO                                                     SP30580
   13 FORMAT(F10.0)                                                  SP30590
      PRINT 14,WO,LENGTH                                             SP30600
   14 FORMAT(28H ACCELERATION PARAMETER, WO=,F10.3,/,                SP30610
     $ 49H NUMBER OF ITERATIONS BETWEEN WATT'S CORRECTION =,I5)      SP30620
      RETURN                                                         SP30630
CAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA0000000
C     *********************                                          SP30640
      ENTRY NEWITA                                                   SP30650
C     *********************                                          SP30660
C     ENTRY POINT NEWITA ENTERED EACH TIME STEP.                     0000000
C     ---COMPUTE TRANSMISSIVITY AND T COEFFICIENTS FOR UPPER         SP30670
C     HYDROLOGIC UNIT WHEN IT IS UNCONFINED---                       SP30680
```

```
      IF (IWATER.NE.ICHK(6)) GO TO 35                                    SP30690
      CALL TRANS(0)                                                      SP30700
   35 CONTINUE                                                           SP30710
      IR=0                                                               SP30720
      ID=1-MBW                                                           SP30730
      DO 36 I=1,NEQBW                                                    SP30740
   36 A(I)=0.                                                            SP30750
      RESO=0.0                                                           ***0760
C  CONSTRUCT COEFFICIENTS AND LOAD INTO A AND RHS                        SP30770
C ***************************************************************************0000000
C ********** CALL SUBROUTINE MULT FOR MULTIAQUIFER OPTION *********     ***0000
      IF(IOPEN.NE.ICHK(12)) GO TO 333                                    ***0771
      CALL MULT(NWEL,A,NB,MBW,JPOS,KPOS,T,NP,KO)                         ***0772
      IWELL=1                                                            ***0773
  333 CONTINUE                                                           ***0774
C ***************************************************************************0000000
CBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB0000000
C   START SLICES TO DEVELOP THE B,D,E,F,H,SU,Z COEFFICIENTS AND LOAD     0000000
C   THEM INTO 'A' ARRAY. ALSO DEVELOP THE RHS PART OF THE RIGHT HAND     0000000
C   SIDE OF THE EQUATIONS.                                               0000000
      DO 60 I=2,I1                                                       SP30780
      DO 47 J=2,J1                                                       SP30790
      IFLAG=0                                                            ***0791
      DO 48 K=1,KO                                                       SP30800
      N=I+JPOS(J)+KPOS(K)                                                SP30810
      IF(NP(N).EQ.0) GO TO 48                                            SP30820
      NIA=N+1                                                            SP30830
      NIB=N-1                                                            SP30840
      NJA=N+IO                                                           SP30850
      NJB=N-IO                                                           SP30860
      NKA=N+NIJ                                                          SP30870
      NKB=N-NIJ                                                          SP30880
      VX=DELX(J)                                                         SP30890
      VY=DELY(I)                                                         SP30900
      VZ=VX*VY                                                           SP30910
      VQ=VZ                                                              SP30920
      IF(IEQN.NE.ICHK(11)) GO TO 123                                     SP30930
      DZ=DELZ(K)                                                         SP30940
      VX=VX*DZ                                                           SP30950
      VY=VY*DZ                                                           SP30960
      VQ=VQ*DZ                                                           SP30970
  123 D=TR(NJB)*VY                                                       SP30971
      F=TR(N)*VY                                                         SP30972
      B=TC(NIB)*VX                                                       SP30973
      H=TC(N)*VX                                                         SP30974
      SU=0.DO                                                            SP30980
      Z=0.DO                                                             SP30990
      SUPH=0.0                                                           SP31000
      ZPH=0.0                                                            SP31010
      IF(K.EQ.1) GO TO 124                                               SP31020
      IF(T(N).EQ.0.0) TK(NKB)=0.0                                        ***1030
      Z=TK(NKB)*VZ                                                       SP31031
```

```
      ZPH=Z*PHI(NKB)                                                    SP31040
  124 IF(K.EQ.KO) GO TO 125                                             SP31050
      IF(T(N).EQ.0.0) TK(N)=0.0                                         ***1060
      SU=TK(N)*VZ                                                       SP31070
      SUPH=SU*PHI(NKA)                                                  SP31080
  125 RHO=S(N)/DELT                                                     SP31090
      QR=0.                                                             SP31120
C ***********************************************************************$$$1121
C *              VARIABLE-DENSITY MODIFICATIONS                        *$$$1122
C *        EXPAND NUMBERING ON QRE TERM TO INCLUDE ALL LAYERS          *$$$1123
C ***********************************************************************$$$1124
C     IF(K.NE.KO) GO TO 42                                              $$$1125
      IF(IQRE.EQ.ICHK(7)) QR=QRE(I+JPOS(J)+KPOS(K))                     $$$1126
   42 CONTINUE                                                          SP31130
C ***********************************************************************0000000
      E=-B-D-F-H-SU-Z-RHO*VQ                                            SP31131
C     RES=RHS IS RIGHT HAND SIDE OF EQUATION FOR TIME STEP AND USES     0000000
C     HEADS FROM PREVIOUS TIME STEP.                                    0000000
      RES=-B*PHI(NIB)-D*PHI(NJB)-E*PHI(N)-F*PHI(NJA)-H*PHI(NIA)-SUPH-ZPHSP31132
     1+(-WELL(N) +(-RHO*OLD(N)-QR))*VQ                                  SP31133
      IR=IR+1                                                           SP31140
      ID=ID+MBW                                                         SP31150
      NN=NP(N)                                                          SP31160
      A(ID)=E+A(ID)                                                     ***1161
      IF(K.EQ.KO) GO TO 44                                              SP31170
      IF(NP(NKA).EQ.0) GO TO 44                                         SP31180
      A(ID+1)=SU+A(ID+1)                                                ***1181
   44 NJ1=NP(NJA)                                                       SP31190
      IF(NJ1.EQ.0) GO TO 46                                             SP31200
      A(ID+NJ1-NN)=F+A(ID+NJ1-NN)                                       ***1210
   46 RHS(IR)=RES                                                       SP31220
      BB(IR)=B                                                          SP31230
      HH(IR)=H                                                          SP31240
C ***********************************************************************0000
C CALL SUBROUTINE RESI FOR MULTI-AQUIFER OPTION ONLY IF ROW,COLUMN OF  ****0000
C * DO LOOP HAS A CORRESPONDING MULTI-AQUIFER NODE SET.                 ****0000
C * FOR NORTHERN MIDWEST STUDY TOP LAYER IS CONSTANT HEAD SO BY-PASS    ****0000
C * LAYER KO. FOR GENERAL APPLICATION LAYER KO MAY BE INCLUDED.         ****0000
      IF(IOPEN.NE.ICHK(12)) GO TO 48                                    ***1241
      IF(K.EQ.KO) GO TO 48                                              ***1242
      IF(I.EQ.IL(IWELL).AND.J.EQ.JL(IWELL)) GO TO 4777                  ***1243
      GO TO 48                                                          ***1244
 4777 CALL RESI(IO,JO,KO,PHI,T,I,J,K,RESO,IWELL,IFLAG)                  ***1245
      RHS(IR)=RHS(IR)-RESO                                              ***1246
   48 CONTINUE                                                          ***1247
      IF(IFLAG.NE.0) IWELL=IWELL+1                                      ***1248
C ***********************************************************************0000000
   47 CONTINUE                                                          SP31249
      IAO=NB(I-1)*MBW+1                                                 SP31250
      NEQS=NB(I)-NB(I-1)                                                SP31260
      IF(NEQS.LT.2) GO TO 60                                            SP31270
```

```
C  DECOMPOSE A MATRIX FOR SLICE                                          SP31280
      CALL BAND(A(IAO),RHS,NEQS,MBW,1)                                   SP31290
   60 CONTINUE                                                           SP31300
CBBBBBBBBBBBB END OF SLICE LOOP FOR 'A' MATRIX AND RHS TERM BBBBBBBBBBBB0000000
CCCCCCCCCCCCCCCCCCCCCCCCCCCC ITERATION LOOP CCCCCCCCCCCCCCCCCCCCCCCCCCCC0000000
      ITT=-1                                                             SP31310
C  BEGIN ITERATIONS                                                      SP31320
C     SOLVE FOR HEAD DIFFERENCE AFTER PLACING B AND H COEFFICIENTS ON     0000000
C     RIGHT SIDE                                                         0000000
      DO 120 IT=1,ITMX1                                                  SP31330
      ITT=ITT+1                                                          SP31340
      TEST3(IT)=0.0                                                      SP31350
      TEST=0.0                                                           SP31360
      BIG=0.                                                             SP31370
      DO 110 I=2,I1                                                      SP31380
      NBI1=NB(I-1)                                                       SP31390
C  COMPLETE RHS TERMS                                                    SP31400
      DO 90 J=2,J1                                                       SP31410
      DO 90 K=1,K0                                                       SP31420
      N=I+JPOS(J)+KPOS(K)                                                SP31430
      IR=NP(N)                                                           SP31440
      IF(IR.EQ.0) GO TO 90                                               SP31450
      IRT=IR+NBI1                                                        SP31460
      RSMAL(IR)=RHS(IRT)-BB(IRT)*DH(N-1)-HH(IRT)*DH(N+1)                 SP31470
   90 CONTINUE                                                           SP31480
      IAO=NBI1*MBW+1                                                     SP31490
      NEQS=NB(I)-NBI1                                                    SP31500
      IF(NEQS.LT.2) GO TO 95                                             SP31510
C  SOLVE FOR DH FOR SLICE                                                SP31520
      CALL BAND(A(IAO),RSMAL,NEQS,MBW,2)                                 SP31530
      GO TO 96                                                           SP31540
   95 IF(NEQS.LT.1) GO TO 110                                            SP31550
      RSMAL(1)=RSMAL(1)/A(IAO)                                           SP31560
   96 CONTINUE                                                           SP31570
C  COMPLETE OVERRELAXATION                                               SP31580
      DO 100 J=2,J1                                                      SP31590
      DO 100 K=1,K0                                                      SP31600
      N=I+JPOS(J)+KPOS(K)                                                SP31610
      NN=NP(N)                                                           SP31620
      IF(NN.EQ.0) GO TO 100                                              SP31630
      DIFF=WO*(RSMAL(NN)-DH(N))                                          SP31640
      DH(N)=DH(N)+DIFF                                                   SP31650
      TCHK=ABS(DIFF)                                                     SP31660
      IF(TCHK.GT.ABS(BIG)) BIG=DIFF                                      SP31670
  100 CONTINUE                                                           SP31680
  110 CONTINUE                                                           SP31690
C ***************************************************************************0000000
C * ALL SLICES FOR THIS ITERATION ARE COMPLETED. NOW TEST FOR CLOSURE. *0000000
C * BIG IS LARGEST CHANGE IN HEAD DIFFERENCES AND ERR IS CLOSURE.      *0000000
C ***************************************************************************0000000
```

```
C  TEST TERMINATION CONDITIONS                                          SP31700
      IF(ABS(BIG).GT.ERR) TEST=1.                                       SP31710
      TEST3(IT)=BIG                                                     SP31720
      IF (TEST.EQ.0.) GO TO 130                                         SP31730
   22 IF(MOD(IT,LENGTH).NE.0) GO TO 120                                 SP31740
C ***********************************************************************0000000
C          COMPUTE WATT'S CORRECTION                                    SP31750
C NOTE:  IF THERE ARE ANY CONSTANT HEAD NODES, IN A SLICE, CORRECTION   SP31751
C IS NOT APPLIED. IN THAT TOP LAYER OF NORTHERN-MIDWEST MODEL           SP31752
C CONSISTS OF CONSTANT HEAD NODES WATT'S CORRECTION WAS NOT USED.       SP31753
      ID=-1                                                             SP31760
      NEQS=0                                                            SP31770
      DO 112 I=2,I1                                                     SP31780
      NBI1=NB(I-1)                                                      SP31790
      IF(NB(I).EQ.NBI1.OR.ICH(I).EQ.1) GO TO 112                        SP31800
      SUM1=0.                                                           SP31810
      SUM2=0.                                                           SP31820
      SUM3=0.                                                           SP31830
      DO 111 J=2,J1                                                     SP31840
      DO 111 K=1,K0                                                     SP31850
      N=I+JPOS(J)+KPOS(K)                                               SP31860
      NN=NP(N)                                                          SP31870
      IF(NN.EQ.0) GO TO 111                                             SP31880
      IRT=NN+NBI1                                                       SP31890
      NIA=N+1                                                           SP31900
      NIB=N-1                                                           SP31910
      VX=DELX(J)                                                        SP31920
      VY=DELY(I)                                                        SP31930
      VQ=VX*VY                                                          SP31940
      IF(IEQN.NE.ICHK(11)) GO TO 23                                     SP31950
      DZ=DELZ(K)                                                        SP31960
      VX=VX*DZ                                                          SP31970
      VQ=VQ*DZ                                                          SP31980
   23 B=TC(NIB)*VX                                                      SP31990
      H=TC(N)*VX                                                        SP32000
   25 RHO=S(N)/DELT                                                     SP32010
      E=-B-H-RHO*VQ                                                     SP32020
      SUM1=SUM1+E                                                       SP32030
      RES=RHS(IRT)-B*DH(NIB)-E*DH(N)-H*DH(NIA)                          SP32040
      SUM3=SUM3+RES                                                     SP32050
      SUM2=SUM2+H                                                       SP32060
  111 CONTINUE                                                          SP32070
      NEQS=NEQS+1                                                       SP32080
      ID=ID+2                                                           SP32090
      AA(ID)=SUM1                                                       SP32100
      AA(ID+1)=SUM2                                                     SP32110
      IF(ICH(I+1).EQ.1) AA(ID+1)=0.                                     SP32120
      RSMAL(NEQS)=SUM3                                                  SP32130
  112 CONTINUE                                                          SP32140
      IF(NEQS.EQ.0) GO TO 120                                           SP32150
      IF(NEQS.LT.2) GO TO 115                                           SP32160
```

```
      CALL BAND(AA,RSMAL,NEQS,2,1)                                      SP32170
      CALL BAND(AA,RSMAL,NEQS,2,2)                                      SP32180
      GO TO 116                                                         SP32190
  115 RSMAL(1)=RSMAL(1)/AA(1)                                           SP32200
  116 NEQS=0                                                            SP32210
      DO 114 I=2,I1                                                     SP32220
      NBI1=NB(I-1)                                                      SP32230
      IF(NB(I).EQ.NBI1.OR.ICH(I).EQ.1) GO TO 114                        SP32240
      NEQS=NEQS+1                                                       SP32250
      DO 113 J=2,J1                                                     SP32260
      DO 113 K=1,K0                                                     SP32270
      N=I+JPOS(J)+KPOS(K)                                               SP32280
      IF(NP(N).EQ.0) GO TO 113                                          SP32290
      DH(N)=DH(N)+RSMAL(NEQS)                                           SP32300
  113 CONTINUE                                                          SP32310
  114 CONTINUE                                                          SP32320
C  ***************** END OF WATT'S CORRECTION SEGMENT OF CODE **********0000000
  120 CONTINUE                                                          SP32330
CCCCCCCCCCCCCCCCCCCCCC END ITERATION LOOP CCCCCCCCCCCCCCCCCCCCCCCCCCCCC0000000
  130 CONTINUE                                                          SP32340
C   FORM FINAL HEADS FOR THIS TIME STEP BY ADDING HEAD DIFFERENCES TO   0000000
C   HEADS OF PREVIOUS TIME STEP.                                        0000000
      DO 140 I=2,I1                                                     SP32350
      DO 140 J=2,J1                                                     SP32360
      DO 140 K=1,K0                                                     SP32370
      N=I+JPOS(J)+KPOS(K)                                               SP32380
      PHI(N)=PHI(N)+DH(N)                                               SP32390
  140 CONTINUE                                                          SP32400
      IT =ITT                                                           SP32410
      IF(TEST.EQ.0.) RETURN                                             SP32420
      IT=ITT+1                                                          SP32430
     WRITE(6,220)                                                       SP32440
      CALL OUTPUT                                                       SP32450
  150 RETURN                                                            SP32460
  220 FORMAT ('0EXCEEDED PERMITTED NUMBER OF ITERATIONS'/' ',39('*'))   SP32470
      END                                                               SP32480
      SUBROUTINE BAND (A,B,N,IHB,KKK)                                   BND0011
      DIMENSION A(1),B(1)                                               BND0012
      NM1=N-1                                                           BND0013
      IHB1=IHB-1                                                        BND0014
      IF(KKK.EQ.2) GO TO 30                                             BND0015
C DECOMPOSE A                                                           BND0016
C                                                                       BND0017
      ID=1-IHB                                                          BND0018
      DO 20   I=1,NM1                                                   BND0019
      ID=ID+IHB                                                         BND0020
      C1=1./A(ID)                                                       BND0021
      LD=ID                                                             BND0022
      L=I                                                               BND0023
      DO 15 J=1,IHB1                                                    BND0024
      L=L+1                                                             BND0025
      LD=LD+IHB                                                         BND0026
      IF(L.GT.N) GO TO 20                                               BND0027
```

```
      IB=ID+J                                                    BND0028
      C=A(IB)*C1                                                 BND0029
      LB =LD-1                                                   BND0030
      DO 10 K=J,IHB1                                             BND0031
      LB=LB+1                                                    BND0032
   10 A(LB)=A(LB)-C*A(ID+K)                                      BND0033
      A(IB)=C                                                    BND0034
   15 CONTINUE                                                   BND0035
   20 CONTINUE                                                   BND0036
      GO TO 80                                                   BND0037
   30 CONTINUE                                                   BND0038
C MODIFY RHS                                                     BND0039
      ID=1-IHB                                                   BND0040
      DO 50 I=1,NM1                                              BND0041
      ID=ID+IHB                                                  BND0042
      C=B(I)                                                     BND0043
      B(I)=C/A(ID)                                               BND0044
      IRBW=MIN0 (IHB1,N-I)                                       BND0045
      DO 50 L=1,IRBW                                             BND0046
      K=I+L                                                      BND0047
   50 B(K)=B(K)-A(ID+L)*C                                        BND0048
      ID=ID+IHB                                                  BND0049
      B(N)=B(N)/A(ID)                                            BND0050
C BACK SUBSTITUTE                                                BND0051
      DO 70 I=1,NM1                                              BND0052
      ID=ID-IHB                                                  BND0053
      L=N-I                                                      BND0054
      SUM=0.                                                     BND0055
      IRBW=MIN0(IHB1,N-L)                                        BND0056
      DO 60 J=1,IRBW                                             BND0057
   60 SUM=SUM-A(ID+J)*B(L+J)                                     BND0058
   70 B(L)=B(L)+SUM                                              BND0059
   80 RETURN                                                     BND0060
      END                                                        BND0061
```

APPENDIX G.--SUBROUTINES MULT, RESI, AND ALOCAT

```
      SUBROUTINE MULT(NWEL,A,NB,MBW,JPOS,KPOS,T,NP,KO)                  MUL0010
C ***************************************************************************MUL0020
C * SUBROUTINE TO COMPUTE AND LOAD MULTIAQUIFER COEFFICIENTS INTO      *MUL0030
C * COMPRESSED COEFFICIENT MATRIX 'A'. COEFFICIENTS ARE THOSE TERMS OF *MUL0040
C * MULTIAQUIFER FORMULATION THAT MULTIPLY HEAD OF LAYERS OPEN TO A    *MUL0050
C * WELL OR WELLS.                                                     *MUL0060
C ***************************************************************************MUL0070
      PARAMETER MW=365,MJ=8,MQ=4                                        MUL0080
      DIMENSION A(1),NB(1),JPOS(1),KPOS(1),T(1),NP(1)                   MUL0090
      COMMON/WIS/IL(MW),JL(MW),NWL(MW),KLL(MQ,MW,MJ),RWL(MW),QWL(MJ,    MUL0100
     1MW),C2(MJ,MW),C1(MQ,MW),NU(MJ,MW),NR(MQ,MW),PI2                   MUL0110
C                                                                       MUL0120
      DO 100 L=1,NWEL                                                   MUL0130
      I=IL(L)                                                           MUL0140
      J=JL(L)                                                           MUL0150
      KT=MQ                                                             MUL0170
      JA=NWL(L)                                                         MUL0180
      DO 700 JJ=1,JA                                                    MUL0190
      DO 70 M=1,KT                                                      MUL0200
      K=KLL(M,L,JJ)                                                     MUL0210
      IF(K.EQ.0) GO TO 70                                               MUL0220
      N=I+JPOS(J)+KPOS(K)                                               MUL0230
      ID=(NB(I-1)+NP(N)-1)*MBW+1                                        MUL0240
      A(ID)=A(ID)-PI2*C1(K,L)*NU(JJ,L)*T(N)                             MUL0250
      DO 60 KL=M,KT                                                     MUL0260
      K1=KLL(KL,L,JJ)                                                   MUL0270
      IF(K1.EQ.0) GO TO 60                                              MUL0280
      N1=I+JPOS(J)+KPOS(K1)                                             MUL0290
      IB=ID+K1-K                                                        MUL0300
      A(IB)=A(IB)+C1(K,L)*PI2*T(N)*NU(JJ,L)*T(N1)*C1(K1,L)*C2(JJ,L)     MUL0310
   60 CONTINUE                                                          MUL0320
   70 CONTINUE                                                          MUL0330
  700 CONTINUE                                                          MUL0340
  100 CONTINUE                                                          MUL0350
      RETURN                                                            MUL0360
      END                                                               MUL0370
      SUBROUTINE RESI(IO,JO,KO,PHI,T,I,J,K,RESO,IWELL,IFLAG)            RES0010
C ***************************************************************************RES0020
C * SUBROUTINE TO COMPUTE THE RIGHT HAND SIDE CONTRIBUTION  OF MULTI-  *RES0030
C * AQUIFER COEFFICIENTS (RHS TERM OF SUBROUTINE SOLVE). THE CONTRIBU- *RES0040
C * TION IS OBTAINED BY MULTIPLYING THE COEFFICIENTS BY CORRESPONDING  *RES0050
C * HEADS PREVAILING AT THE END OF THE PREVIOUS TIME STEP AND SUMMING  *RES0060
C * THE PRODUCTS. THE SUM TERMED RESO WHEN PASSED BACK TO SUBROUTINE   *RES0070
C  SOLVE IS THEN ADDED TO THE RHS TERM FOR THE NODE.                  *RES0080
C ***************************************************************************RES0090
      REAL *8PHI                                                        RES0100
      PARAMETER MW=365,MJ=8,MQ=4                                        RES0110
      DIMENSION PHI(IO,JO,KO),T(IO,JO,KO)                               RES0120
      COMMON/WIS/IL(MW),JL(MW),NWL(MW),KLL(MQ,MW,MJ),RWL(MW),QWL(MJ,    RES0130
     1MW),C2(MJ,MW),C1(MQ,MW),NU(MJ,MW),NR(MQ,MW),PI2                   RES0140
C                                                                       RES0150
```

```
      RES2=0.0                                                          RES0160
      RES1=0.0                                                          RES0170
      KK=MQ                                                             RES0190
      KW=0                                                              RES0200
      L=IWELL                                                           RES0210
      JA=NWL(L)                                                         RES0220
C  IWELL IS A COUNTER FOR PUMPED NODE(MULTIAQUIFER)                     RES0230
C  KW COUNTS THE NUMBER OF TIMES LAYER K IS TAPPED                      RES0240
C  IF LAYER K IS NOT TAPPED LOOP TO 70                                  RES0250
      DO 70 JJ=1,JA                                                     RES0260
      SUMZ=0.0                                                          RES0270
      DO 60 KL=1,KK                                                     RES0280
      K1=KLL(KL,L,JJ)                                                   RES0290
      IF(K1.EQ.K) KW=KW+1                                               RES0300
      IF(K1.EQ.0.AND.KL.EQ.K) GO TO 70                                  RES0310
      IF(K1.EQ.0) GO TO 60                                              RES0320
      SUMZ=SUMZ+PHI(I,J,K1)*C1(K1,L)*T(I,J,K1)                          RES0330
   60 CONTINUE                                                          RES0340
      RES2=RES2+PI2*C1(K,L)*NU(JJ,L)*C2(JJ,L)*T(I,J,K)*SUMZ             RES0350
   70 CONTINUE                                                          RES0360
      IF(KW.EQ.0) GO TO 75                                              RES0370
      RES1=-C1(K,L)*PI2*T(I,J,K)*PHI(I,J,K)*NR(K,L)                     RES0380
   75 RESO=RES2+RES1                                                    RES0390
      IFLAG=1                                                           RES0400
      RETURN                                                            RES0410
      END                                                               RES0420
      SUBROUTINE ALOCAT(PHI,OLD,T,S,TK,DELX,DELY,DELZ,IO,JO,KO,K5,HW,Q, ALO0010
     1NS1,II,I,J,TC,TR,QRE)                                             ALO0020
C ***************************************************************** ALO0030
C * SUBROUTINE TO COMPUTE AND PRINT FLOW COMPONENTS FOR LAYERS OF NODE *ALO0040
C * SETS CONTAINING MULTIAQUIFER WELLS AND TO COMPUTE BALANCE OF FLOWS *ALO0050
C * SUBROUTINE PRIMARILY USED TO TEST AND DEBUG MULTIAQUIFER CHANGES   *ALO0060
C * IN SUBROUTINE SOLVE AND CODE OF SUBROUTINES MULT AND RESI.         *ALO0070
C * SUBROUTINE ALOCAT IS CALLED FRON SUBROUTINE CHECKI(STATEMENT NUMBER*ALO0080
C * CHK1578) AND IS PRESENTLY INVOKED FOR THE LAST STRESS PERIOD ONLY. *ALO0090
C * QL IS SUM OF LATERAL FLOW BETWEEN NODE AND ADJACENT NODES.         *ALO0100
C * QS IS FLOW IN/OUT OF STORAGE                                       *ALO0110
C * QQ IS SUM OF FLOW BETWEEN LAYER K AND LAYERS K-1,AND K+1           *ALO0120
C * QOUT IS FLOW BETWEEN NODE AND WELL                                 *ALO0130
C * QQQ IS QOUT IN CUBIC FEET PER DAY                                  *ALO0140
C * PST IS VARIABLE DENSITY SOURCE TERM FOR NODE                       *ALO0150
C * DIFF IS SUM OF ALL FLOWS I.E. IT IS NODAL BALANCE                  *ALO0160
C ***************************************************************** ALO0170
      REAL *8PHI,P                                                      ALO0180
      PARAMETER MW=365,MJ=8,MQ=4                                        ALO0190
      DIMENSION PHI(IO,JO,KO),OLD(IO,JO,KO),T(IO,JO,KO),S(IO,JO,KO),    ALO0200
     1TK(IO,JO,K5),DELX(JO),DELY(IO),DELZ(KO),TC(IO,JO,KO),TR(IO,JO,KO) ALO0210
      DIMENSION QV(6),HW(MJ),QRE(IO,JO,KO)                              ALO0220
      COMMON/WIS/IL(MW),JL(MW),NWL(MW),KLL(MQ,MW,MJ),RWL(MW),QWL(MJ,    ALO0230
     1MW),C2(MJ,MW),C1(MQ,MW),NU(MJ,MW),NR(MQ,MW),PI2                   ALO0240
      COMMON /SPARAM/ TMAX,CDLT,DELT,ERR,TEST,SUM,SUMP,QR               ALO0250
C                                                                       ALO0260
```

```
      AREA=DELX(J)*DELY(I)                                              AL00270
      KK=MQ                                                             AL00290
      WRITE(6,21)                                                       AL00300
   21 FORMAT(1HO,'    NODE')                                            AL00310
      K6=KO+1                                                           AL00320
      DO 70 L=1,K6                                                      AL00330
   70 QV(L)=0.0                                                         AL00340
      DO 71 L1=2,KO                                                     AL00350
      L=L1-1                                                            AL00360
      QV(L1)=TK(I,J,L)*(PHI(I,J,L)-PHI(I,J,L1))                         AL00370
   71 QV(L1)=+QV(L1)*AREA                                               AL00380
      DO 320 L=1,KK                                                     AL00390
      P=PHI(I,J,L)                                                      AL00400
      QA=TR(I,J-1,L)*(PHI(I,J-1,L)-P)*DELY(I)                           AL00410
      QB=TR(I,J,L)*(PHI(I,J+1,L)-P)*DELY(I)                             AL00420
      QC=TC(I-1,J,L)*(PHI(I-1,J,L)-P)*DELX(J)                           AL00430
      QD=TC(I,J,L)*(PHI(I+1,J,L)-P)*DELX(J)                             AL00440
      QL=(QA+QB+QC+QD)                                                  AL00450
      QS=S(I,J,L)*(P-OLD(I,J,L))                                        AL00460
      QS=-QS*AREA/DELT                                                  AL00470
      IF(S(I,J,L).LT.0.0) QS=999999.                                    AL00480
      QOUT=0.0                                                          AL00490
      JA=NWL(II)                                                        AL00500
      DO 79 JJ=1,JA                                                     AL00510
      K=KLL(L,II,JJ)                                                    AL00520
      IF(K.EQ.0) GO TO 79                                               AL00530
      QOUT=QOUT+PI2*C1(K,II)*NU(JJ,II)*T(I,J,K)*(P-HW(JJ))              AL00540
   79 CONTINUE                                                          AL00550
      QQQ=QOUT*86400.                                                   AL00560
      QOUT=-QOUT                                                        AL00570
      QQ=QV(L)-QV(L+1)                                                  AL00580
      PST=QRE(I,J,L)                                                    AL00590
      DIFF=QL+QS+QQ+QOUT+PST                                            AL00600
      IF(S(I,J,L).LT.0.0) DIFF=999.99999                                AL00610
  699 CONTINUE                                                          AL00620
      WRITE(6,20)I,J,L,QL,QS,QQ,QOUT,PST,QQQ,DIFF                       AL00630
   20 FORMAT(2X,I2,',',I2,',',I2,6(3X,E14.8),6X,F12.5)                  AL00640
  320 CONTINUE                                                          AL00650
      END                                                               AL00660
```


☆ U.S. GOVERNMENT PRINTING OFFICE: 1988—201 - 979 / 60007