Digital Models of Ground-Water Flow in the Cape Cod Aquifer System, Massachusetts

United States Geological Survey Water-Supply Paper 2209

Prepared in cooperation with the Commonwealth of Massachusetts, Water Resources Commission; Barnstable County; and the National Park Service

Digital Models of Ground-Water Flow in the Cape Cod Aquifer System, Massachusetts

By JOHN H. GUSWA and DENIS R. LeBLANC

Prepared in cooperation with the
Commonwealth of Massachusetts,
Water Resources Commission;
Barnstable County; and the
National Park Service

U.S. GEOLOGICAL SURVEY WATER-SUPPLY PAPER 2209

DEPARTMENT OF THE INTERIOR
DONALD PAUL HODEL, Secretary

U.S. GEOLOGICAL SURVEY
Dallas L. Peck, Director

UNITED STATES GOVERNMENT PRINTING OFFICE: 1985

For sale by the Distribution Branch, U.S. Geological Survey,
604 South Pickett Street, Alexandria, VA 22304

Library of Congress Cataloging in Publication Data

Guswa, John H.
Digital models of ground-water flow in the Cape Cod aquifer system, Massachusetts.
(Water-supply paper ; 2209)
Bibliography: p.
Supt. of Docs. no. : I 19.13:2209
1. Groundwater flow—Massachusetts—Cape Cod—Mathematical models. 2. Groundwater flow—Massachusetts—Cape Cod—Data processing. I. Guswa, John H. II. LeBlanc, Denis R. III. Title. IV. Series: U.S. Geological Survey water-supply paper ; 2209.
GB1197.7.G87 551.49'09744'92 82-600101

CONTENTS

FIGURES

TABLES

Digital Models of Ground-Water Flow in the Cape Cod Aquifer System, Massachusetts

By JOHN H. GUSWA *and* DENIS R. LEBLANC

Abstract

The Cape Cod aquifer system was simulated with three-dimensional finite-difference ground-water-flow models. Five areas were modeled to provide tools that can be used to evaluate the hydrologic impacts of regional water development and waste disposal.

The model boundaries were selected to represent the natural hydrologic boundaries of the aquifer. The boundary between fresh and saline ground water was treated as an interface along which there is no dispersion. The saline-water zone was treated as static (nonflowing).

Comparisons of calculated and observed values of head, position of the boundary between fresh and saline water, and ground-water discharge (at locations where data were available) indicate that the simulated ground-water reservoirs generally agree with field conditions.

Model analyses indicate that the total steady-state freshwater-flow rate through the five modeled areas is approximately 412 cubic feet per second.

INTRODUCTION

Background

Ground water is the principal source of freshwater for Cape Cod, Massachusetts. The Cape is composed of unconsolidated glacial moraines and outwash plains that form a hook-shaped peninsula extending 40 mi into the ocean (fig. 1). It is separated from the mainland by a sea-level canal connecting Buzzards Bay and Cape Cod Bay. A lens-shaped reservoir of fresh ground water is maintained in dynamic equilibrium beneath the Cape by recharge from precipitation and discharge to the sea.

Demand for water has increased as the number of year-round residents and summer vacationers on Cape Cod has grown. State and local governments are concerned that increased pumpage to meet the increased demand may cause undesirable changes in water-table and pond levels and may reduce discharge of freshwater to coastal brackish-water bodies. Also, there is great concern that land disposal of solid and liquid waste may deteriorate water quality, and contaminants may move through the aquifer to wells, ponds, and streams.

Several problems are generally encountered in the management of ground-water resources. Hydrologic properties of aquifers are dependent on geology, and determination of these properties is usually complex and expensive. In addition, local stresses can have regional hydraulic effects on an aquifer. Since ground-water movement generally is slow and cannot be observed directly, it must be inferred from indirect measurements and abstract mathematical reasoning. Therefore, the consequences of management decisions may not be noticed for decades and the results may then be, for all practical purposes, irreversible.

Flow in an aquifer can be described analytically by differential equations. These equations will yield direct solutions, which may not be useful for some problems because of the complexity of most aquifer systems. Approximate solutions of the flow equations describing aquifers with spatial variation of hydrologic properties and stresses and irregular geometries can be obtained with numerical methods. The reliability of these numerical solutions is dependent upon the accuracy of the input data and the size of the aquifer element which the flow equations describe.

Purpose and Scope

The purpose of this report is to describe digital simulation models prepared as part of a 4-year study of the ground-water resources of Cape Cod, in cooperation with the Massachusetts Water Resources Commission, Divisions of Water Resources and Water Pollution Control; Barnstable County; and the National Park Service. Included in this report are the sources and types of data used in constructing five models, model calibration, and possible sources of inaccuracies in the results. The five modeled areas and their names are shown in figure 2.

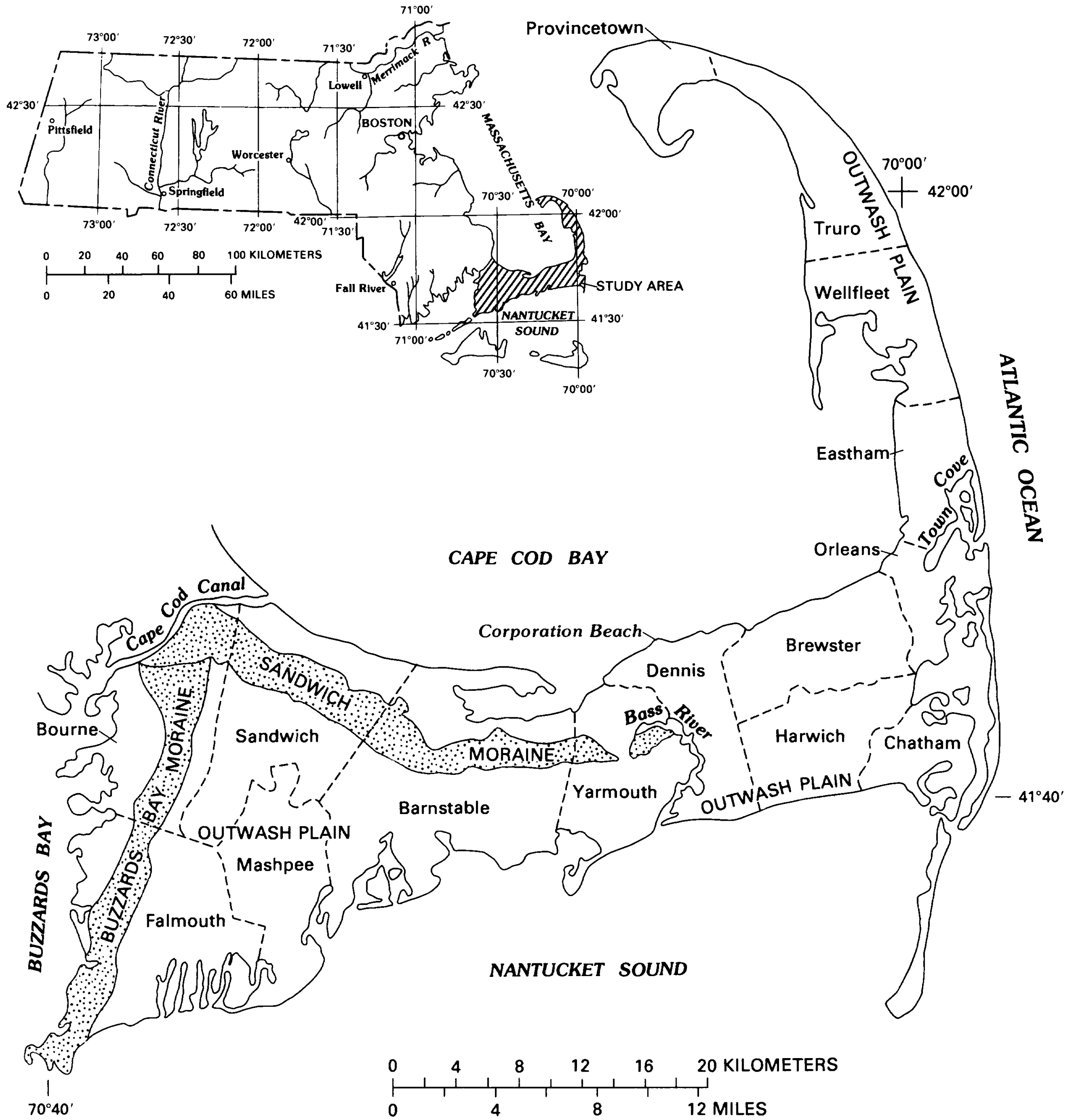


Figure 1. Physical features of Cape Cod (from Oldale, 1976).

The models provide information regarding the regional behavior of the aquifer system. Although detailed analyses of local hydrologic conditions are beyond the scope of this study, the principles used to construct these models can be used to construct more detailed models of smaller areas if sufficient data are available.

HYDROGEOLOGY OF CAPE COD

Geologic Framework and Water-Bearing Characteristics

Bedrock underlies Cape Cod, but is not exposed at land surface. The altitude of the irregular bedrock surface

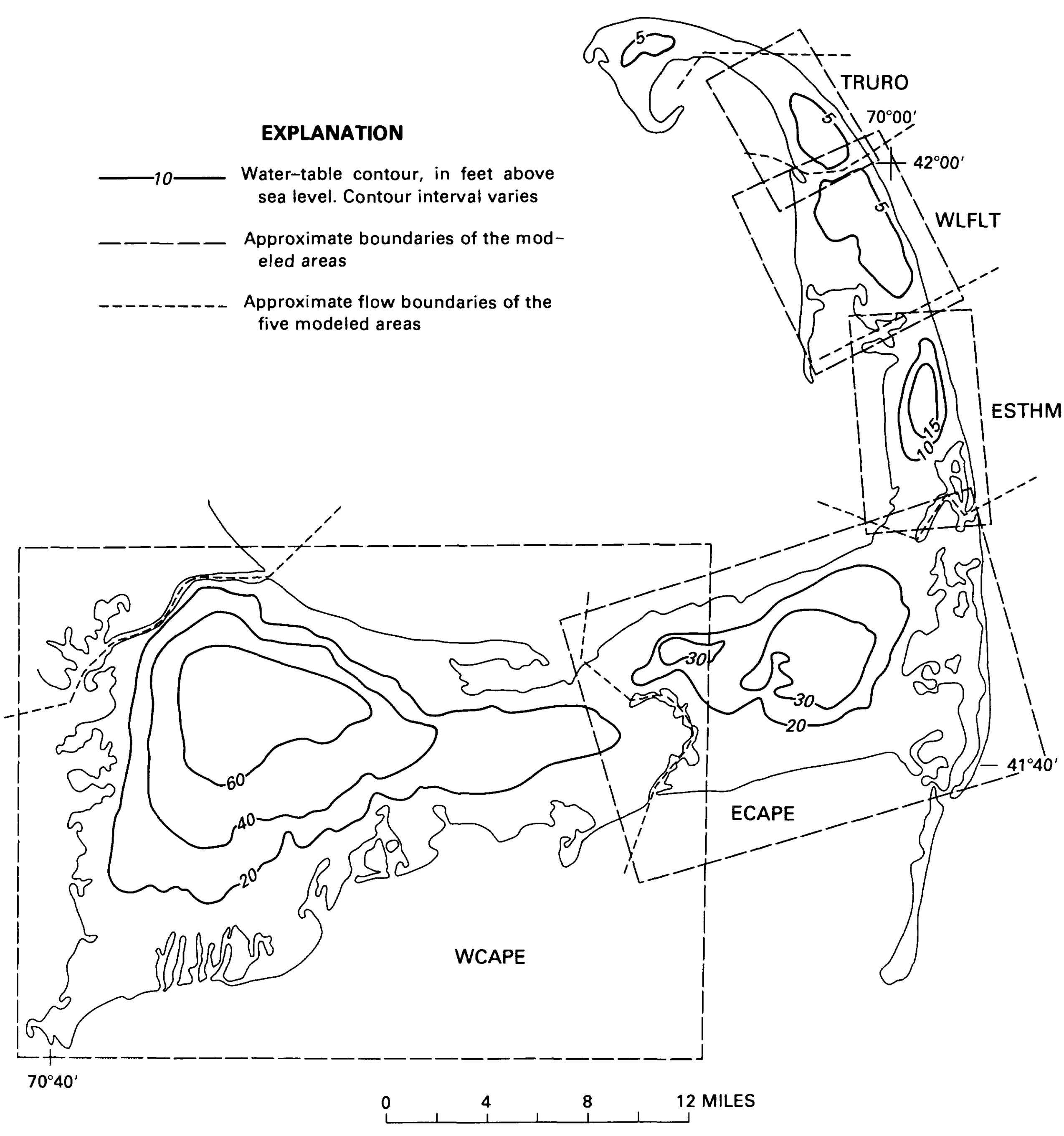


Figure 2. Freshwater-flow systems and the approximate boundaries of the modeled areas, Cape Cod, May 25–27, 1976.

ranges from 80 feet below sea level near the Cape Cod Canal to more than 900 feet below sea level near Provincetown (Oldale, 1969). The bedrock is overlain by unconsolidated sediments deposited by ice sheets during the Pleistocene Epoch as a series of end moraines and outwash plains that characterize coastal New York (Long Island), Rhode Island, and Massachusetts (Oldale, 1976). The glacial sediments consist of sand, gravel, silt, clay, and till. Along the coast, these deposits have been reworked since the Pleistocene by ocean currents and wind.

Sandy till mixed with stratified sand, gravel, and silt forms the Buzzards Bay and Sandwich moraines (fig. 1). These moraines are low, broad north- and east-trending ridges of moderately rugged topography. East of the Bass River, the east-trending Sandwich moraine deposits are buried by outwash-plain deposits.

Extreme lithologic variation over short distances and depths to water generally greater than 30 ft have discouraged exploration for water supply in the moraines. Projected yields of 4.5 ft^3/s (2,000 gal/min) have been reported (Thomas Mullen, water superintendent, Barnstable Fire District, oral commun., 1977) for 24-inch diameter gravel-packed wells screened in sand and gravel in the moraine area of Barnstable. Wells drilled in similar nearby areas of the moraine, however, have penetrated silt and clay layers several hundred feet thick and have been reported to be dry holes. The lack of test-hole information and the extreme lithologic variation in the moraines do not permit stratigraphic correlation.

The western part of the outwash-plain deposits in Bourne, Sandwich, Falmouth, Mashpee, and western Barnstable is composed primarily of stratified sand and gravel but has local silt and clay layers. These deposits generally become finer-grained with depth, especially to the south, where very fine sand, silt, and clay predominate below 100 ft in many places.

The eastern part of the outwash plain, in eastern Barnstable, Yarmouth, Dennis, Brewster, Harwich, Chatham, and Orleans, is composed of stratified sand and gravel which, in places, is mixed with till and ice-contact deposits, silt, and clay. The northern section of this part of the outwash plain is generally coarser-grained than the southern section and is mixed with or overlies very coarse-grained ice-contact deposits and till, especially in Yarmouth, Dennis, and Chatham. To the south, clay and silt layers are commonly interbedded with sand and gravel. Along Nantucket Sound, the eastern part of the outwash plain is underlain by a silt and clay deposit which is thicker than 150 ft in places. The silt and clay are reported to have been deposited in a lake that occupied the present site of eastern Nantucket Sound (Oldale, 1976).

The lithologic variation within the outwash plain deposits is not as extreme as within the moraine areas, and well yields range from 0.45 to 1.56 ft^3/s (200 to 700 gal/min), or 0.05 to 0.16 ft^3/s (20 to 70 gal/min) per foot of 24-inch diameter gravel-packed screen.

North of the Sandwich moraine are deposits of sand and gravel which, in some areas, grade northward into finer-grained deposits (silt and clay). Results of test drilling along the southern shore of Cape Cod Bay (from the Cape Cod Canal to Brewster) indicate that silt and clay layers, more than 100 ft thick in places, are commonly interbedded with the sand and gravel beds.

The silt and clay deposits confine sand and gravel in many areas along the bay. Because of the confining conditions, the interface between fresh and saline ground water is displaced offshore. For example, at Corporation Beach in Dennis (fig. 1) a well located 100 ft inland from the shoreline was drilled through unconfined and confined sediments to bedrock at a depth of 316 ft and penetrated only freshwater (chloride concentration less than 250 mg/L). Hydraulic head was 5 ft greater in the deep aquifer, confined by 35 ft of silt and clay, than in the shallow water-table aquifer.

From Orleans to Truro, the Cape is underlain by glacial outwash deposits consisting primarily of sand and gravel interbedded with silt and clay. Exposures of these deposits along the eastern shore of the outer Cape indicate that the silt and clay layers are contorted and discontinuous. Lack of well-log data precludes determination of the areal extent of the silt and clay layers. The only large-scale water-supply system developed in these deposits is in the town of Truro.

Holocene deposits include the salt marshes, beaches, spits, and dunes. They are composed of sand, silt, and clay. Few wells have been drilled in these deposits.

Hydraulic Properties of Sediments

The transmissivity of an aquifer may be estimated by multiplying a calculated or estimated hydraulic conductivity value by aquifer thickness. A more general method of calculating the transmissivity of an aquifer consisting of several hydrogeologic units is given by equation 1.

$$T = \sum_{i=1}^{n} K_i m_i, \qquad (1)$$

in which

T is the transmissivity (L^2/T);

K_i is the hydraulic conductivity of the ith hydrogeologic unit (L/T);

m_i is the thickness of the ith hydrogeologic unit (L);

and

n is the number of hydrogeologic units within the specified aquifer thickness.

Analysis of Specific-Capacity Data

The hydraulic conductivities of different lithologies on Cape Cod were estimated by analyzing specific-capacity data (well pumping rate/observed drawdown) from 265 well-performance tests. The tested wells generally had short screens that tapped a small part of the total thickness of the aquifer. Within the screened interval, the lithology was generally consistent and could be identified.

The specific capacity of each well was converted to transmissivity for the screened interval using the nonsteady-flow equation for unconfined aquifers given by Theis (1963) and shown below:

$$T' = \frac{Q}{s}(K - 264 \log_{10} 5S + 264 \log_{10} t), \qquad (2)$$

in which

Q is the well discharge, in gallons per minute;

s is the drawdown in the well, in feet;

K is a factor equal to $-66 - 264 \log_{10} (3.74 r^2 \cdot 10^{-6})$;

Table 1. Estimates of average hydraulic conductivity for lithologic types calculated from specific-capacity data

Lithology	Number of samples	Hydraulic conductivity (ft/d) Mean value	Standard deviation	Median value
Fine sand	34	450	310	440
Fine to medium sand	51	450	280	400
Medium sand	14	500	450	310
Medium to coarse sand	22	600	380	480
Coarse sand	11	430	220	400
Fine to coarse sand	42	580	290	520
All sand	174	500	310	400
Sand and gravel	16	630	430	420
Fine sand and gravel	48	530	370	420
Medium sand and gravel	9	740	460	730
Coarse sand and gravel	18	630	430	420
All sand and gravel	91	580	400	400

r is the radial distance to the well where the drawdown is observed, in feet;
t is the length of the pumping period, in days;
S is the aquifer specific yield (dimensionless);
and
T' is a value related to transmissivity by means of an equation given by Theis (1963, p. 332, eq. 2), in gallons per day per foot.

The value of T' was converted to transmissivity, T, for the screened interval of the well by solving the relationship between T and T' presented by Theis:

$$T' \simeq T - (264Q \log_{10}(T \cdot 10^{-5}))/s.$$

The value of transmissivity for the screened interval was divided by the screened length of the well to obtain the average hydraulic conductivity of the screened interval.

The well-performance data and the calculated hydraulic conductivities were divided into groups according to the lithologic types reported for the screened intervals of the wells. The values within each group were averaged to obtain the estimated hydraulic conductivity for each type of material (table 1).

Because of the large deviation of values from the calculated means and because most of the available lithologic information did not have the same level of detail as table 1, the sediment-type categories were grouped into larger classes, and average values were applied to each class (table 2).

Table 2. Average hydraulic conductivity values used to estimate transmissivity.

Lithology	Hydraulic conductivity (ft/d)
Silt and (or) clay	1
Sand	450
Sand and gravel	500
Gravel	600

None of the wells were screened in silt, clay, or gravel only. The hydraulic conductivity of silt and clay is low relative to the other materials, and was therefore assigned a value of 1 ft/d. Gravel was assigned a hydraulic conductivity of 600 ft/d.

During the process of estimating hydraulic conductivity from specific-capacity data, it was assumed that the storage coefficient was 0.2; well-entrance losses were negligible; flow into the well was horizontal, radial, and uniform along the well screen; and the discharge was sustained by withdrawal from aquifer storage within the screened interval. Flow rates in these tests were generally low, and well-entrance losses were probably minimal. Errors generated by deviation from the remaining assumptions and by incorrect estimation of storage coefficient were assumed to be random and, therefore, would be minimized in the averaging process. These estimates of hydraulic conductivity may contain a small systematic error. The vertical span of the flow pattern toward the well screen would always exceed the screened interval of the well. This would cause the calculated values to be high.

The specific-capacity data and the lithologic information used in the hydraulic conductivity calculation were obtained from drillers' records. The method of specific-capacity testing varied from one driller to another, and the identification of lithologic type is somewhat subjective. The data, therefore, were of variable quality. The error due to the variable quality of the data probably is random and would be minimized in the averaging process.

The values of hydraulic conductivity in table 2 were applied by means of equation 1 to numerous locations on Cape Cod for which a lithologic log was available to estimate transmissivities of selected zones of the aquifer. These zones correspond to different layers of the flow models and will be discussed in a later section. The initial estimates of transmissivity, calculated by this method, were adjusted during model calibration.

Aquifer Test Analyses

Lateral hydraulic conductivity, the ratio of lateral to vertical hydraulic conductivity, and the storage properties of the aquifer were estimated from analysis of three aquifer tests. A 72-hour test in Truro, Mass., on a well with a 70-ft screened interval (Guswa and Londquist, 1976) was used to estimate that the average lateral hydraulic conductivity for a predominantly very fine to coarse sand is 220 ft/d. Analysis of 5-day tests in Orleans and Yarmouth, Mass., indicated a lateral hydraulic conductivity of 300 ft/d for a coarse to very coarse sand and very fine gravel with some medium sand, and 200 ft/d for a fine to medium sand.

Analysis of the Truro and Orleans aquifer-test data using a radial-flow model yielded a ratio of lateral to vertical hydraulic conductivity of less than 10:1. Because other field data were not available and other studies (Getzen, 1977, p. 10; Meyer and others, 1975, p. 19) have indicated that this ratio is reasonable for glacial outwash, the 10:1 ratio was applied to each lithologic type.

There are no reliable data available for the vertical hydraulic conductivity of the silt and clay lenses which are scattered throughout the Cape Cod aquifer system. A value of 0.1 ft/d, which is similar to that used in the analog-model analyses of ground-water flow on Long Island (Getzen, 1977), was used for these beds.

Field data for storage coefficients of the Cape Cod aquifer system are also meager. Palmer (1977, p. 45) reports a range of specific yield of 0.13 to 0.26. His values were based on analysis of aquifer tests in the Falmouth area. Analysis of the Truro (Guswa and Londquist, 1976) and Orleans aquifer-test data indicate a specific yield of between 0.10 and 0.15.

Hydrologic System

The Cape Cod aquifer system comprises the freshwater-saturated sediments through which water moves. The boundaries of this system are the water table, saline surface water bodies, and either the bedrock surface (or other material of low hydraulic conductivity) or a boundary between fresh ground water and saline ground water.

The saline-surface-water boundaries include the ocean, bays, streams, and marshes, which are generally at a fixed elevation and serve as discharge boundaries for the ground-water-flow system. At several locations, these discharge boundaries extend sufficiently far inland to subdivide the aquifer system into several nearly independent aquifers (fig. 2). Under existing hydrologic conditions, there is no flow between adjacent aquifers across the boundaries. Conditions of severe hydrologic stress may change the nature of these boundaries and cause water to flow between the individual aquifers.

Bedrock has a low hydraulic conductivity compared to the sand and gravel deposits that form the Cape and can be considered to be a no-flow boundary at the bottom of the aquifer system. In some areas, particularly in Harwich and the southern portions of Dennis and Yarmouth, thick accumulations of silt and clay underlie the sand and gravel aquifer and overlie the bedrock. These sediments also have a low hydraulic conductivity and, where they directly overlie bedrock, their upper surface is considered to be the bottom of the aquifer system.

Fresh ground water is bounded by saline surface water at the shores of the ocean, Nantucket Sound, Buzzards Bay, Cape Cod Canal, and Cape Cod Bay (fig. 1) and by saline ground water at depth. Fresh ground water is slightly less dense than saline water and, therefore, "floats" as a lens-shaped body upon underlying saline water in the ground-water reservoir. The boundary between freshwater and saline water is a zone of mixing (or diffusion) of fresh and saline water. This mixing zone is the lower boundary of the aquifer system in those areas where it is above the bedrock surface or the thick silt and clay layers.

DIGITAL-SIMULATION MODEL

The purpose of the simulation model is to calculate the hydraulic head in an aquifer at specified locations under steady-flow conditions. This is achieved by solving the steady-state differential equation of ground-water flow, which requires that the hydraulic properties, boundaries, and inflow and outflow be defined for the modeled area.

A three-dimensional steady-state-flow model was chosen for this study for the following reasons:

1. Existing hydrogeologic information and a previous study (Burns and others, 1975) indicated that the three-dimensional variability of the aquifer system had to be included in the analyses.
2. There were few long-term records of head and stress changes which could be used to calibrate a three-dimensional transient model.
3. The approximation used to represent the boundary between fresh and saline ground water is valid only for equilibrium conditions.

Flow Equation

The differential equation describing three-dimensional steady-state flow in a porous medium is

$$\frac{\partial}{\partial x}\left(K_{xx}\frac{\partial h}{\partial x}\right)+\frac{\partial}{\partial y}\left(K_{yy}\frac{\partial h}{\partial y}\right)+\frac{\partial}{\partial z}\left(K_{zz}\frac{\partial h}{\partial z}\right)=W(x,y,z), \quad (3)$$

in which

K_{xx}, K_{yy}, K_{zz} are the principal components of the hydraulic-conductivity tensor alined with the principal Cartesian coordinate axes (L/T);

h is the hydraulic head in the aquifer (L),

and

$W(x,y,z)$ is a volumetric flux per unit volume $(1/T)$ and is referred to as a source term.

It is sometimes convenient to represent one or more hydrogeologic units as a single layer. If this is done, then equation 3 is multiplied by b, the saturated thickness of the layer, giving approximately,

$$\frac{\partial}{\partial x}\left(T_{xx}\frac{\partial h}{\partial x}\right)+\frac{\partial}{\partial y}\left(T_{yy}\frac{\partial h}{\partial y}\right)+b\,\frac{\partial}{\partial z}\left(K_{zz}\frac{\partial h}{\partial z}\right)=bW(x,y,z), \quad (4)$$

in which

T_{xx}, T_{yy} are the principal components of the transmissivity tensor (L^2/T).

The source term $W(x,y,z)$ can include well discharge; recharge from precipitation or other sources such as septic tanks, sewage-treatment plants, and irrigation; and steady leakage into or out of the aquifer system through a stream, pond, marsh, or ocean bottom.

In the model, this source term is computed as

$$bW=\frac{Q_w}{\Delta x\Delta y}-Q_{re}-\frac{K_z}{m}\,(h_s-h), \quad (5)$$

in which

Q_w is well discharge from a "block" or element of the aquifer (L^3/T);

Δx is the length of the "block" in the x direction (L);

Δy is the length of the "block" in the y direction (L);

Q_{re} is aquifer recharge and is the volumetric flux per unit area of the uppermost hydrogeologic unit (L/T);

K_z is the vertical hydraulic conductivity of the streambed or ocean bottom (L/T);

m is the thickness of the streambed or ocean bottom (L);

h_s is the hydraulic head in the streambed or ocean bottom (L);

and

h is the head in aquifer (L).

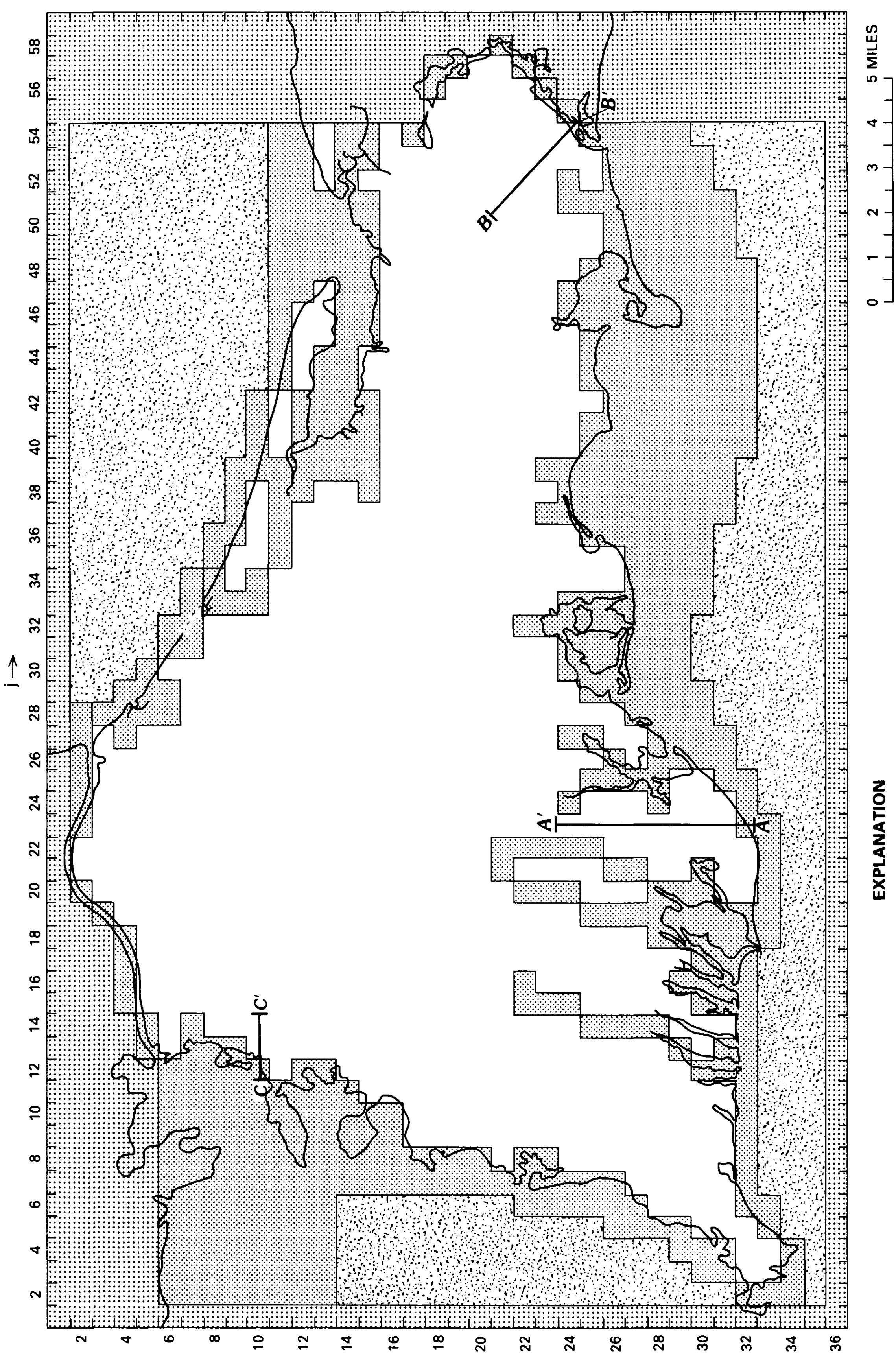
Direct solution of equation 4 is generally impossible, but a numerical solution of high accuracy can be obtained using a digital computer.

Numerical Method

To obtain a numerical solution, the partial differential equation is replaced by an approximating finite-difference equation (Trescott, 1975; Trescott and Larson, 1976) and the aquifer is subdivided into discrete blocks. Each aquifer block has one finite-difference equation describing flow within it. This yields a set of finite-difference equations which must be solved simultaneously.

There are many procedures available for solving a large number of simultaneous equations. The one used in this study is the strongly implicit procedure (SIP) developed by Stone (1968). For a complete discussion of how the finite-difference equations are solved, see Trescott (1975) and Trescott and Larson (1976).

The computer code used in this study is a modification of the code developed by Trescott (1975); input documentation and a source-code listing are included in the Supplemental Data section of this report.



EXPLANATION

- Specified-head boundary, layer 5
- Specified-head boundary, layer 4
- Area not modeled (all layers)
- A—A' Line of cross section for figure 11
- B—B' Line of cross section for figure 12
- C—C' Line of cross section for figure 13

Figure 3. Plan view of finite-difference grid for WCAPE model.

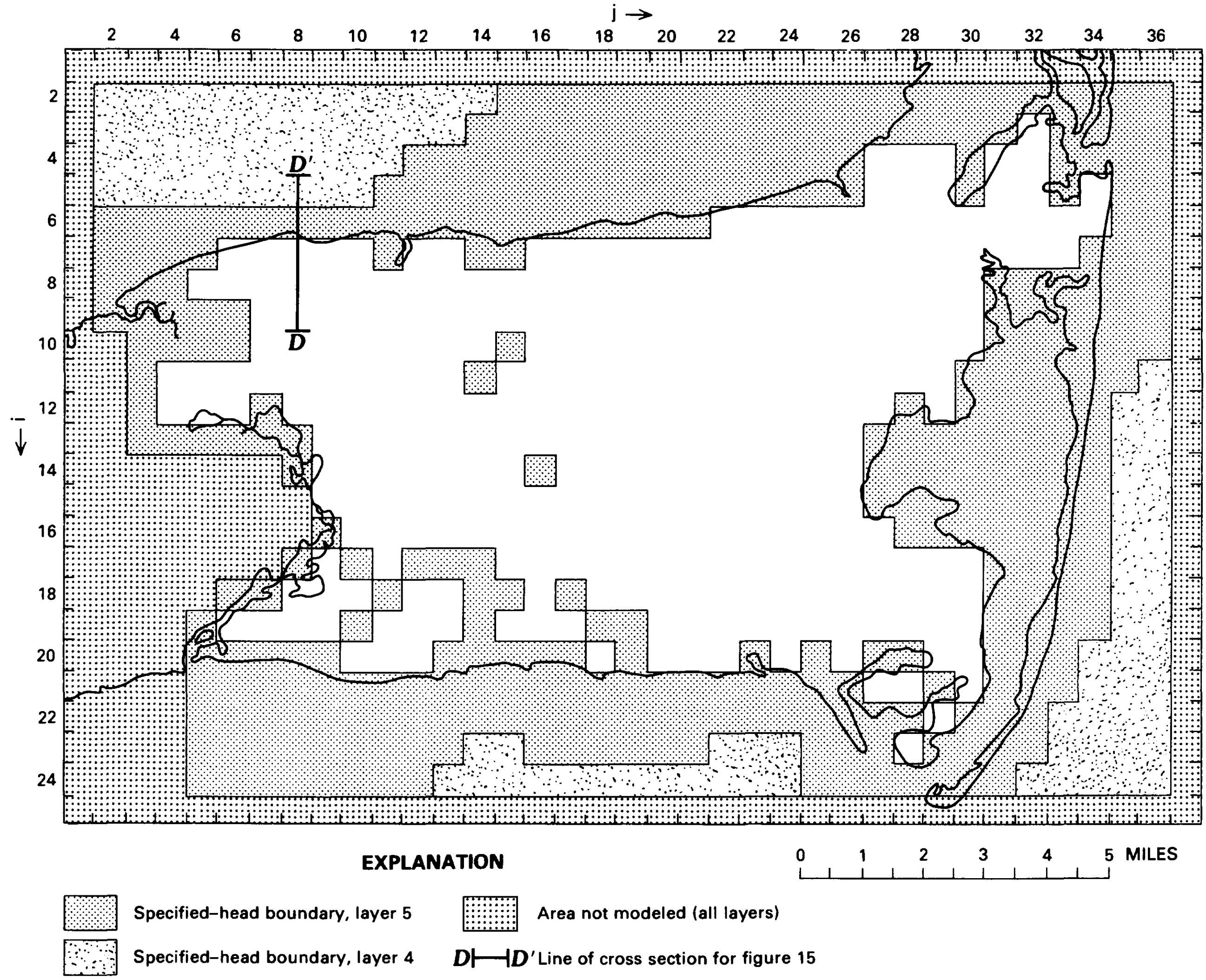


Figure 4. Plan view of finite-difference grid for ECAPE model.

APPLICATION OF SIMULATION MODEL

Model Specifications

Ground-water-flow models were developed for five areas of Cape Cod. The present flow pattern allowed construction of separate models for five of the six individual aquifers (fig. 2). Limits of the modeled areas were selected to include or nearly coincide with the natural flow boundaries of the system. These lateral boundaries are discharge boundaries, such as the seabed, streams, and marshes. They are modeled as specified-head seepage boundaries.

The modeled areas were subdivided into rectangular finite-difference grids with uniform horizontal spacing (figs. 3–7) and uneven vertical spacing. The descriptive information for the individual finite-difference grids can be seen in table 3.

The layers of each model represent elevation horizons within the aquifer system. These horizons contain one or more hydrogeologic units. Aquifer properties and stresses are assumed to be uniform within any given block and must be defined at all nodes of the grid.

By convention, nodes are located at the centers of the blocks of the grid. Any specific node or block may be referenced by citing its row (i), column (j) and layer (k) location.

Water-Transmitting Properties of the Aquifer

At about 320 selected locations in the aquifer system, lithologic logs and the average values of hydraulic conductivity for the lithologic types in table 2 were used in equation 1 to calculate transmissivity values for layers of the model. An average, or equivalent, lateral hydraulic

Table 3. Descriptive information for finite-difference grids of modeled areas

Modeled area	Number of rows	Number of col-umns	Number of layers	Horizontal grid spacing (feet)	Elevation of bottom of layer (feet below sea level) Layer number 1	2	3	4	5	6	7
WCAPE	36	59	5	2640	400	240	140	70	20	—	—
ECAPE	25	37	5	2640	400	240	140	70	20	--	--
ESTHM	22	36	7	1320	600	450	300	200	110	50	10
WLFLT	26	32	7	1320	400	280	200	140	80	40	10
TRURO	20	26	7	1320	400	280	200	140	80	40	10

conductivity for each layer was also calculated using the relationship

$$\overline{K}=\frac{\sum_{i=1}^{n} K_i m_i}{\sum_{i=1}^{n} m_i}, \qquad (6)$$

in which

$\overline{K}$ is the average lateral hydraulic conductivity (L/T);

K_i is the average lateral hydraulic conductivity of the ith hydrogeologic unit (L/T);

m_i is the thickness of the ith hydrogeologic unit (L);

and

n is the number of hydrogeologic units that occur within a layer; the top of the uppermost layer coincides with the estimated position of the water table.

Maps of the spatial variation in transmissivity and lateral hydraulic conductivity were made for all layers and served as initial values of the water-transmitting properties of the aquifer.

Equivalent vertical hydraulic conductivities were also calculated from lithologic logs at the 320 locations. Darcy's Law and the analogy between flow in an aquifer and flow of electric current were used to calculate these values. One-dimensional steady-state flow through an aquifer prism can be defined by

$$Q=KA\frac{\Delta h}{\Delta z}, \qquad (7)$$

in which

Q is the rate of flow through the prism (L^3/T);

K is the hydraulic conductivity of the aquifer in the direction of flow (L/T);

A is the cross-sectional area of the prism (L^2);

Δh is the head difference between two measuring points along the flow direction (L);

and

Δz is the length between the two measuring points (L).

The term $\frac{KA}{\Delta z}$ is analogous to electrical conductance (Prickett, 1975) and will be defined herein as hydraulic conductance, C. In a manner similar to calculating the equivalent electrical conductance when two or more conductors are connected in series, the equivalent vertical hydraulic conductance of a layered aquifer can be calculated using the relationship

$$\frac{1}{C_{v_{eq}}}=\sum_{i=1}^{n}\frac{1}{C_{v_i}}, \qquad (8)$$

in which

$C_{v_{eq}}$ is the equivalent vertical hydraulic conductance of an aquifer (L^2/T);

C_{v_i} is the vertical hydraulic conductance of the ith layer (L^2/T);

and

n is the number of layers of the aquifer.

As an example, an aquifer that contains two layers can be seen in figure 8.

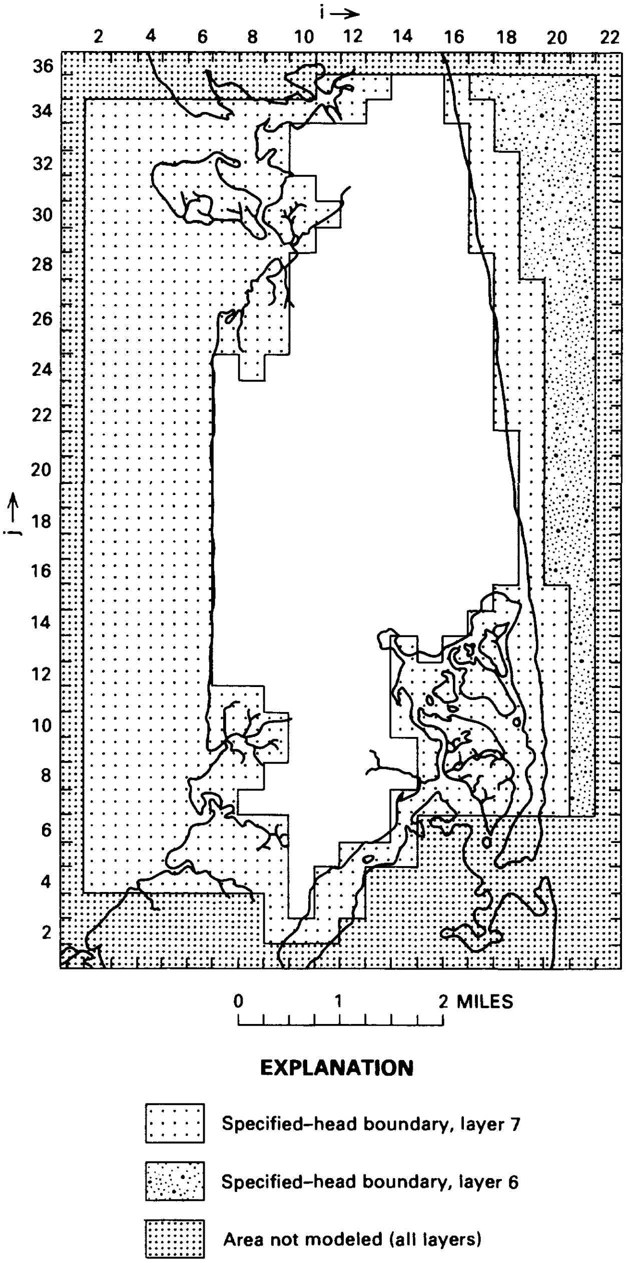


Figure 5. Plan view of finite-difference grid for ESTHM model.

With the appropriate substitutions, equation 8 can be rewritten as

$$\frac{1}{\dfrac{K_{v_{eq}} \cdot A}{\Delta z_T}} = \sum_{i=1}^{n} \frac{1}{\dfrac{K_{v_i} \cdot A}{\Delta z_i}}, \tag{9}$$

in which

$K_{v_{eq}}$ is the equivalent vertical hydraulic conductivity (L/T);

A is the cross-sectional area of flow (L^2);

Δz_T is the total thickness of the aquifer and is equal to $\sum_{i=1}^{n} \Delta z_i$ (L);

K_{v_i} is the vertical hydraulic conductivity of the ith layer (L/T);

and

Δz_i is the thickness of the ith layer (L).

Multiplying both sides of equation 9 by the constant A and rearranging terms yields

$$K_{v_{eq}} = \Delta z_T \frac{1}{\sum_{i=1}^{n} \Delta z_i / K_{v_i}}, \tag{10}$$

which also can be written as

$$K_{v_{eq}} = \frac{\sum_{i=1}^{n} \Delta z_i}{\sum_{i=1}^{n} \Delta z_i / K_{v_i}}. \tag{11}$$

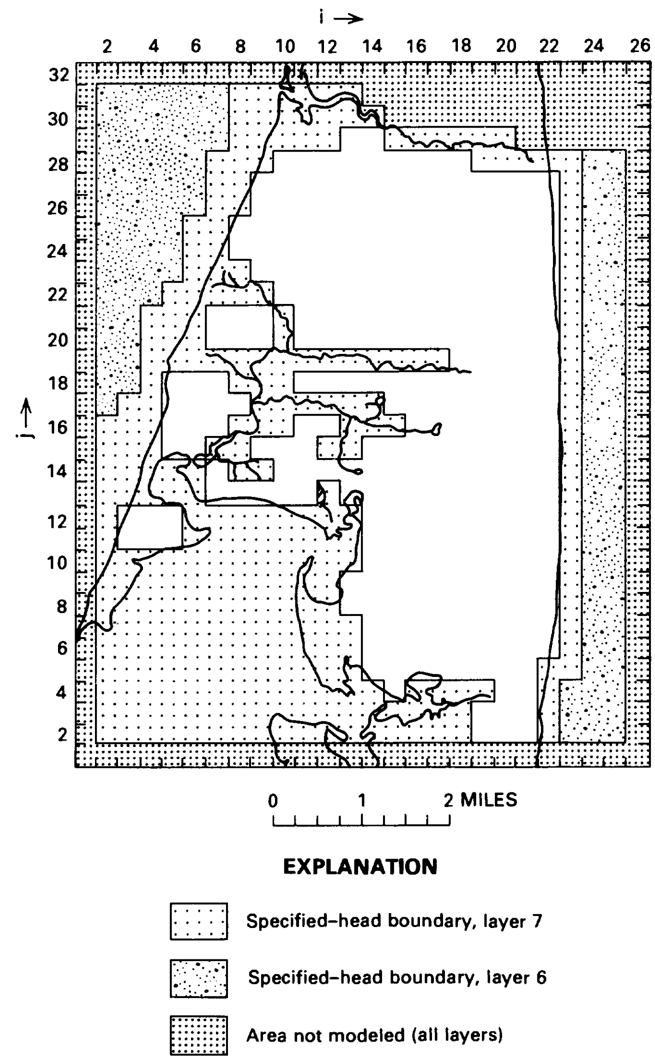


Figure 6. Plan view of finite-difference grid for WLFLT model.

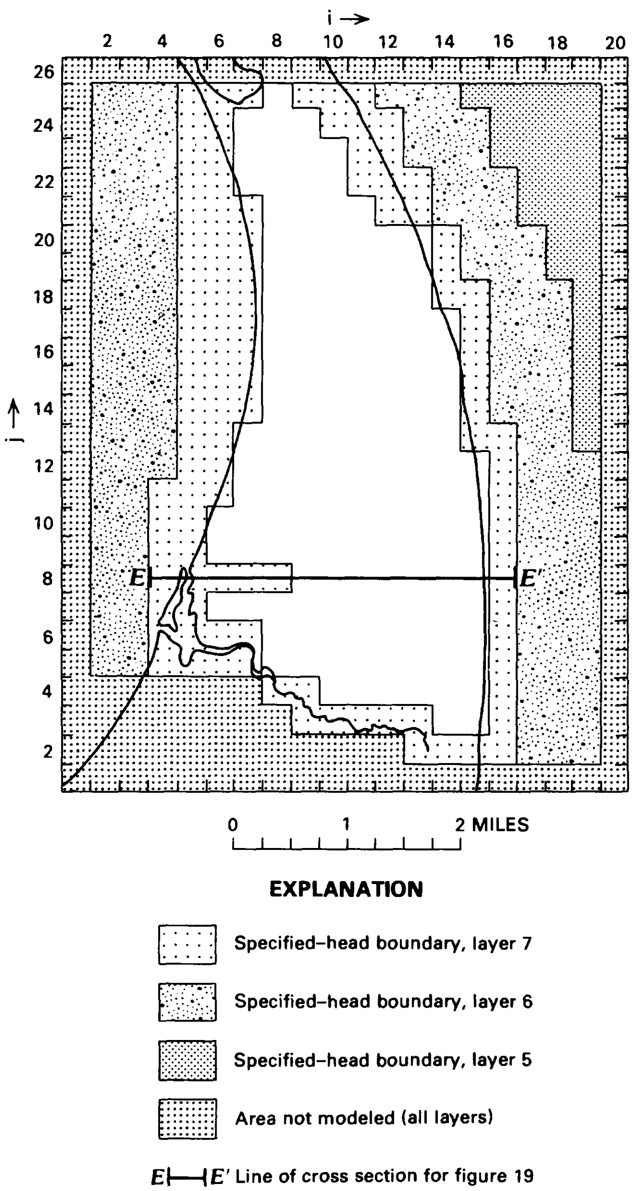


Figure 7. Plan view of finite-difference grid for TRURO model.

Equivalent vertical hydraulic conductivity values for selected nodes were compared with the estimated lateral hydraulic conductivity values to determine a ratio of lateral to vertical hydraulic conductivity. Using values for lateral transmissivity, hydraulic conductivity, the estimated ratio of lateral to vertical hydraulic conductivity, and the thickness of each block, vertical-flow coefficients (Trescott and Larson, 1976, p. X) were calculated.

During model calibration, it was necessary to adjust the initial estimates of transmissivity and lateral and vertical hydraulic conductivity. The final values used in the models are presented in the Supplemental Data section of this report.

Boundaries and Hydraulic Stresses

Well Discharge

Sites of known and significant well discharge within the study area were represented in the model by specifying a constant flux rate at the nodes representing the blocks containing the wells (well discharges are summarized in Supplemental Data, table 10). These average withdrawal rates are based on actual pumpage for 1975 and 1976. The pumpage data were provided by the local water-service districts.

Boundary Between Fresh and Saline Water

The saline surface water surrounding Cape Cod forms a surface boundary to the ground-water reservoir. The seabed is a specified-head boundary to the ground-water-flow system. The distribution of freshwater head along this boundary is a function of the saline-water depth and the density difference between freshwater and saline water. At the bottoms of the bays and the ocean, freshwater heads at the seabed-water interface must be equal to the equivalent freshwater head resulting from the saline surface-water column extending from the seabed to sea level. If the surface-water body is seawater, the resulting equivalent freshwater head above sea level is about 2.5 percent of the saline surface-water depth. It is proportionately less if the density of the surface-water body is less than the density of seawater. This equality of freshwater and saline-water head at the seabed must also exist at the boundary between fresh ground water and saline ground water if (1) the boundary is a sharp interface; (2) the saline ground water is static (nonflowing); (3) the flow system is in a state of dynamic equilibrium; and (4) the only forces acting on the ground water are head gradients arising from gravitational forces (Hubbert, 1940). Figure 9 illustrates this relationship between fresh ground water and saline surface and ground water.

The seabed was modeled as a specified-head seepage boundary. The layers for which this boundary condition was specified were determined by comparing model-layer elevations to the seabed elevation determined from bathymetric and topographic maps. The seabed elevation was specified as the bottom elevation of the source bed. The equivalent freshwater head resulting from the density difference between freshwater and seawater was specified as the source-bed head. A leakance coefficient of 20 (ft/d)/ft was used. Sensitivity analyses indicate that changing this value by an order of magnitude has no significant effect on the calculated heads or flux rates. Provided sufficient precision is used in mass-balance calculations, this is an acceptable way of approximating a constant-head condition (Trescott and others, 1976). Data on the

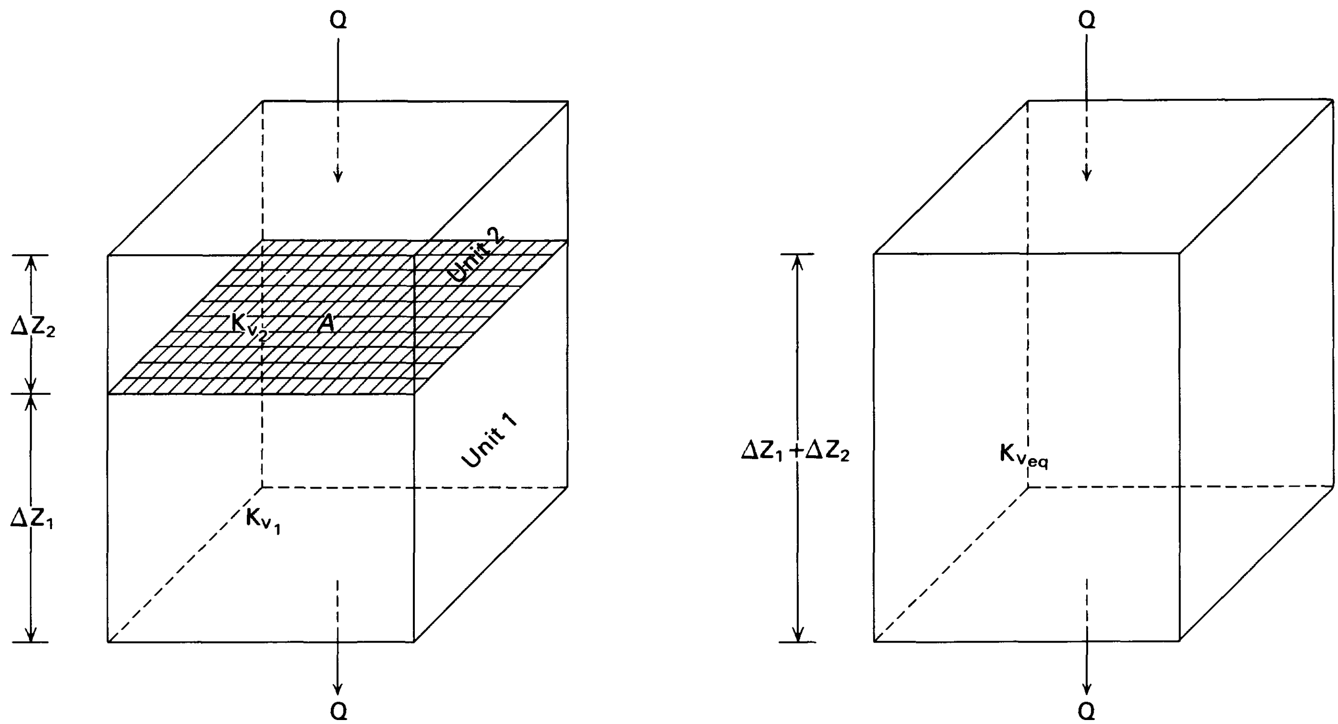


Figure 8. Schematic representation of equivalent vertical hydraulic conductivity.

spatial variability of vertical permeability and thickness of the seabed sediments, if they become available, can be easily incorporated into the models.

The boundary between fresh and saline ground water was modeled as an interface, and the saline-water zone was assumed to be static. This boundary is actually a zone of diffusion or mixing, but data collected concerning the thickness of this zone suggest that it is narrow, and for the purposes of this study it can be treated as a sharp interface.

At specified iterations, the position of this interface is calculated in accordance with the hydrodynamic model of Hubbert (1940). Starting with the bottom layer and proceeding through the grid in a systematic fashion, block transmissivity values are adjusted to reflect the percentage of the block thickness which is occupied by freshwater. For example, the transmissivity of a block in which only the upper 40 percent of the thickness contains freshwater is reduced to 40 percent of its original value. A block calculated to be wholly in the saline-water zone is assigned a transmissivity value of zero for the rest of the simulation. If a block contains freshwater partially or totally, then all blocks directly above it are assumed to contain only freshwater. The vertical-flow coefficients remain unchanged during a simulation unless the freshwater-saturated thickness of a block is determined to be zero. When this occurs, the coefficient describing flow between the block with zero freshwater thickness and the overlying block is set to zero. An identical adjustment for water-table conditions is described in Trescott and Larson (1976, p. XII).

The computational scheme requires that the starting head values be sufficiently large to define an interface position that is seaward of and deeper than the real interface position. During the iteration sequence, the calculated interface moves landward and upward until it is in balance with the freshwater-flow system.

Streams and Marshes

Many of the streams and marshes are significant boundaries to the freshwater-flow system. The effect of these boundaries varies as a function of the head within the aquifer; they generally derive some or all of their flow from ground-water discharge or seepage, and at certain locations or times of the year may be a source of aquifer recharge. These boundaries were modeled as areas of specified head, separated from the aquifer by a streambed layer. The bottom elevation of the source bed was defined to coincide with the altitude of the stream or marsh bottom. The source-bed head was set equal to the altitude of the stream or marsh surface. A leakance coefficient of 20 (ft/d)/ft was used. This value might represent a 2-ft-thick sand layer.

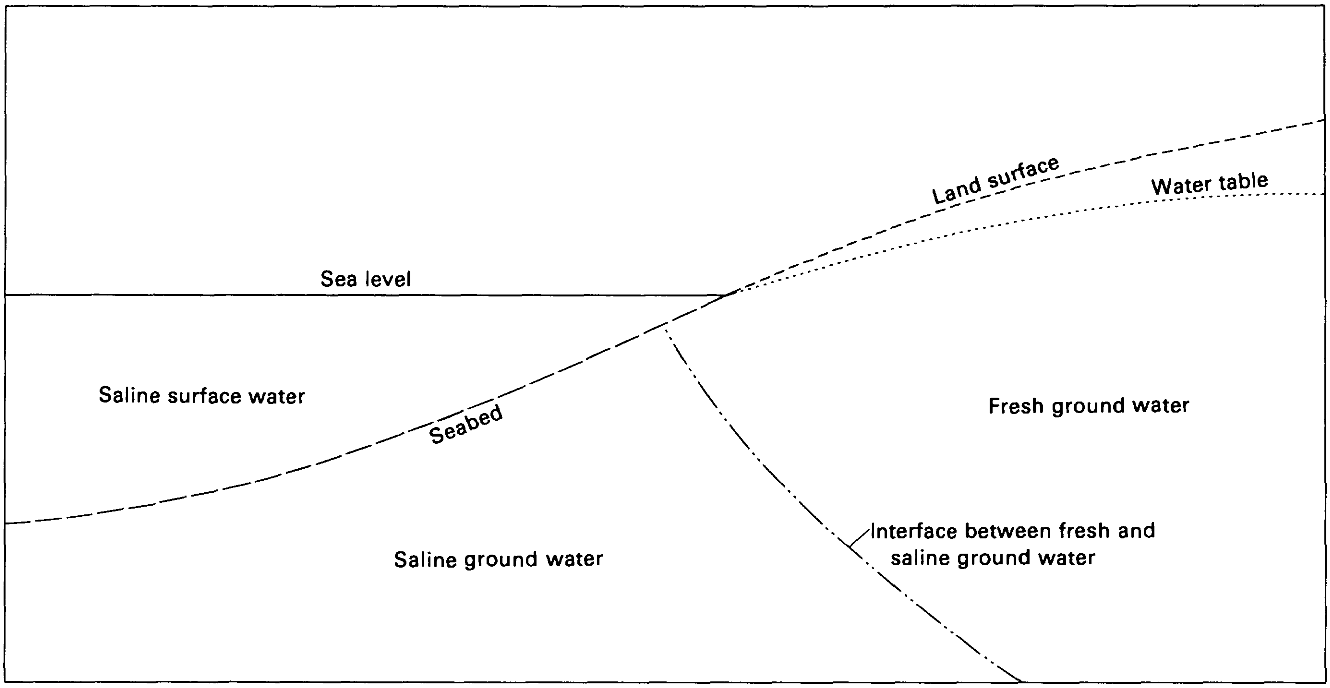


Figure 9. Schematic hydrologic section of the seaward boundary of fresh ground-water flow. (Along the seabed, freshwater head must be equal to the equivalent freshwater head resulting from the saline surface-water column extending from sea level to the seabed. Along the interface, freshwater head balances saline-water head in the aquifer.)

Aquifer Recharge

Aquifer recharge is a combination of natural recharge from precipitation and artificial recharge from sources such as septic tanks and wastewater-treatment plants. The evapotranspiration rate was estimated for several locations on Cape Cod by the Thornthwaite method (Thornthwaite and Mather, 1957). This empirical determination is based on analysis of climatological data. U.S. Weather Bureau data for 1947–1976 were used in the analysis. The calculated average evapotranspiration rate and the average precipitation rate for the same period were used to estimate an average rate of natural recharge. The rates obtained compare favorably with estimates of average natural recharge on Cape Cod made by Strahler (1972) and Palmer (1977). For each node in the top layer, this value was added to the estimated artificial recharge rate, where applicable, to obtain an average total recharge rate.

Estimated rates of artificial recharge were based on the assumption that artificial recharge occurs in those areas where the water disposal site is not in the same model block as the source of water supply. For example, the disposal lagoons of a wastewater-treatment plant and residential areas serviced by both public water supply and onsite disposal systems were treated as areas of artificial recharge. No net artificial recharge or discharge was simulated in areas serviced by both shallow domestic wells and onsite disposal systems.

The estimated artificial recharge rate was calculated on a town-by-town basis by making these assumptions:

1. Those areas identified in the draft Cape Cod 208 Water Quality Management Plan (U.S. Environmental Protection Agency, 1978) as wastewater-management problem areas (U.S. Environmental Protection Agency, 1978, map 4.1) are areas of artificial recharge because they represent areas of high concentration of septic tanks and are serviced by public water supply.
2. The rate of artificial recharge is uniform within these areas and proportional to the difference between the rate of withdrawal by the public water-supply system and any wastewater-treatment plant flow rates.

Another factor which influences the rates of aquifer recharge is the thickness of the unsaturated zone. Where the water table is close to land surface, as in low-lying swampy areas, proportionally more of the water that infiltrates the soil may be transpired by plants or evaporated, and consequently, less rainfall becomes a net addition to the ground-water reservoir than in areas with a thicker unsaturated zone. To approximate this effect for the WCAPE and ECAPE models, the calculated aquifer recharge was reduced proportionally (by approximately 10 percent) in low-lying areas near the coast. The finer grid spacing of the ESTHM, WLFLT, and TRURO models allowed a closer approximation of the positions of streams and wetlands where the water table is close to the land surface. These natural discharge areas were modeled as constant head zones. Consequently, net recharge rates did not have to be reduced.

The final recharge distributions used are presented in the Supplemental Data section of this report.

CALIBRATION OF THE STEADY-STATE FLOW MODELS

Purpose and Procedure

An important objective of calibrating the steady-state model is to improve the conceptual model of the aquifer. Developing the conceptual model requires an understanding of the physical and functional nature of the aquifer. This includes identifying sources of recharge and discharge, rates and direction of flow, variation of aquifer properties and hydraulic head, and the relation of the aquifer to surface water. The simulation model numerically integrates the effects of these factors and the computed results are therefore internally consistent with all input data, allowing one to determine if any element of the conceptual model must be revised. After initial best estimates of the input data are made, model development is an evolutionary process in which results of previous simulations are interpreted to make modifications and adjustments to the model. The testing process of adjusting input data and comparing the calculated results with field observations (calibration) allows for a better understanding of the flow system and an improvement of the conceptual model.

To demonstrate that the flow models are reasonable, field observations must be closely correlated with model results. Field observations available for the Cape Cod aquifer system include observed water levels, position of the interface between fresh and saline ground water, and discharge measurements for selected stream segments. A map was prepared from records of water levels at selected long- and short-term observation sites that show the estimated average water table from 1963 to 1976 and the interface position at several locations.

The calibration procedure minimizes differences between observed and computed values by adjusting the input data (aquifer properties, boundary conditions, and hydraulic stresses). Because of the large number of interrelated factors affecting ground-water flow, this is a subjective procedure. The degree of allowable adjustment, however, of any parameter generally is directly related to the uncertainty of its value or specification. For example, withdrawal rates are well defined, and their values were not adjusted. Hydraulic conductivity and transmissivity, however, are generally imprecisely known because lithologic variation is usually not well defined and because the methods by which they are determined are subject to many limitations. During the calibration, these values were adjusted by as much as 20 percent for those areas where the variation of lithology was believed to be gradual. These areas usually, but not always, coincided with the outwash plains. For areas where lithologic variation is known to be extreme, particularly the moraine, transmissivity and hydraulic conductivity values were adjusted by as much as a factor of 10.

Because the calibrated model is based on an interpretation of field observations, the accuracy of the model results is restricted by the accuracy of the interpretation. The water-table configuration and the position of the interface between fresh and saline ground water are estimates based on limited data. The control points for calibration of the models were approximately 150 sites for observation of water-table elevation; 27 sites for observation of head variation, chloride concentration, and specific conductance of ground water with depth; and 2 stream-segment discharge measurements. If any major corrections, revisions, or additions are made to the data base, the models should be recalibrated to reflect this better knowledge of the aquifer system.

Results

WCAPE Model

For the WCAPE model the agreement between observed average water levels and calculated water levels under steady-state conditions can be seen in table 4 and figure 10. Some of the disagreement, particularly in the higher elevations of the water table, is due to a lack of information on water levels and transmissivity in the moraine. Another reason for disagreement is that the observed average water level refers to the water level at the observation well itself, but the calculated water level refers to the water level at the center of the block which contains the well. Therefore, the discrepancy between calculated and observed water levels is greatest where the hydraulic gradient is steepest, a fact common to all five models.

Figures 11–13 show calculated altitudes of the interface between fresh and saline ground water for selected cross sections (see fig. 3). Measured chloride concentrations and a resistivity log of a test hole drilled in Mashpee are also included in figure 11. The resistivity measurement corresponding to the depth from which the lowermost sample in figure 11A was taken is approximately 750 ohm-meters, interpreted as being within the transition zone between fresh and saline ground water. On figure 11B, the nearly vertical resistivity line from approximately −310 ft to −400 ft represents a

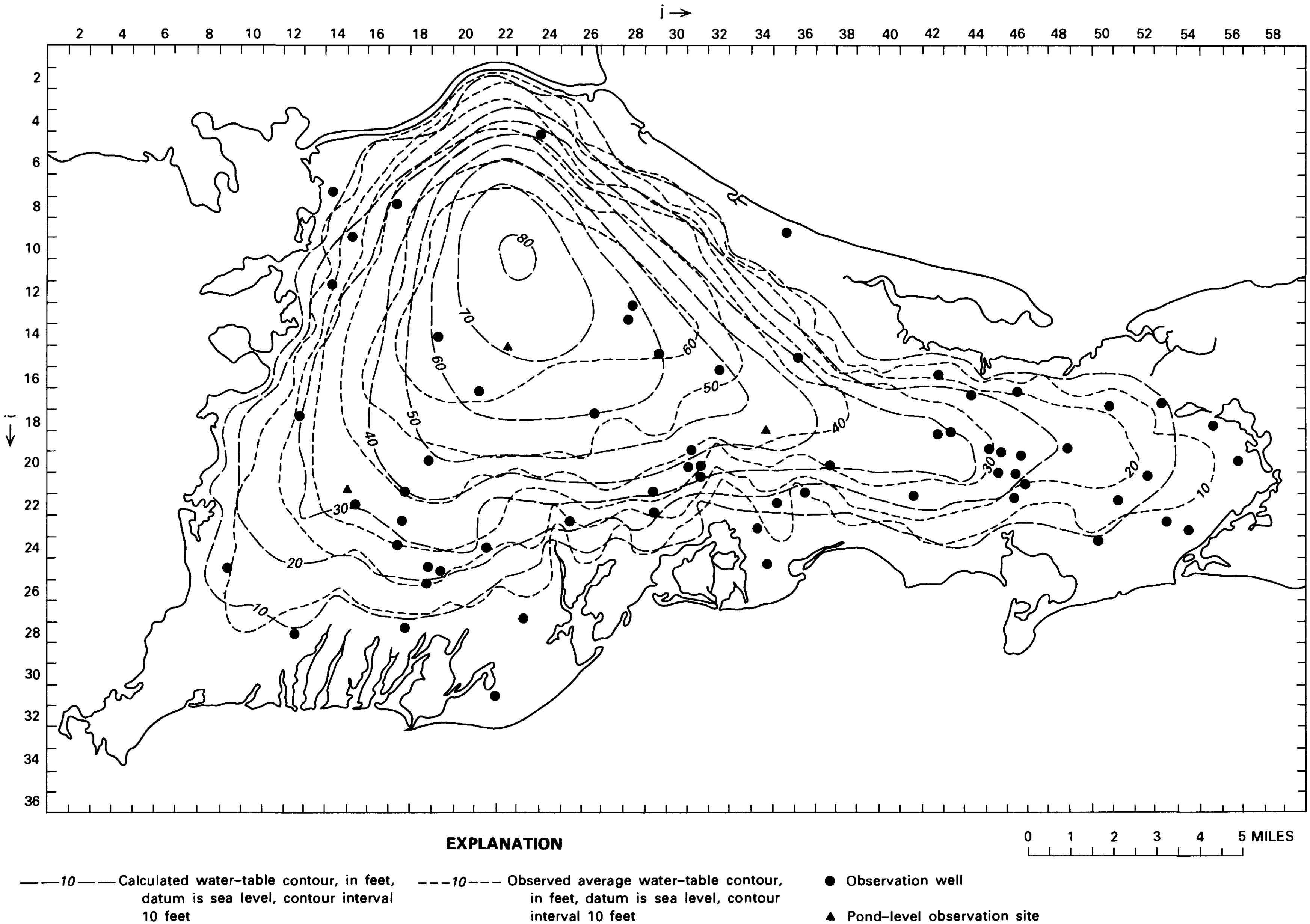


Figure 10. Calculated and observed average water table, 1963–1976, WCAPE model.

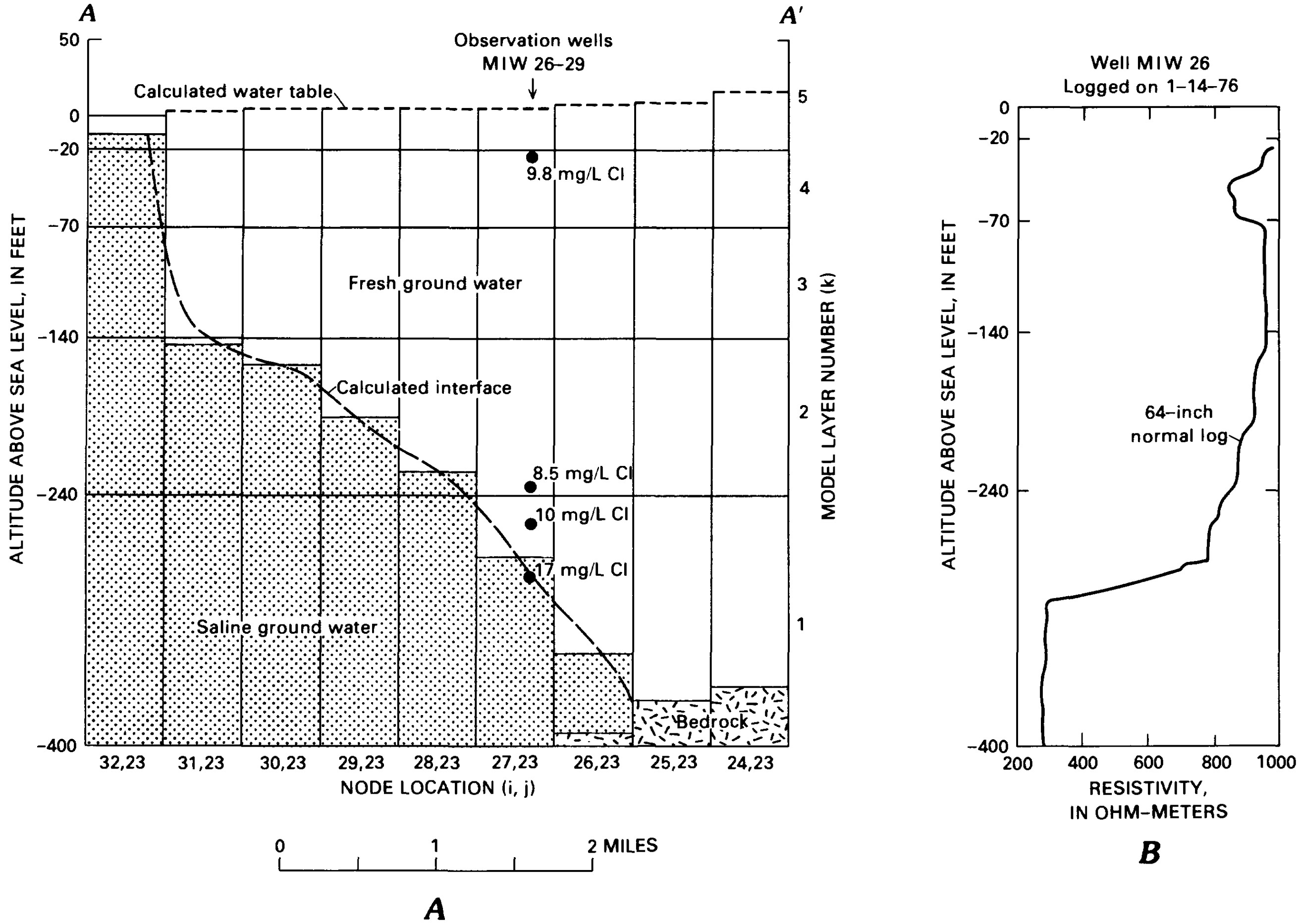


Figure 11. *A*, Calculated water table and interface between fresh and saline ground water, and chloride (Cl) content, in milligrams per liter (mg/L), of water samples collected from wells on June 20, 1979. *B*, Resistivity log for well MIW 26. Line of cross section *A–A'* is shown in figure 3.

chloride concentration equal to or greater than 15,000 mg/L, interpreted as representing the saline ground-water zone.

The lack of streamflow data hindered evaluation of the model's accuracy in simulating discharge to surface-water bodies. Stream discharges for two river segments in Falmouth were measured when streamflow on the Cape was about average and represented ground-water discharge. The measured discharges were 13 and 19 ft^3/s, and the calculated discharges to the nodes chosen to represent these stream segments were 9 and 20 ft^3/s. Collection of additional streamflow data will help improve the model and provide a better data base for future calibration.

A mass balance was calculated during each simulation. As part of these calculations, a hydrologic budget for the aquifer is computed and tabulated, providing a measure of the importance of each element of the budget. For the WCAPE model, the steady-state-recharge rate is 267 ft^3/s; the rate of withdrawal by wells is 17 ft^3/s; and discharge to the streams, marshes, and the ocean is 250 ft^3/s.

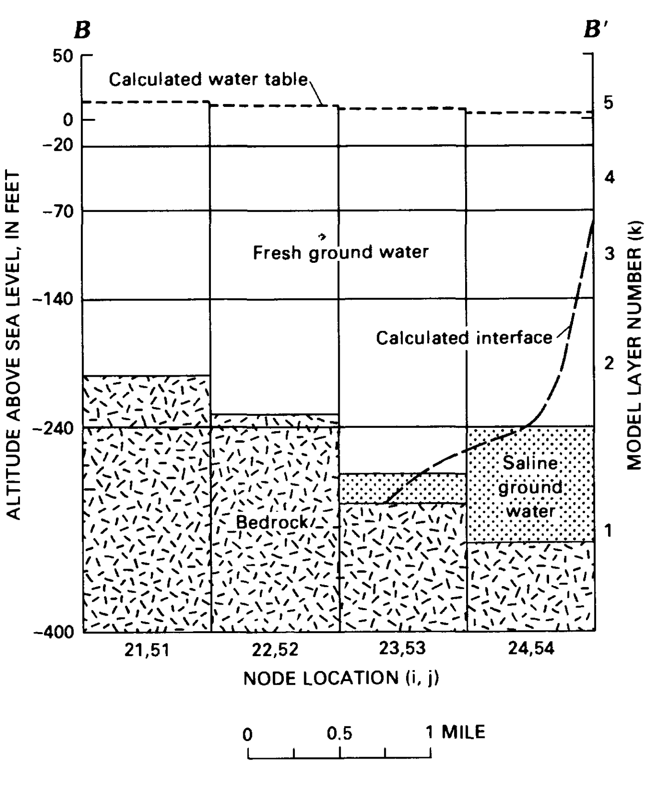


Figure 12. Calculated water table and interface between fresh and saline ground water. Line of cross section *B–B'* is shown in figure 3.

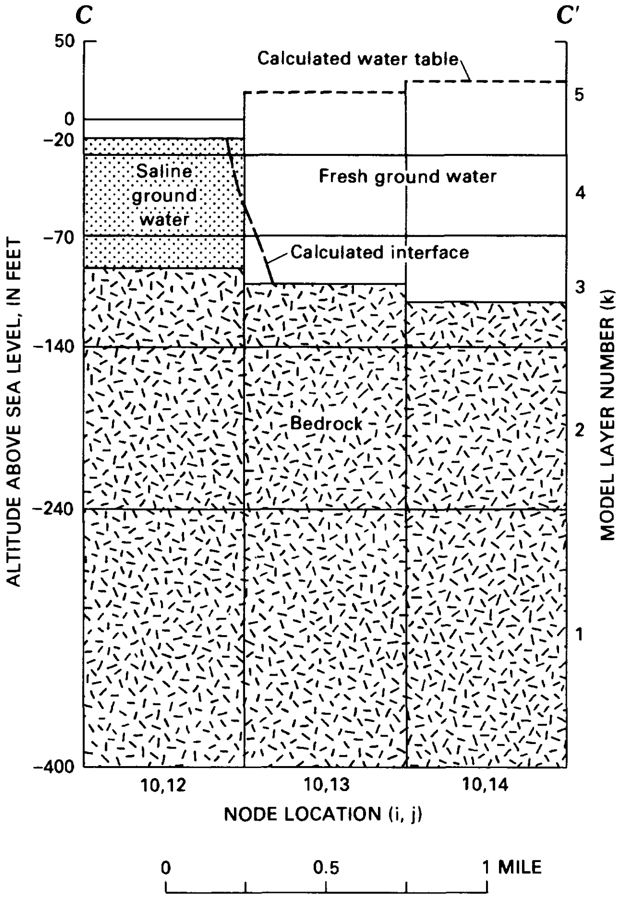


Figure 13. Calculated water table and interface between fresh and saline ground water. Line of cross section *C–C'* is shown in figure 3.

ECAPE Model

For the ECAPE model, agreement between the observed average and the calculated water levels can be seen in table 5 and figure 14. Disagreement is generally in the area where outwash plains overlie moraine deposits. The model was not able to simulate the 40-foot water-level contour in Dennis (fig. 14), nor the water levels observed deep in the aquifer in the northern part of Dennis. The model probably does not adequately approximate the complex stratigraphy associated with moraine deposits; improvement of this aspect of the model will require additional hydrogeologic information.

The calculated position of the interface between fresh and saline ground water for a selected cross section (D-D' in fig. 4) is shown in figure 15.

The mass balance calculated for the ECAPE model indicates that the steady-state recharge to the aquifer is 91 ft^3/s; withdrawal by pumpage is 7 ft^3/s; and discharge to streams, marshes, and the ocean is 84 ft^3/s.

ESTHM Model

Table 6 and figure 16 show the agreement between the calculated and observed average water levels for the ESTHM model.

The mass-balance calculations indicate that the steady-state rate of aquifer recharge is 19 ft^3/s. This rate is balanced by an equal rate of discharge to streams, marshes, and the ocean.

WLFLT Model

The agreement between calculated and observed average water levels for the WLFLT model is shown by table 7 and figure 17.

Mass-balance calculations indicate that the steady-state recharge rate, 23 ft^3/s, is balanced by an equal rate of discharge to streams, marshes, and the ocean.

TRURO Model

The agreement between calculated and observed average water levels for the TRURO model is shown in table 8 and figure 18. Mass-balance calculations for the TRURO model indicate a steady-state recharge rate of 12.3 ft^3/s; a withdrawal by pumping of 1.4 ft^3/s; and a discharge rate to streams, marshes, and the ocean of 10.9 ft^3/s.

In figure 19 the calculated position of the interface between fresh and saline ground water along cross section E-E' (in fig. 7) can be seen. Measured chloride concentrations in water samples from a group of wells near that line of section are included for comparison.

SUMMARY AND CONCLUSIONS

Cape Cod satisfies its water-supply demands almost entirely from the freshwater contained within the unconsolidated sediments of Pleistocene and Holocene age. Management of this ground-water resource requires an understanding of the behavior of this large complex aquifer system. Three-dimensional finite-difference ground-water-flow models were prepared for five areas of Cape Cod to provide information on the regional behavior of this system.

The models were developed using these premises: The boundary between fresh and saline ground water can be treated as an interface; the saline-water zone is static; and the natural discharge boundaries subdivide the aquifer system into individual small aquifers which can be modeled separately.

The models were calibrated for steady-state conditions by comparing calculated and observed water levels, position of the interface between fresh and saline ground water, and ground-water discharge rates, at approximately 150, 27, and 2 control points, respectively.

Table 4. Comparison of calculated nodal head values and observed average water levels for selected wells, 1963-1976, WCAPE model

Node (i,j,k)	U.S. Geological Survey well No.	Water level, in feet above sea level		Node (i,j,k)	U.S. Geological Survey well No.	Water level, in feet above sea level	
		observed average	calculated at center of grid block			observed average	calculated at center of grid block
5,24,5	SDW 263	36.7	39.5	20,56,5	YAW 111	7.5	5.0
7,14,5	BHW 62	7.2	3.1	21,14,5	FSP 4	34.9	33.3
8,17,5	BHW 215	46.0	36.2	21,15,5	FSP 4	34.9	35.6
9,15,5	BHW 211	27.3	23.8	21,17,5	FSW 167	41.3	40.8
9,35,5	SDW 252	6.3	5.3	21,29,5	AIW 60	35.3	34.2
12,14,5	BHW 198	22.0	29.0	21,31,5	AIW 264	26.4	27.7
13,28,5	SDW 33	64.0	63.0	21,36,5	AIW 183	13.2	19.2
13,28,5	SDW 256	64.3	63.0	21,45,5	AIW 301	26.6	22.5
14,19,5	BHW 27	66.0	64.5	21,46,5	AIW 302	19.7	19.1
14,22,5	SDP 4	69.0	72.1	21,46,5	AIW 300	24.3	19.1
15,22,5	SDP 4	69.0	69.8	21,52,5	YAW 89	18.0	11.3
15,29,5	SDW 253	61.4	61.2	22,15,5	FSW 169	32.6	32.0
15,36,5	AIW 315	32.5	29.5	22,29,5	AIW 298	24.3	26.8
16,32,5	AIW 291	50.4	52.7	22,35,5	AIW 255	17.0	14.9
16,42,5	AIW 294	18.1	16.4	22,41,5	AIW 306	27.2	19.0
17,21,5	SDW 154	62.6	63.1	22,46,5	AIW 230	18.4	13.0
17,44,5	AIW 295	26.7	22.2	22,51,5	YAW 104	16.6	11.4
17,46,5	AIW 247	20.3	19.0	23,17,5	FSW 168	35.7	32.4
17,50,5	YAW 98	15.5	14.5	23,25,5	MIW 21	23.2	20.4
17,53,5	YAW 93	9.1	10.2	23,34,5	AIW 260	6.8	9.1
18,12,5	FSW 179	25.3	29.4	23,53,5	YAW 117	7.4	7.4
18,26,5	SDW 258	53.7	58.9	23,54,5	YAW 96	5.2	6.2
18,34,5	AIP 1	44.2	47.3	24,17,5	FSW 173	29.8	27.8
18,55,5	YAW 108	6.3	6.6	24,21,5	MIW 19	26.9	23.8
19,31,5	AIW 263	38.2	43.3	24,50,5	YAW 94	7.7	7.2
19,42,5	AIW 293	34.0	33.2	25, 9,5	FSW 172	7.9	13.2
19,43,5	AIW 292	34.1	31.6	25,18,5	FSW 175	25.7	21.6
19,45,5	AIW 287	30.1	25.0	25,19,5	FSW 176	22.4	21.0
19,48,5	YAW 85	22.7	19.5	25,34,5	AIW 307	5.0	5.4
20,18,5	SDW 262	47.6	47.8	26,18,5	FSW 185	19.7	16.3
20,31,5	AIW 267	33.7	36.8	27,23,5	MIW 29	6.4	7.0
20,31,5	AIW 313	27.5	36.8	28,12,5	FSW 181	5.7	8.9
20,37,5	AIW 254	34.9	31.6	28,17,5	FSW 5	6.2	7.6
20,45,5	AIW 289	29.6	25.3	31,22,5	MIW 18	3.0	3.4
20,46,5	AIW 297	24.8	20.6				

Table 5. Comparison of calculated nodal head values and observed average water levels for selected wells, 1963-1976, ECAPE model

Node (i,j,k)	U.S. Geological Survey well No.	Water level, in feet above sea level: observed average	Water level, in feet above sea level: calculated at center of grid block	Node (i,j,k)	U.S. Geological Survey well No.	Water level, in feet above sea level: observed average	Water level, in feet above sea level: calculated at center of grid block
5,31,5	OSW 22	4.2	5.2	11,25,5	BMW 46	27.5	26.2
6,24,5	BMW 22	19.8	12.9	12,10,5	DGW 107	20.6	25.0
6,27,5	OSW 25	15.1	14.0	12,12,5	DGW 157	25.9	30.6
8,27,5	OSP 1	21.7	21.0	12,18,5	HJP 3	31.6	31.1
8,29,5	OSP 3	7.5	15.1	12,19,5	HJP 3	31.6	31.0
9,6,5	DGW 170	6.4	6.5	12,20,5	HJP 3	31.6	31.0
9,16,5	BMW 44	26.4	25.3	13,12,5	DGW 88	25.9	27.7
9,24,5	BMP 3	28.0	27.2	13,18,5	HJP 3	31.6	31.2
9,24,5	BMW 21	27.2	27.2	13,20,5	HJP 3	31.6	31.1
9,25,5	BMP 5	25.8	26.1	13,21,5	HJP 3	31.6	30.8
9,26,5	BMP 5	25.8	24.6	14,10,5	DGW 158	19.2	17.7
9,29,5	OSP 3	7.5	14.8	14,12,5	DGP 7	27.0	26.1
10,8,5	DGW 100	44.7	25.8	14,15,5	HJW 145	31.7	29.0
10,8,5	DGW 114	28.3	25.8	14,25,5	HJW 141	19.2	20.5
10,11,5	DGW 143	30.5	33.7	15,22,5	HJW 150	30.3	28.3
10,12,5	DGW 123	32.8	33.6	15,25,5	HJW 151	11.4	19.6
10,12,5	DGW 144	30.6	33.6	16,12,5	DGW 160	8.2	14.2
10,28,5	OSW 24	18.5	17.8	17,18,5	HJW 148	20.8	22.4
11,8,5	DGW 124	16.9	21.9	17,26,5	CGP 4	10.4	16.3
11,8,5	DGP 4	24.5	21.9	18,25,5	CGW 176	12.5	16.8
11,9,5	DGW 135	16.7	25.6	18,25,5	CGP 5	12.8	16.8
11,10,5	DGP 2	20.3	29.3	18,26,5	CGP 3	12.7	15.6
11,10,5	DGP 3	20.9	29.3	18,28,5	CGW 177	11.9	11.9
11,11,5	DGW 146	25.7	31.9	19,23,5	CGW 138	11.3	12.9
11,11,5	DGP 6	27.6	31.9	19,27,5	CGP 1	14.8	9.7
11,18,5	BMW 45	31.4	30.6				

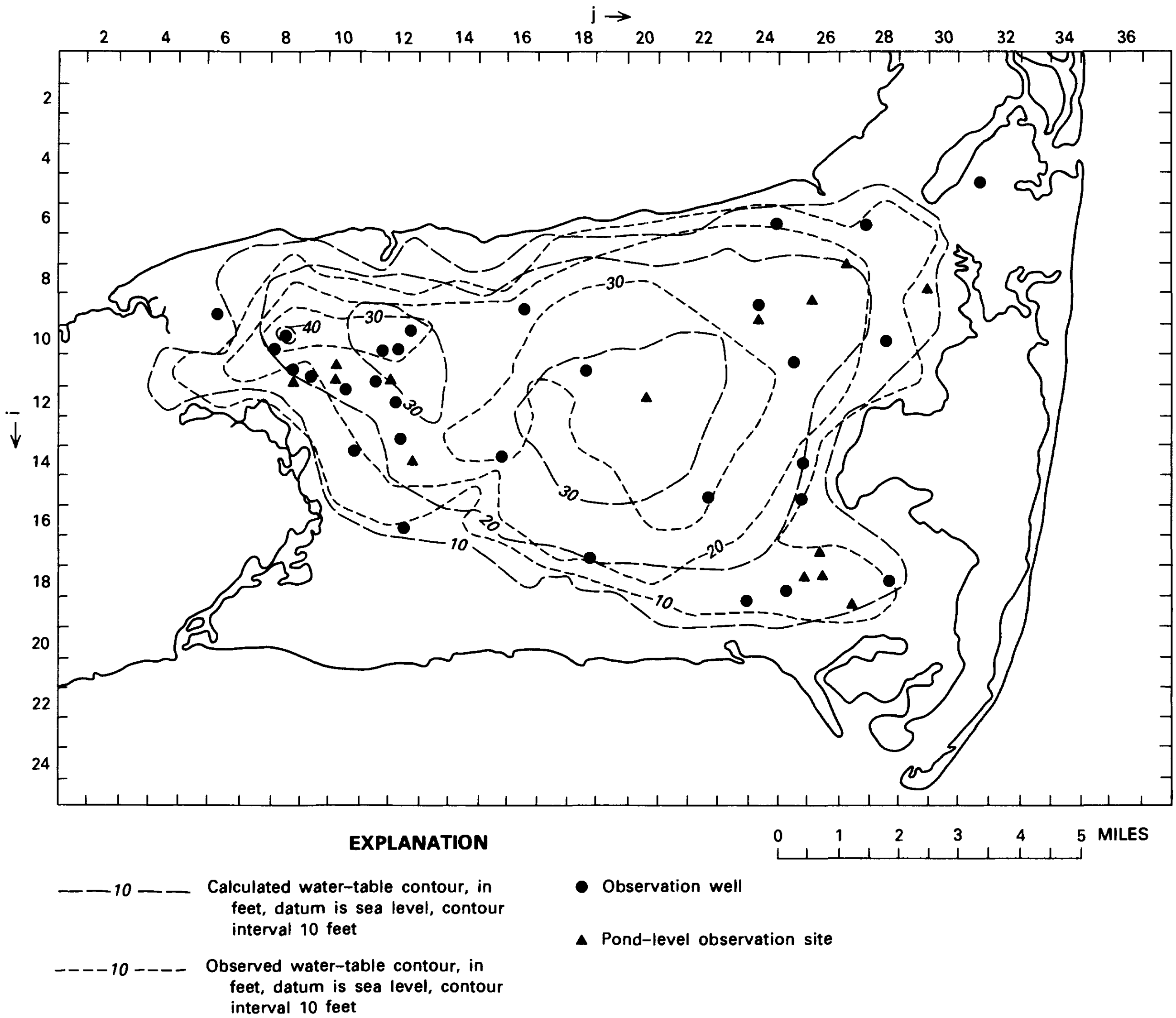


Figure 14. Calculated and observed average water table, 1963–1976, ECAPE model.

The accuracy of the model results is limited by the accuracy of the input data that describe aquifer properties, boundary conditions, and recharge rates. As additional data become available, the models should be recalibrated to improve model accuracy. Continued monitoring of water-level changes, streamflow, and movement of the boundary between fresh and saline ground water in response to natural and man-caused changes in hydrologic stresses will help improve the aquifer models and make them more useful.

The large scale of the models precludes detailed analyses of hydrologic conditions for local areas. More detailed, smaller scale models of local areas can be constructed using similar principles if sufficient data are available.

Mass-balance calculations for the five modeled areas indicate a total steady-state natural and artificial recharge rate of 412 ft^3/s; a total rate of withdrawal by wells of 25 ft^3/s; and a total discharge rate to streams, marshes, and the ocean of 387 ft^3/s.

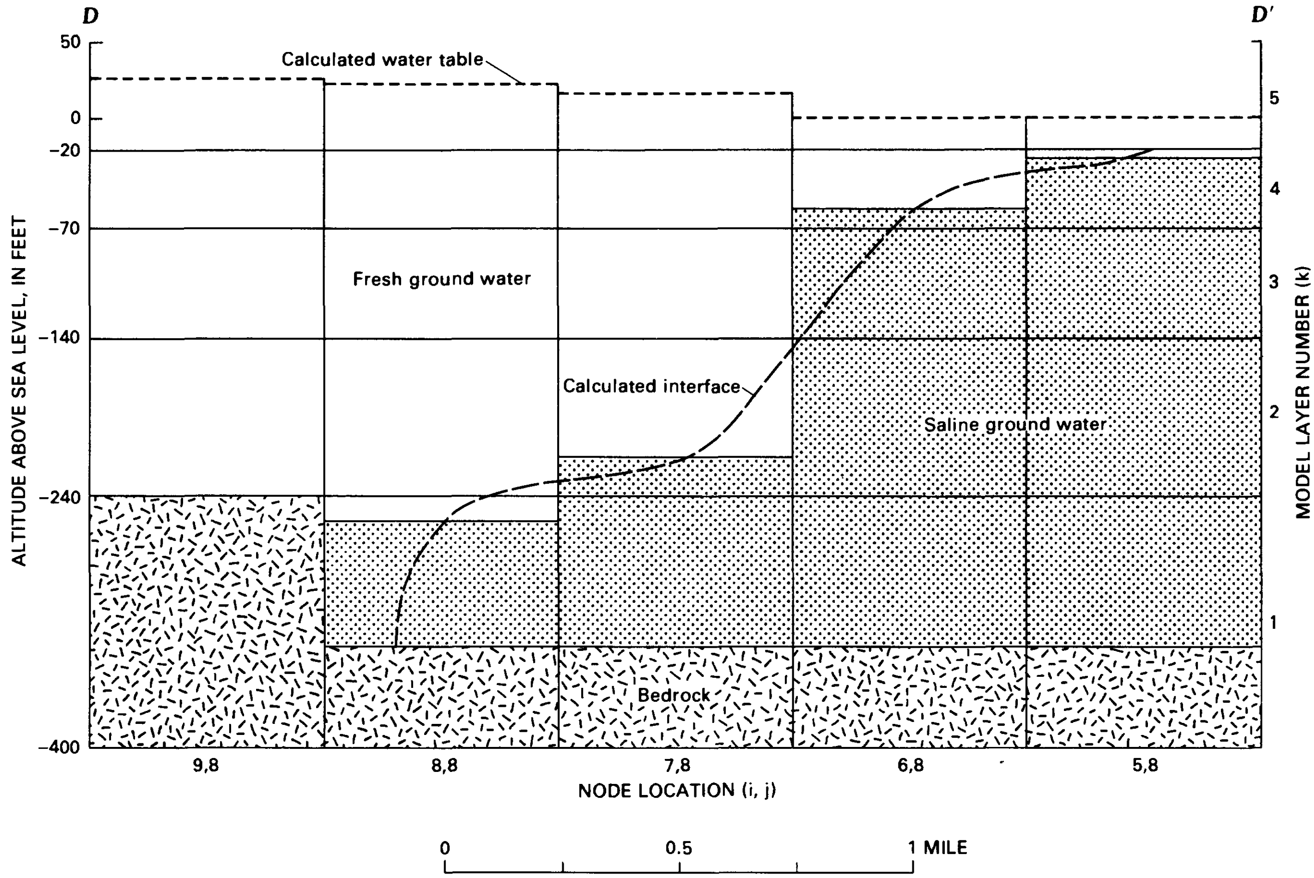
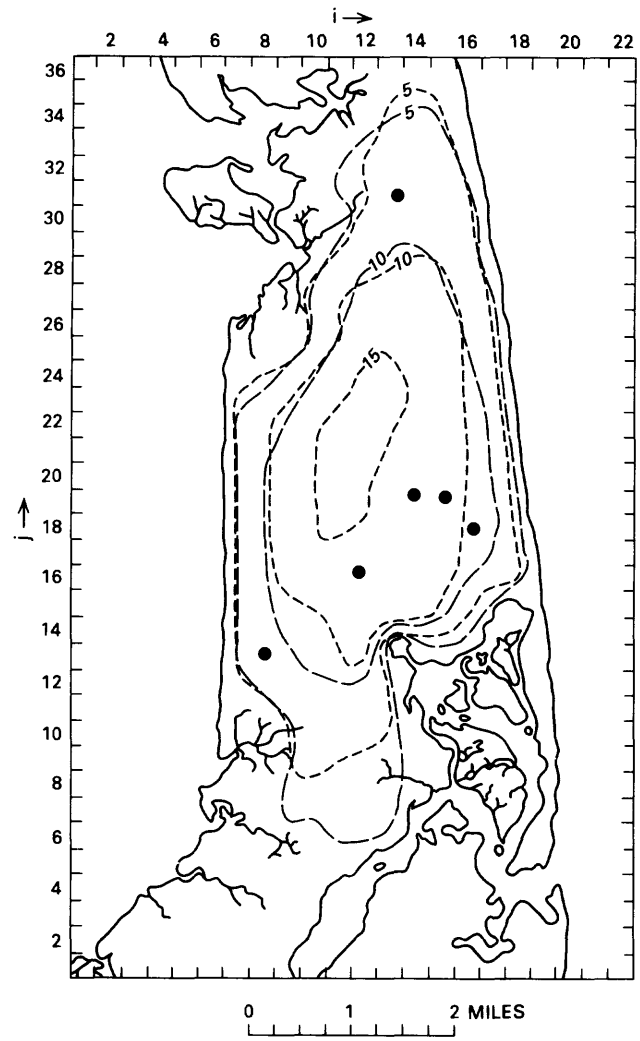


Figure 15. Calculated water table and interface between fresh and saline ground water. Line of cross section *D–D'* is shown in figure 4.



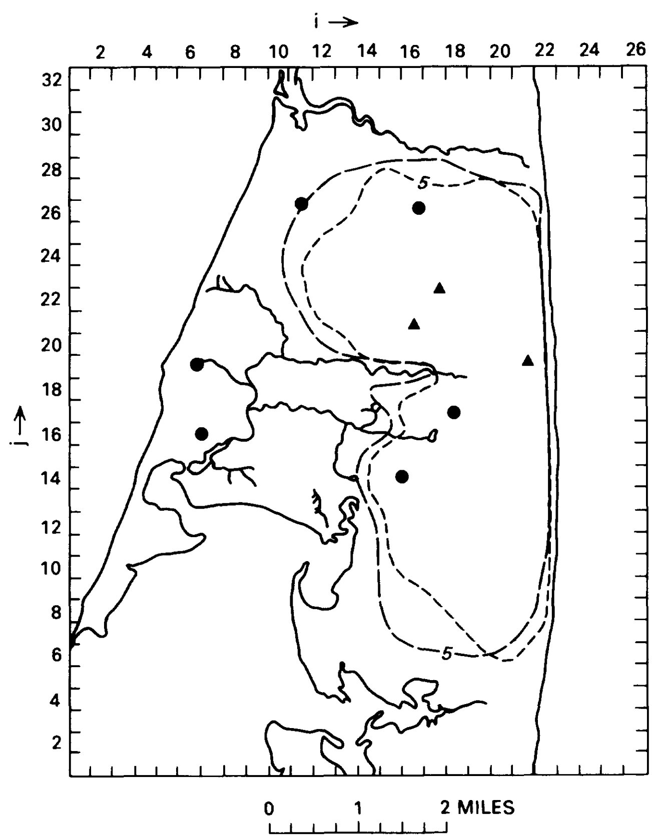
EXPLANATION

——5—— Calculated water-table contour, in feet, datum is sea level, contour interval 5 feet

----5---- Observed average water-table contour, in feet, datum is sea level, contour interval 5 feet

● Observation well

Figure 16. Calculated and observed average water table, 1963–1976, ESTHM model.



EXPLANATION

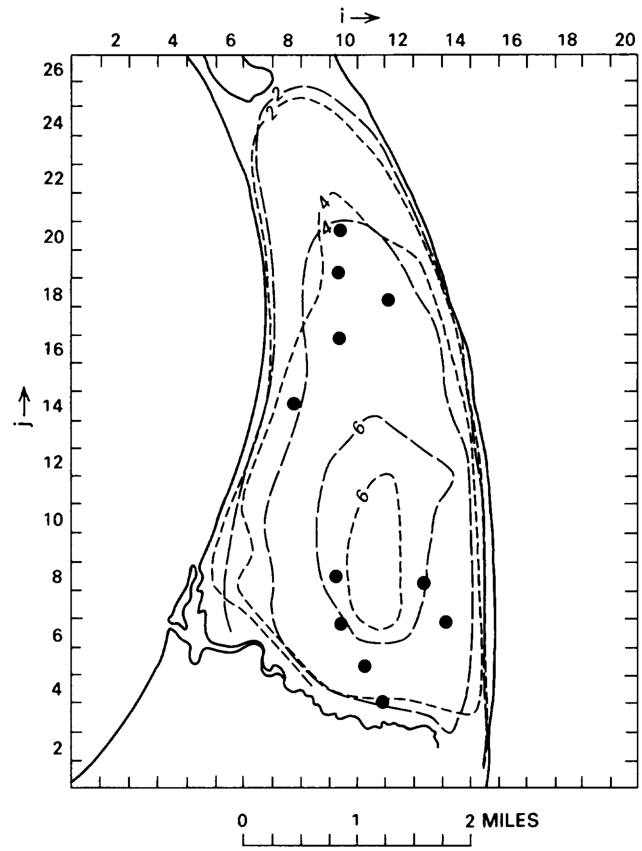
——5—— Calculated water-table contour, in feet, datum is sea level, contour interval 5 feet

----5---- Observed average water-table contour, in feet, datum is sea level, contour interval 5 feet

● Observation well

▲ Pond-level observation site

Figure 17. Calculated and observed average water table, 1963–1976, WLFLT model.



EXPLANATION

——2—— Calculated water-table contour, in feet, datum is sea level, contour interval 2 feet

----2---- Observed average water-table contour, in feet, datum is sea level, contour interval 2 feet

● Observation well

Figure 18. Calculated and observed average water table, 1963–1976, TRURO model.

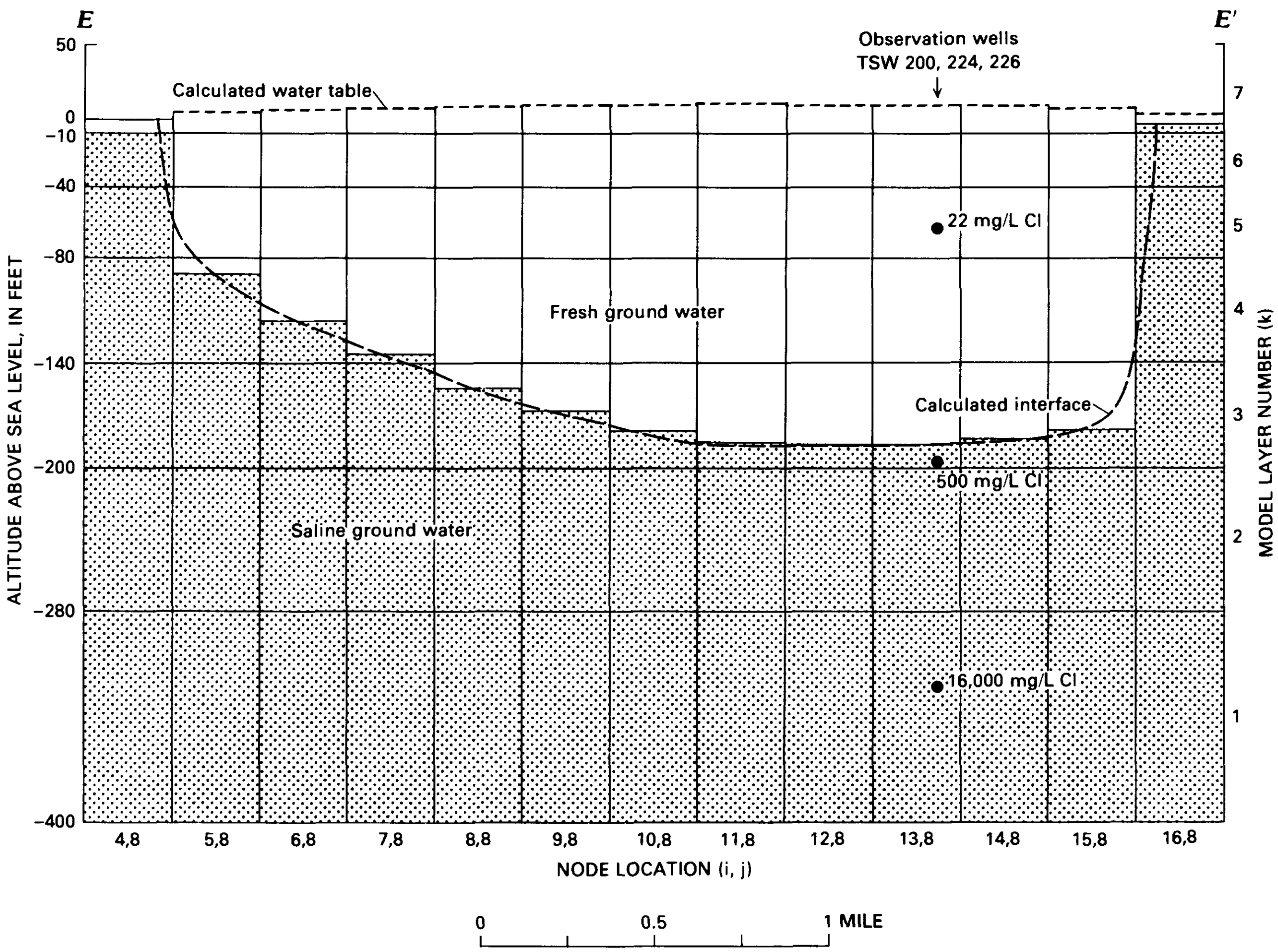


Figure 19. Calculated water table and interface between fresh and saline ground water, and chloride (Cl) content, in milligrams per liter (mg/L), of water samples collected from wells on December 5, 1978. Line of cross section *E–E'* is shown in figure 7.

Table 6. Comparison of calculated nodal head values and observed average water levels for selected wells, 1963-1976, ESTHM model

Node (i,j,k)	U.S. Geological Survey well No.	Water level, in feet above sea level: observed average	Water level, in feet above sea level: calculated at center of grid block	Node (i,j,k)	U.S. Geological Survey well No.	Water level, in feet above sea level: observed average	Water level, in feet above sea level: calculated at center of grid block
8,13,7	EGW 37	8.0	8.2	14,19,7	EGW 36	13.5	13.9
12,16,7	EGW 39	13.8	13.5	15,19,7	EGW 32	12.8	13.1
13,31,7	WNW 17	8.7	8.3	16,18,7	EGW 40	8.6	11.7

Table 7. Comparison of calculated nodal head values and observed average water levels for selected wells, 1963-1976, WLFLT model.

Node (i,j,k)	U.S. Geological Survey well No.	Water level, in feet above sea level: observed average	Water level, in feet above sea level: calculated at center of grid block	Node (i,j,k)	U.S. Geological Survey well No.	Water level, in feet above sea level: observed average	Water level, in feet above sea level: calculated at center of grid block
6,16,7	WNW 78	2.7	3.4	16,26,7	TSW 198	7.6	7.1
11,26,7	TSW 216	4.1	5.3	17,22,7	TSP 16	7.6	7.9
15,14,7	WNW 30	6.6	6.7	18,17,7	WNW 34	8.0	8.0
16,14,7	WNW 30	6.6	7.4	21,19,7	TSP 18	6.4	6.5
16,21,7	TSP 17	6.8	7.2				

Table 8. Comparison of calculated nodal head values and observed average water levels for selected wells, 1963-1976, TRURO model

Node (i,j,k)	U.S. Geological Survey well No.	Water level, in feet above sea level: observed average	Water level, in feet above sea level: calculated at center of grid block	Node (i,j,k)	U.S. Geological Survey well No.	Water level, in feet above sea level: observed average	Water level, in feet above sea level: calculated at center of grid block
8,14,7	TSW 157	4.4	3.8	11,5,7	TSW 218	5.2	5.4
10,6,7	TSW 176	5.6	5.6	12,3,7	TSW 181	4.7	3.6
10,8,7	TSW 170	5.9	6.1	12,4,7	TSW 181	4.7	4.9
10,16,7	TSW 89	4.4	5.2	12,18,7	TSW 134	4.9	4.5
10,17,7	TSW 89	4.4	5.0	13,8,7	TSW 203	5.7	6.0
10,19,7	TSW 136	4.3	4.6	14,6,7	TSW 174	5.6	5.0
10,20,7	TSW 126	4.2	4.3				

REFERENCES

Burns, A. W., Frimpter, M. H., and Willey, R. E., 1975, Evaluation of data availability and examples of modeling for ground-water management on Cape Cod, Massachusetts: U.S. Geological Survey Water Resources Investigation 16–75, 22 p.

Getzen, R. T., 1977, Analog-model analysis of regional three-dimensional flow in the ground-water reservoir of Long Island, New York: U.S. Geological Survey Professional Paper 982, 49 p.

Guswa, J. H., and Londquist, C. J., 1976, Potential for development of ground water at a test site near Truro, Massachusetts: U.S. Geological Survey Open-File Report 76–614, 22 p.

Hubbert, M. K., 1940, The theory of ground-water motion: Journal of Geology, v. 48, no. 8, p. 785–944.

Meyer, William, Reussow, J. P., Gillies, D. C., and Shampine, W. J., 1975, Availability of ground water in Marion County, Indiana: U.S. Geological Survey Open-File Report 75–312, 87 p.

Oldale, R. N., 1969, Seismic investigations on Cape Cod, Martha's Vineyard, and Nantucket, Massachusetts, and a topographic map of the basement surface from Cape Cod Bay to the Islands: U.S. Geological Survey Professional Paper 650–B, p. B122–B127.

———, 1976, Notes on the generalized geologic map of Cape Cod: U.S. Geological Survey Open-File Report 76–765, 23 p.

Palmer, C. D., 1977, Hydrogeological implications of various wastewater management proposals for the Falmouth area of Cape Cod, Massachusetts: Pennsylvania State University, unpublished M.S. thesis, 142 p.

Prickett, T. A., 1975, Modeling techniques for groundwater evaluation, in Chow, V. T., ed., Advances in hydrosciences, v. 10: New York, Academic Press, p. 1–143.

Stone, H. L., 1968, Iterative solution of implicit approximations of multidimensional partial differential equations: Society of Industrial and Applied Mathematics, Journal of Numerical Analysis, v. 5, no. 3, p. 530–558.

Strahler, A. N., 1972, The environmental impact of ground water use on Cape Cod: Impact Study III, Association for the Preservation of Cape Cod: Orleans, Mass., 68 p.

Theis, C. V., 1963, Estimating the transmissibility of a water-table aquifer from the specific capacity of a well, in Bentall, Ray, Methods of determining permeability, transmissibility, and drawdown: U.S. Geological Survey Water-Supply Paper 1536–I, p. 332–336.

Thornthwaite, C. W., and Mather, J. R., 1957, Instructions and tables for computing potential evapotranspiration and the water balance: Drexel Institute of Technology Publications in Climatology, v. 10, no. 3, p. 185–311.

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, 30 p.

Trescott, P. C., and Larson, S. P., 1976, Supplement to Open-File Report 75–438, Documentation of finite-difference model for simulation of three-dimensional ground-water flow: U.S. Geological Survey Open-File Report 76–591, 17 p.

Trescott, P. C., Pinder, G. F., and Larson, S. P., 1976, Finite-difference model for aquifer simulation in two dimensions with results of numerical experiments: U.S. Geological Survey Techniques of Water Resources Investigations, Chapter C1, Book 7, p. 42.

U.S. Environmental Protection Agency, 1978, Water Quality Management Plan/EIS for Cape Cod: Draft plan/Environmental Impact Statement, 397 p

SUPPLEMENTAL DATA-MODEL DATA, INPUT DOCUMENTATION, AND SOURCE CODE

Table 9. Values for transmissivity, average hydraulic conductivity, and ratio of lateral to vertical hydraulic conductivity used in calibrated models.

(*Values apply uniformly within the modeled area identified in figs. 3-7 of the report.)

Model	Layer	Transmissivity (T) for each node in layer (ft^2/d)	Average lateral hydraulic conductivity for each node in layer (ft/d)	Ratio of lateral to vertical hydraulic conductivity
WCAPE	1	figure 20	T for node/160	*10
	2	figure 21	T for node/100	figure 25
	3	figure 22	T for node/70	figure 26
	4	figure 23	T for node/50	figure 27
	5	NA	figure 24	figure 28
ECAPE	1	figure 29	T for node/160	*10
	2	figure 30	T for node/100	figure 34
	3	figure 31	T for node/70	figure 35
	4	figure 32	T for node/50	figure 36
	5	NA	figure 33	figure 37
ESTHM	1	*20,250	*135	*10
	2	*20,250	*135	*10
	3	*13,500	*135	*10
	4	*12,150	*135	*10
	5	figure 38	T for node/60	figure 41
	6	figure 39	T for node/40	figure 42
	7	NA	figure 40	figure 43
WLFLT	1	*9,600	*80	*10
	2	*6,400	*80	*10
	3	*4,800	*80	*10
	4	figure 44	T for node/60	*10
	5	figure 45	T for node/40	figure 48
	6	figure 46	T for node/30	figure 49
	7	NA	figure 47	figure 50
TRURO	1	*10,800	*90	*10
	2	*7,200	*90	*10
	3	*5,400	*90	*10
	4	*7,200	*120	*10
	5	figure 51	T for node/40	figure 54
	6	figure 52	T for node/30	figure 55
	7	NA	figure 53	*10

TK arrays, ESTHM, layers 1-2, 2-3, and 3-4 were multiplied by 0.45.
TK array, WLFLT, layers 3-4 was multiplied by 0.89.
TK arrays, TRURO, layers 1-2, 2-3, and 3-4 were multiplied by 0.9.

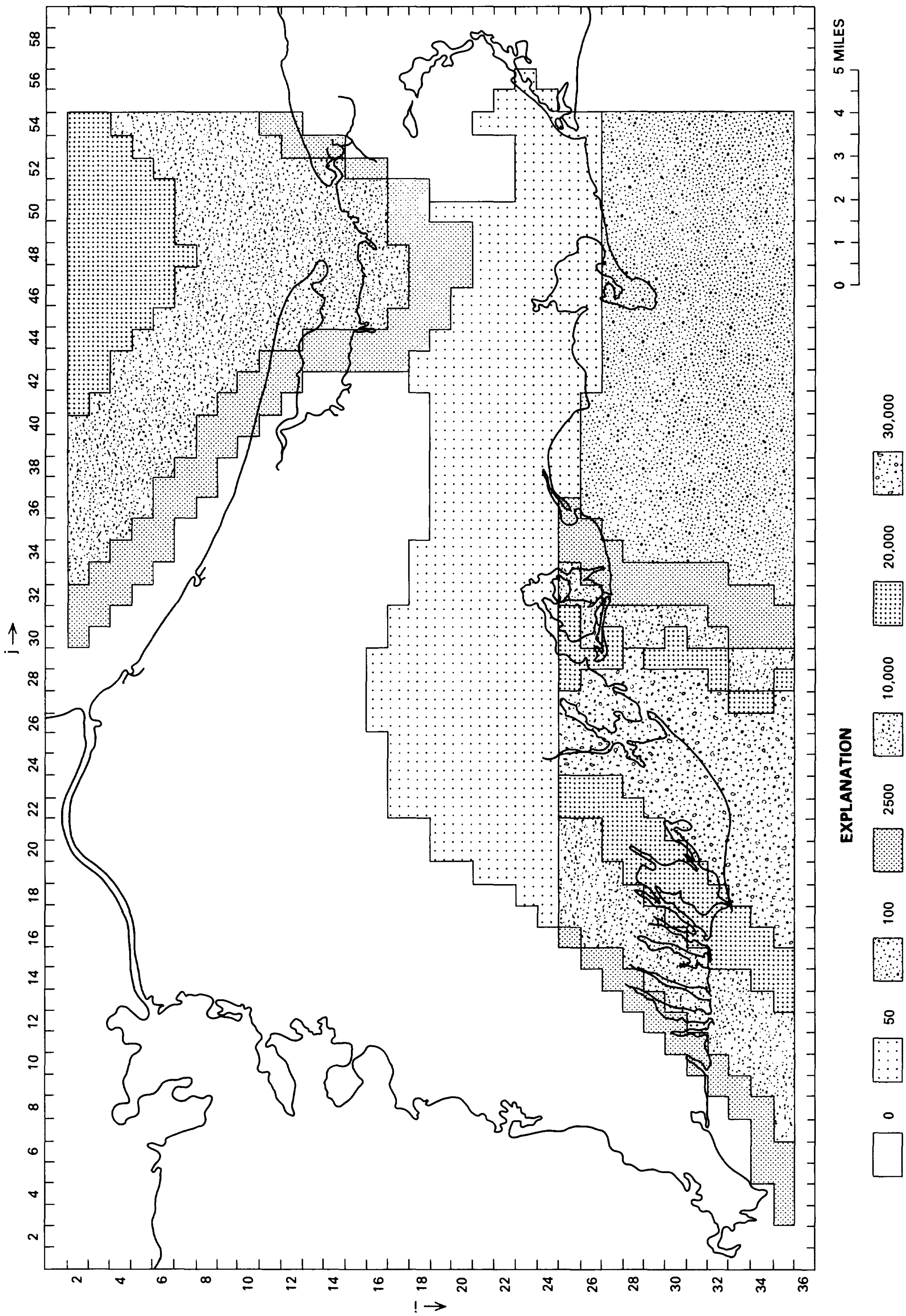


Figure 20. Transmissivity, in feet squared per day, WCAPE model, layer 1.

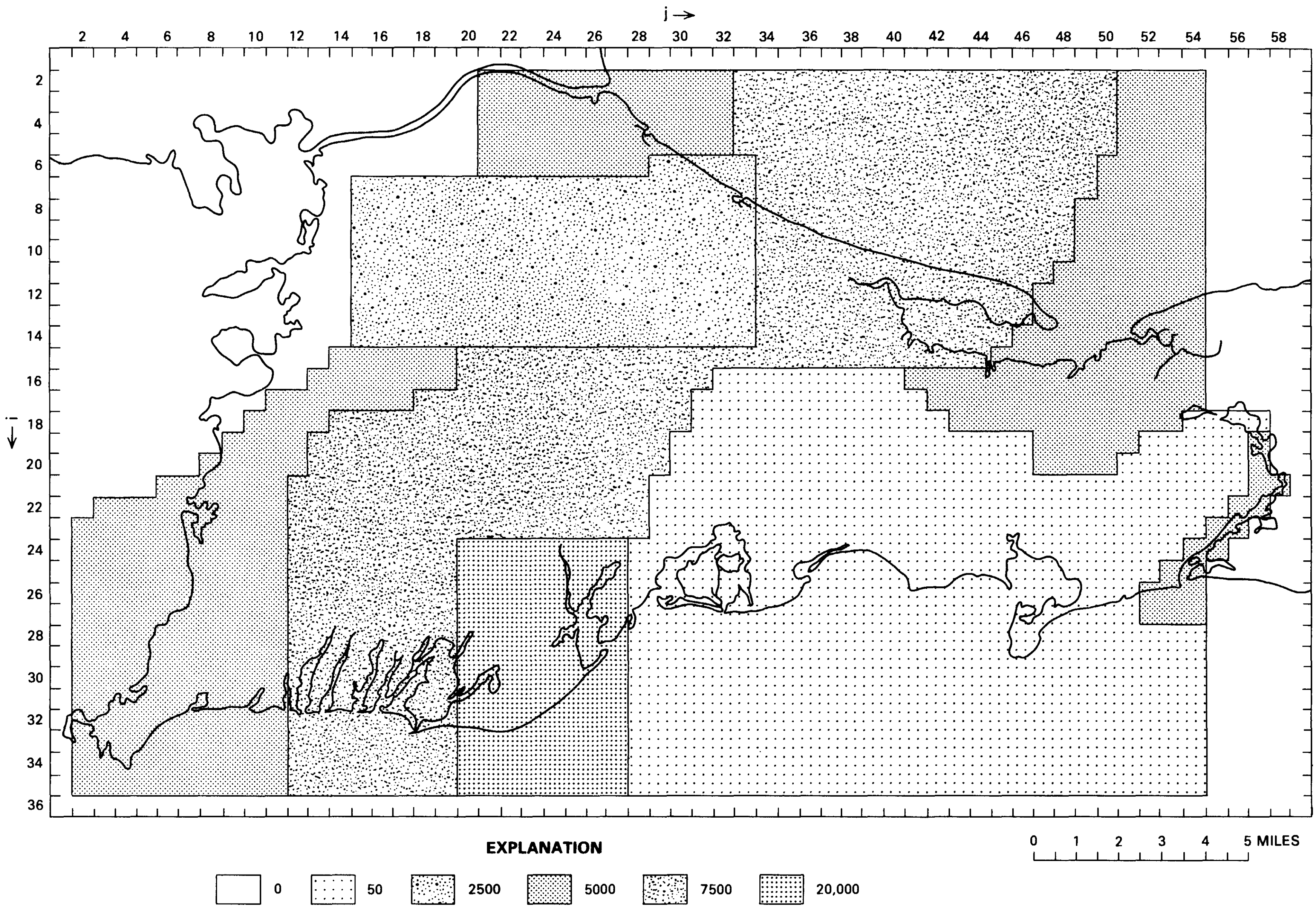


Figure 21. Transmissivity, in feet squared per day, WCAPE model, layer 2.

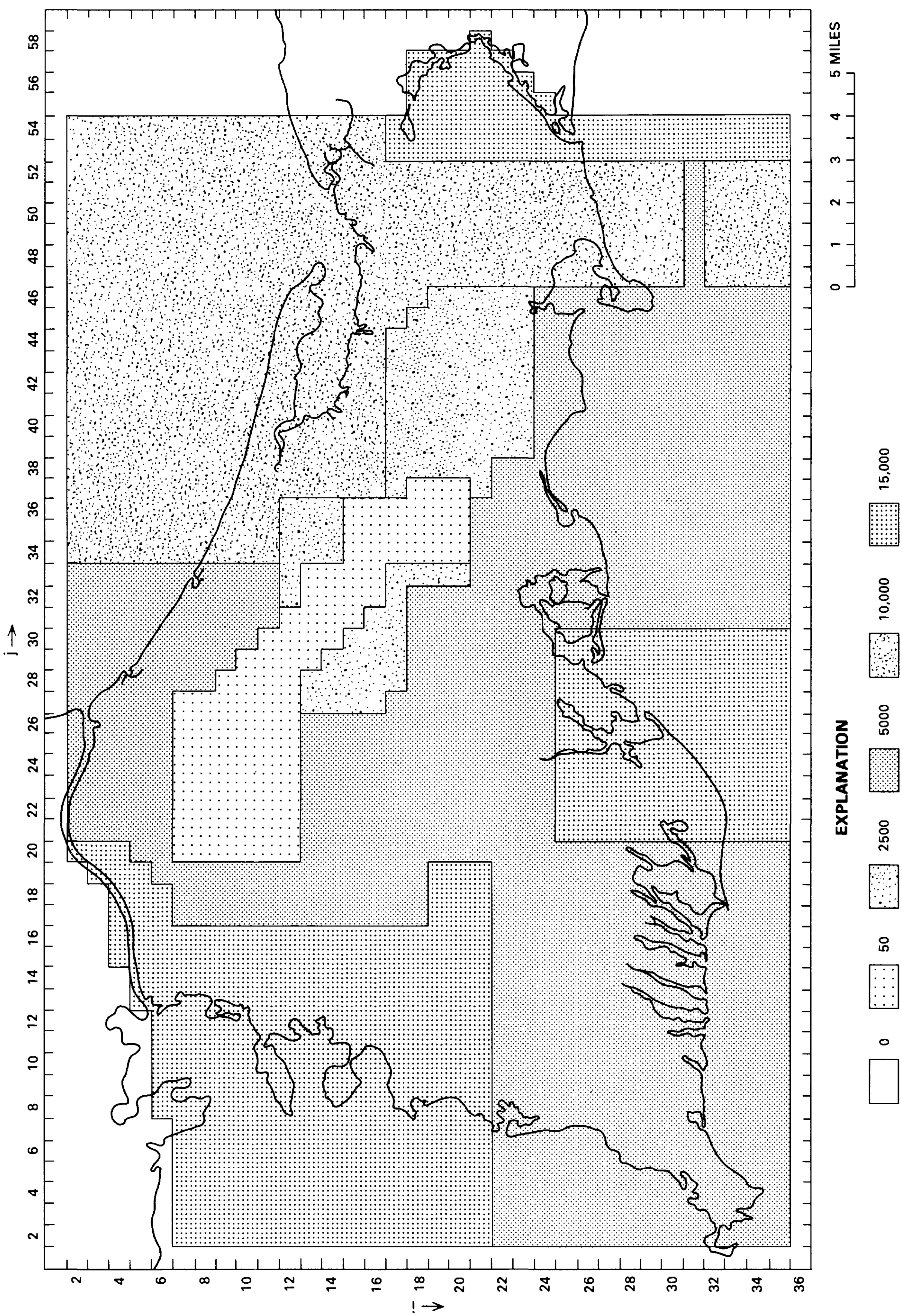


Figure 22. Transmissivity, in feet squared per day, WCAPE model, layer 3.

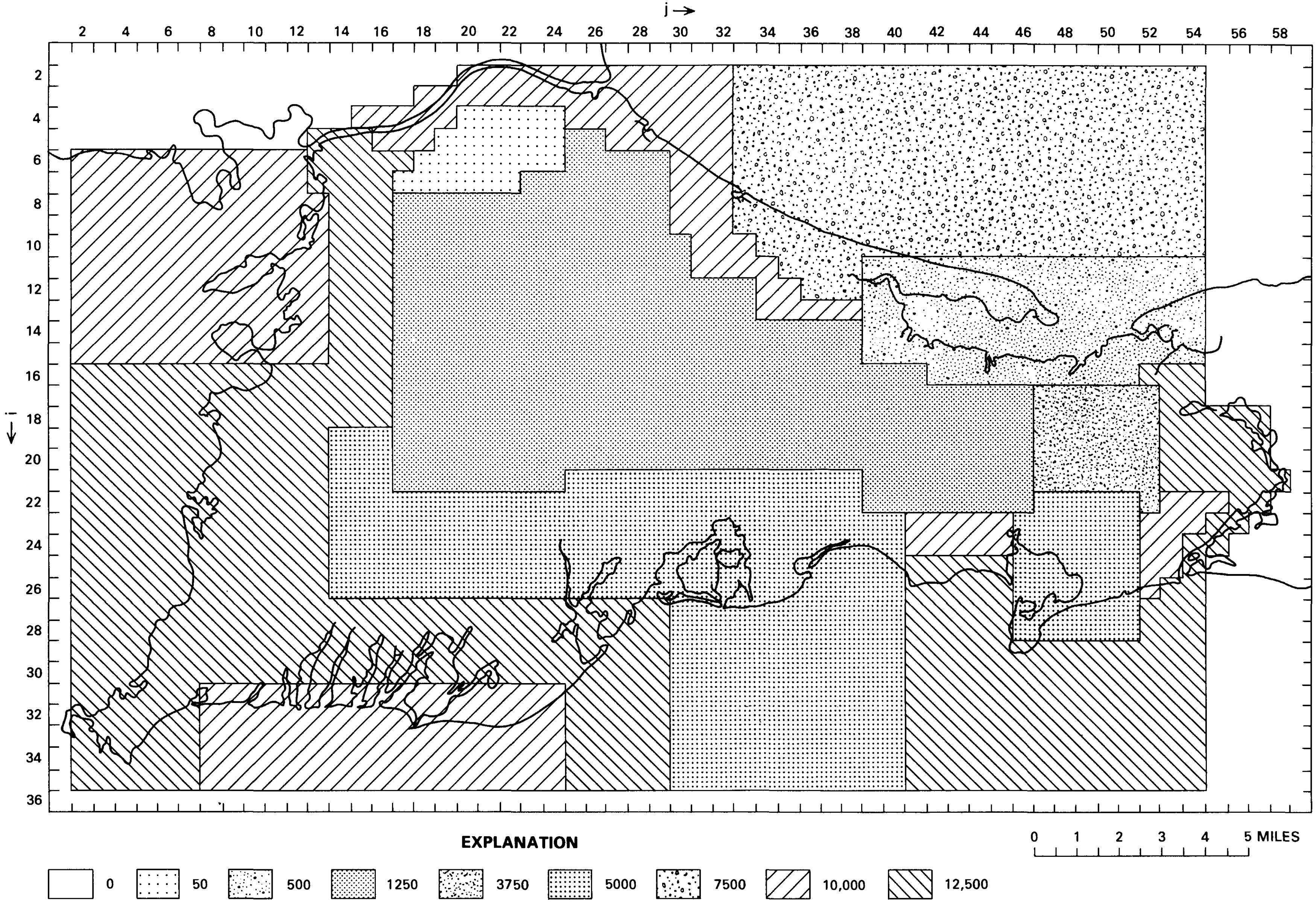


Figure 23. Transmissivity, in feet squared per day, WCAPE model, layer 4.

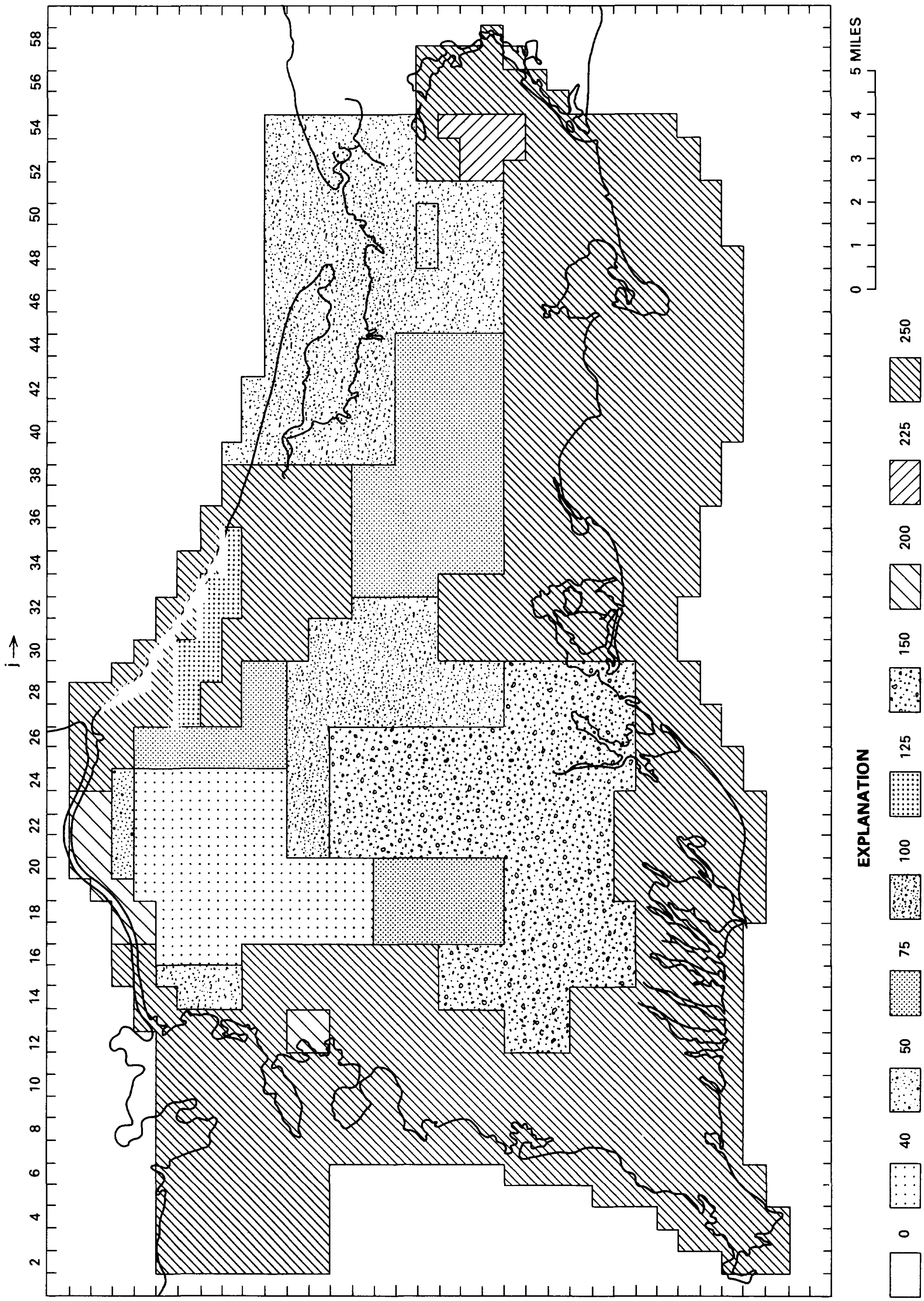


Figure 24. Hydraulic conductivity, in feet per day, WCAPE model, layer 5.

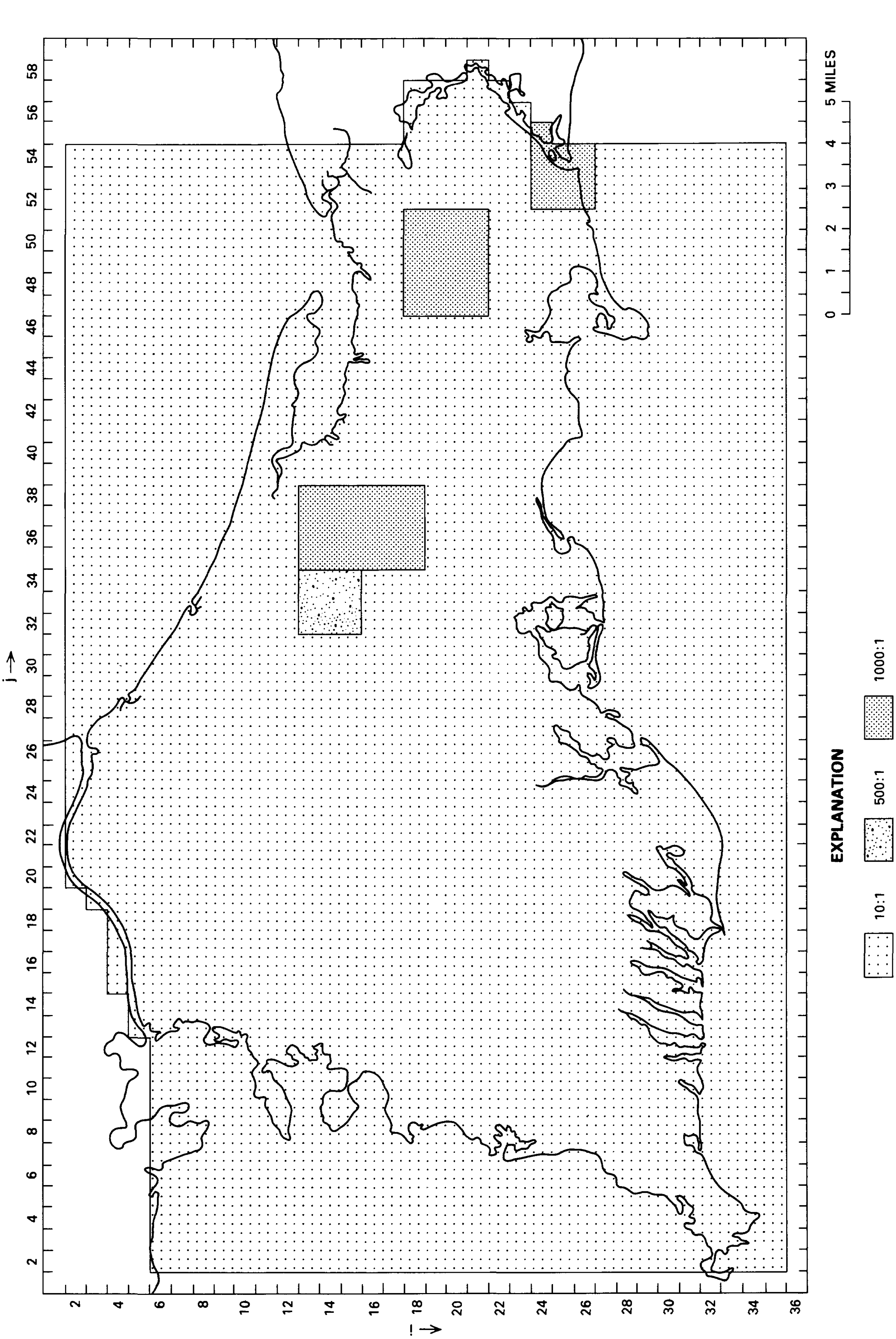


Figure 25. Ratio of lateral to vertical hydraulic conductivity, WCAPE model, layer 2.

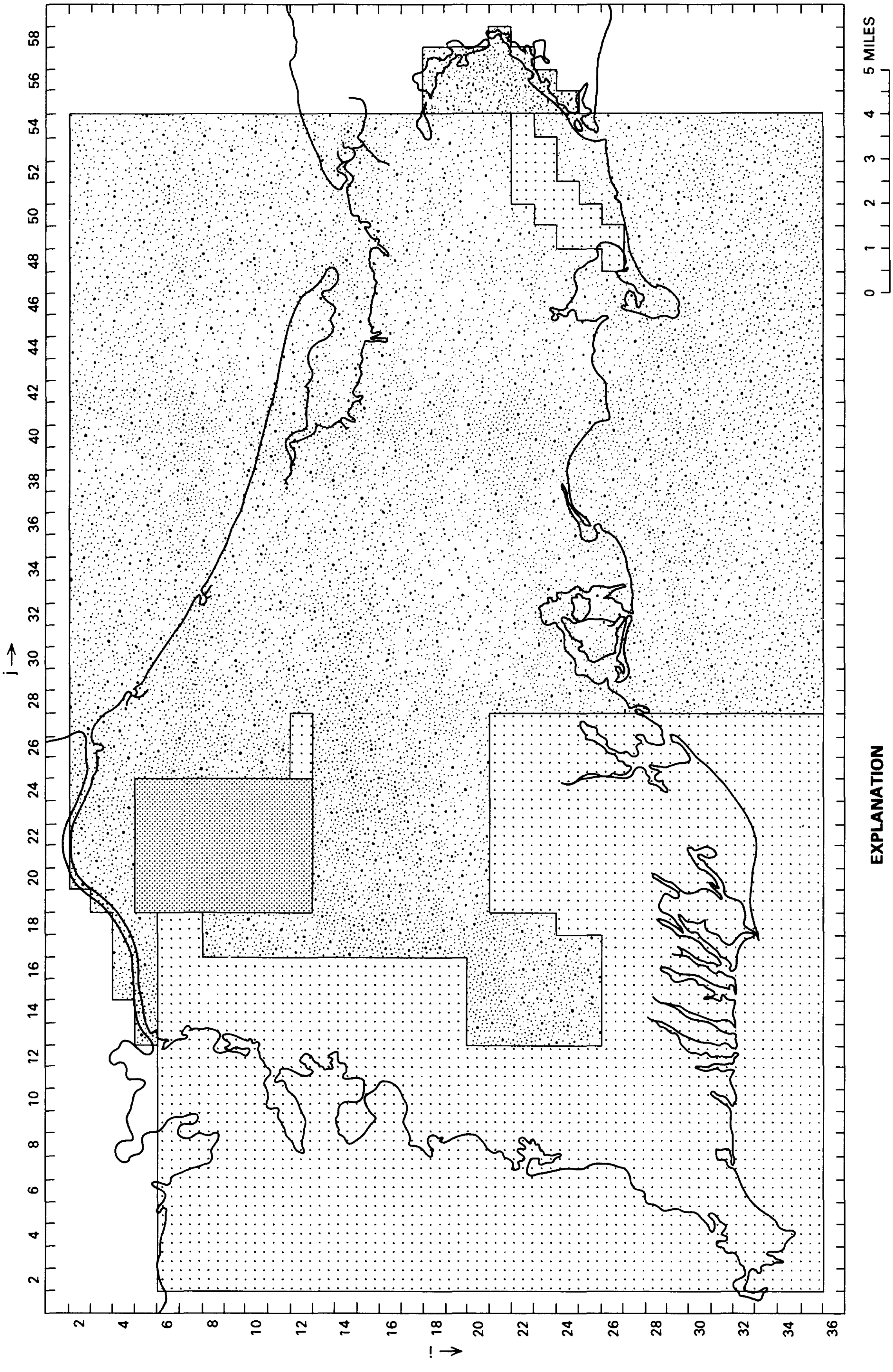


Figure 26. Ratio of lateral to vertical hydraulic conductivity, WCAPE model, layer 3.

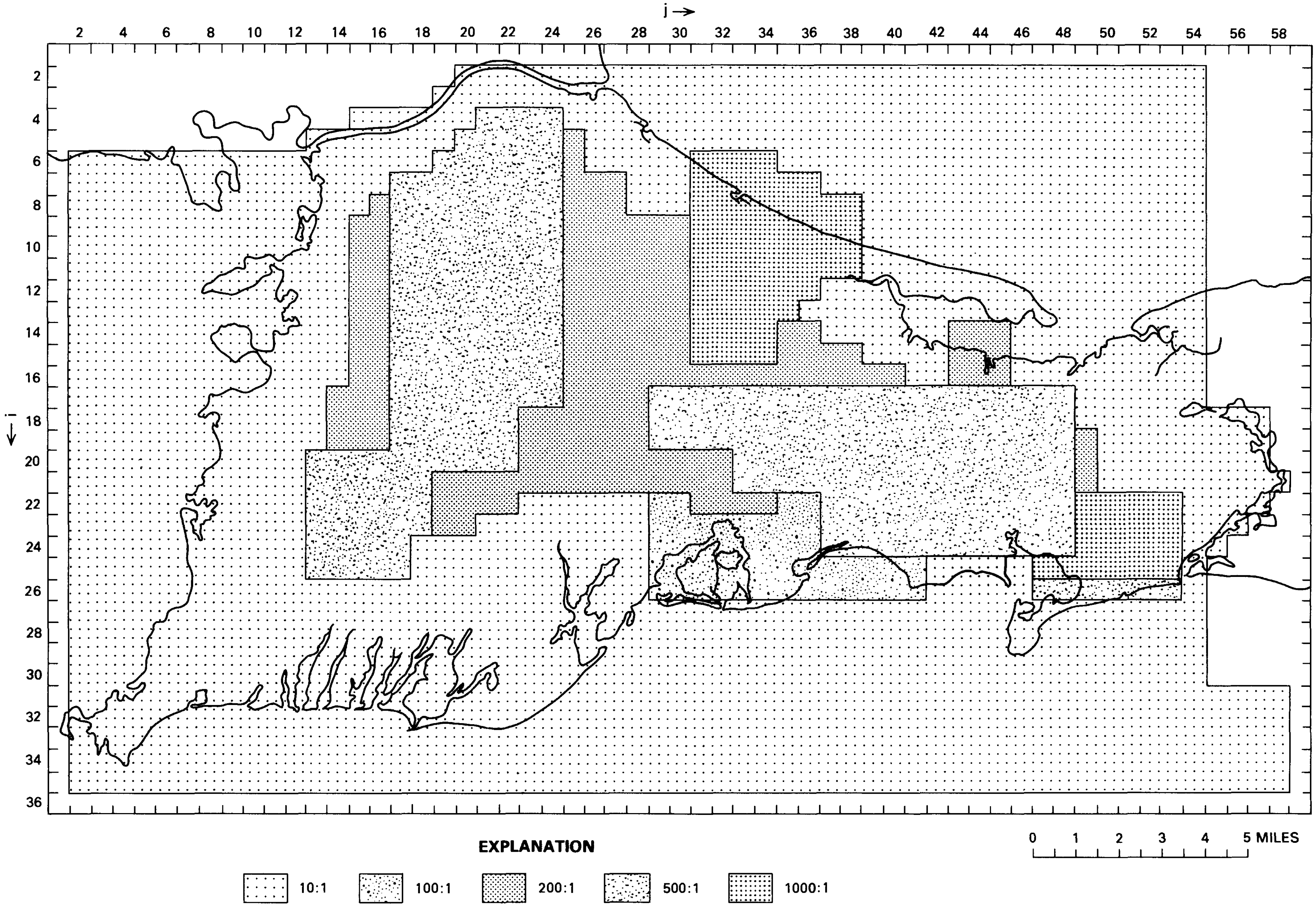


Figure 27. Ratio of lateral to vertical hydraulic conductivity, WCAPE model, layer 4.

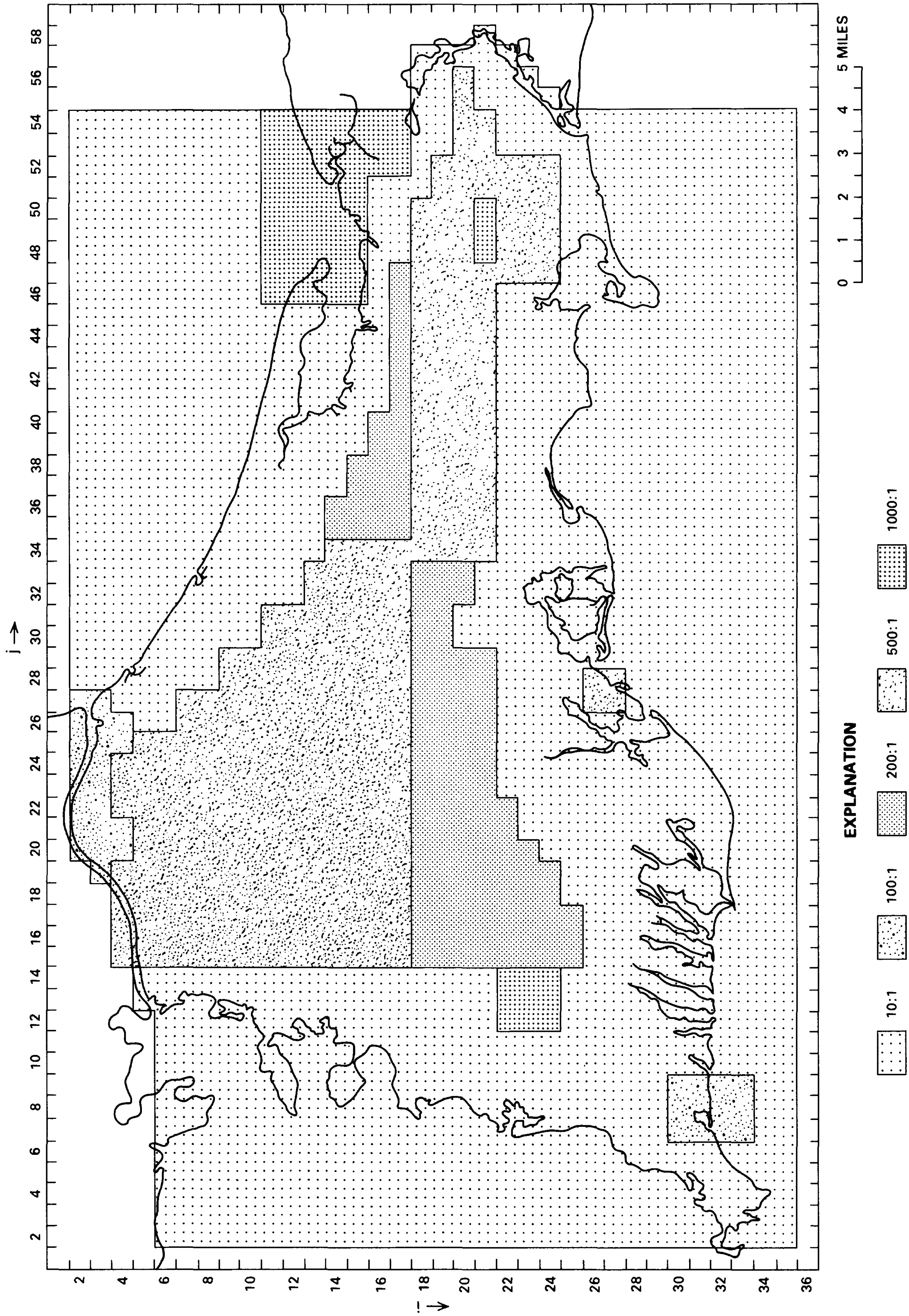


Figure 28. Ratio of lateral to vertical hydraulic conductivity, WCAPE model, layer 5.

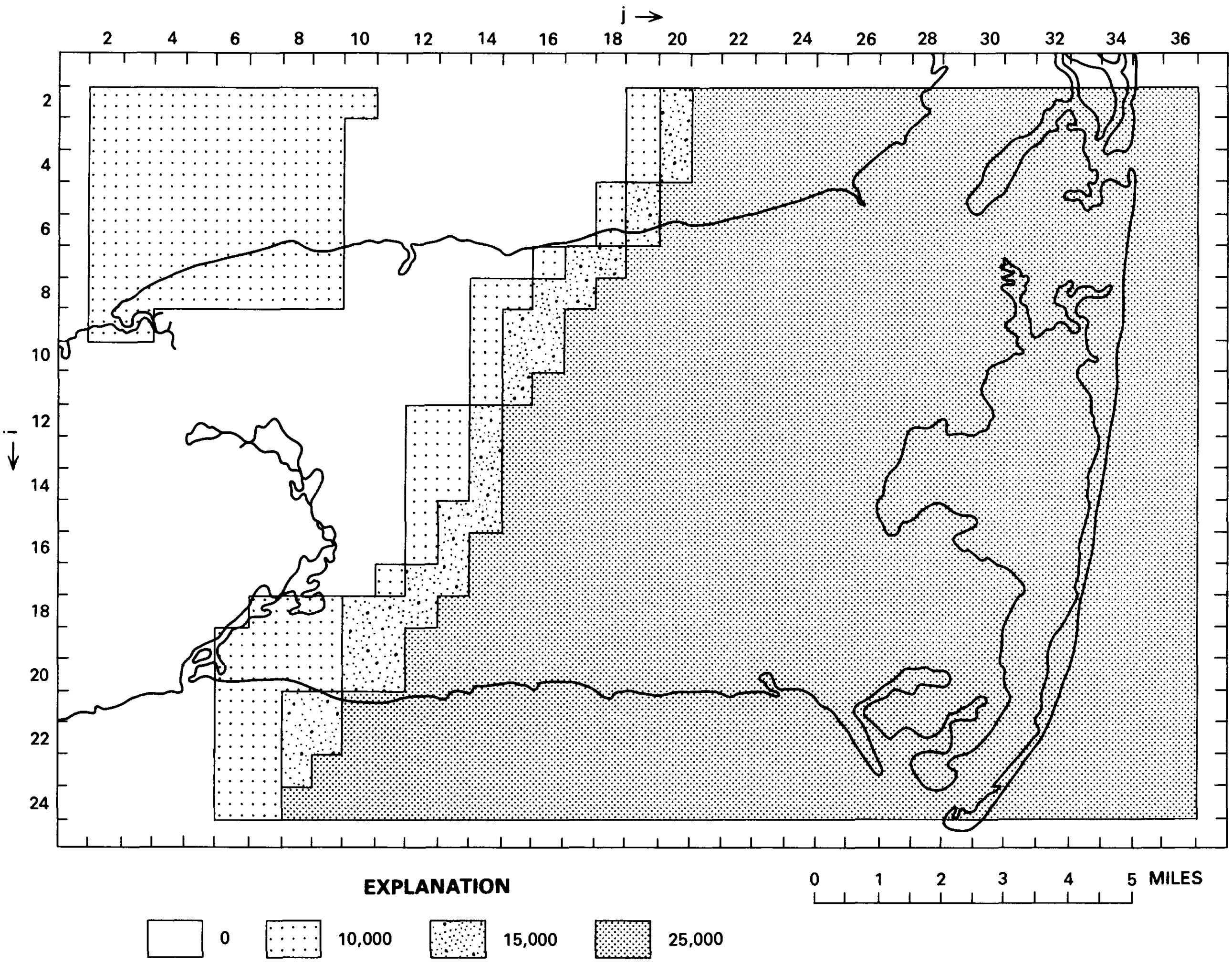


Figure 29. Transmissivity, in feet squared per day, ECAPE model, layer 1.

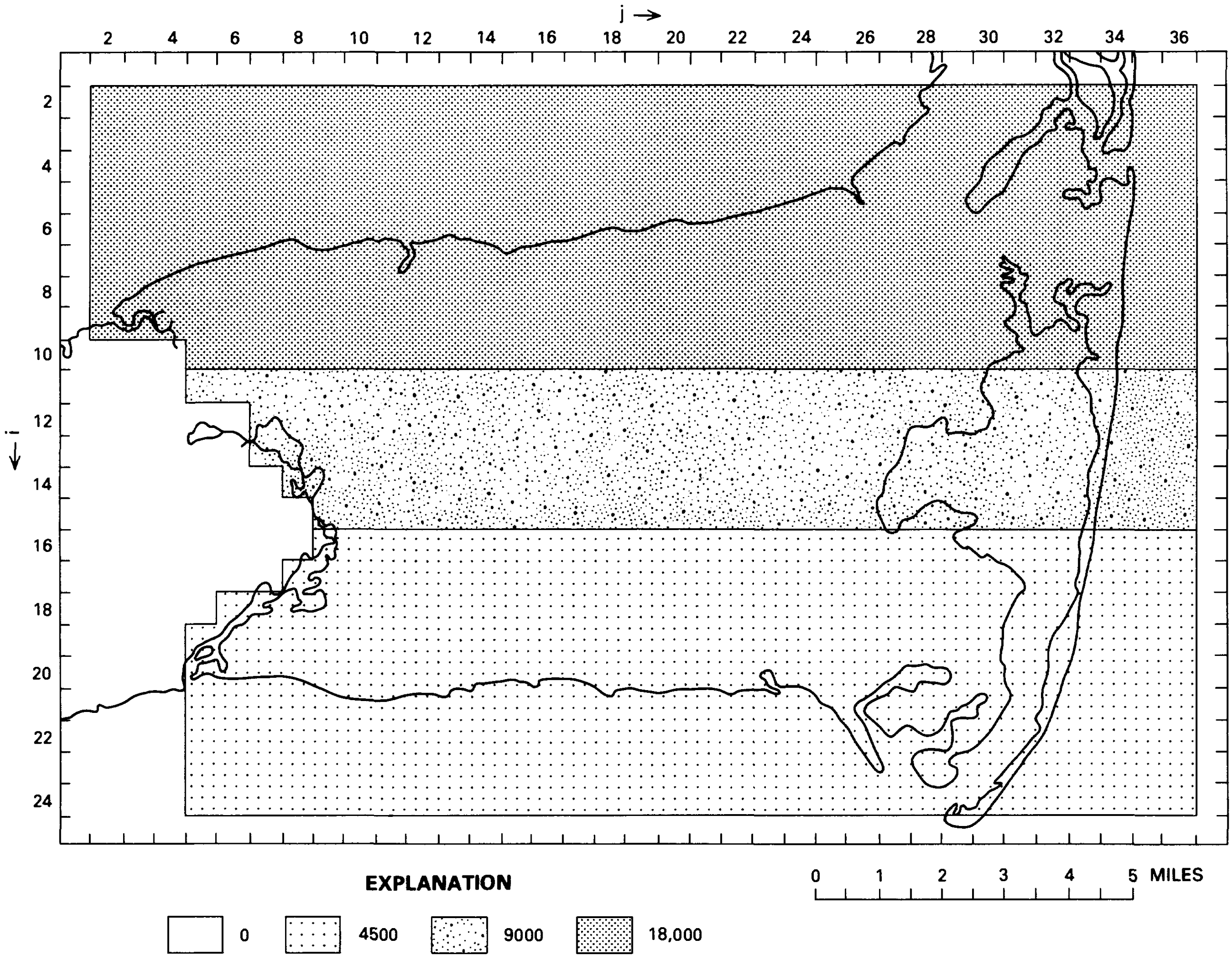


Figure 30. Transmissivity, in feet squared per day, ECAPE model, layer 2.

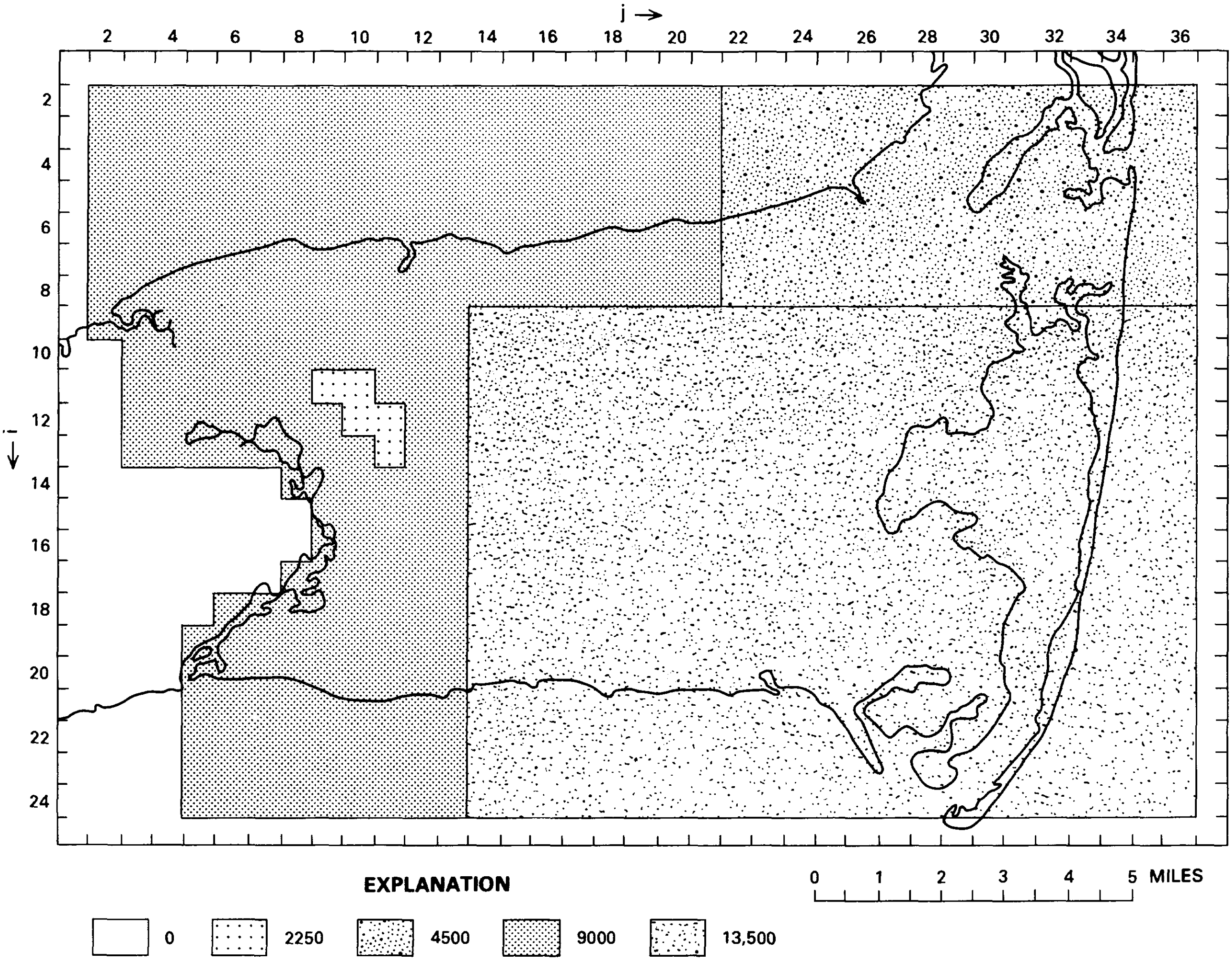


Figure 31. Transmissivity, in feet squared per day, ECAPE model, layer 3.

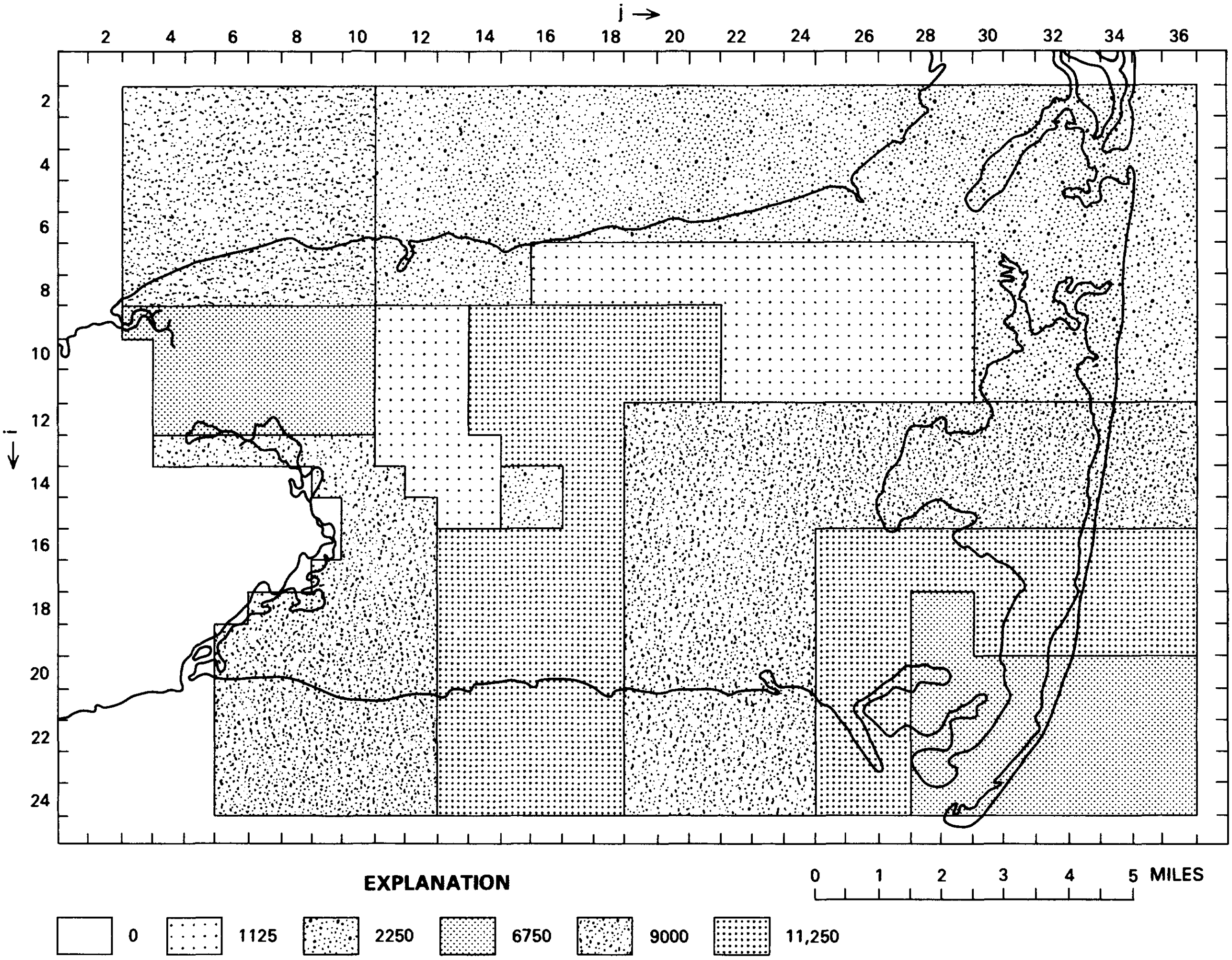


Figure 32. Transmissivity, in feet squared per day, ECAPE model, layer 4.

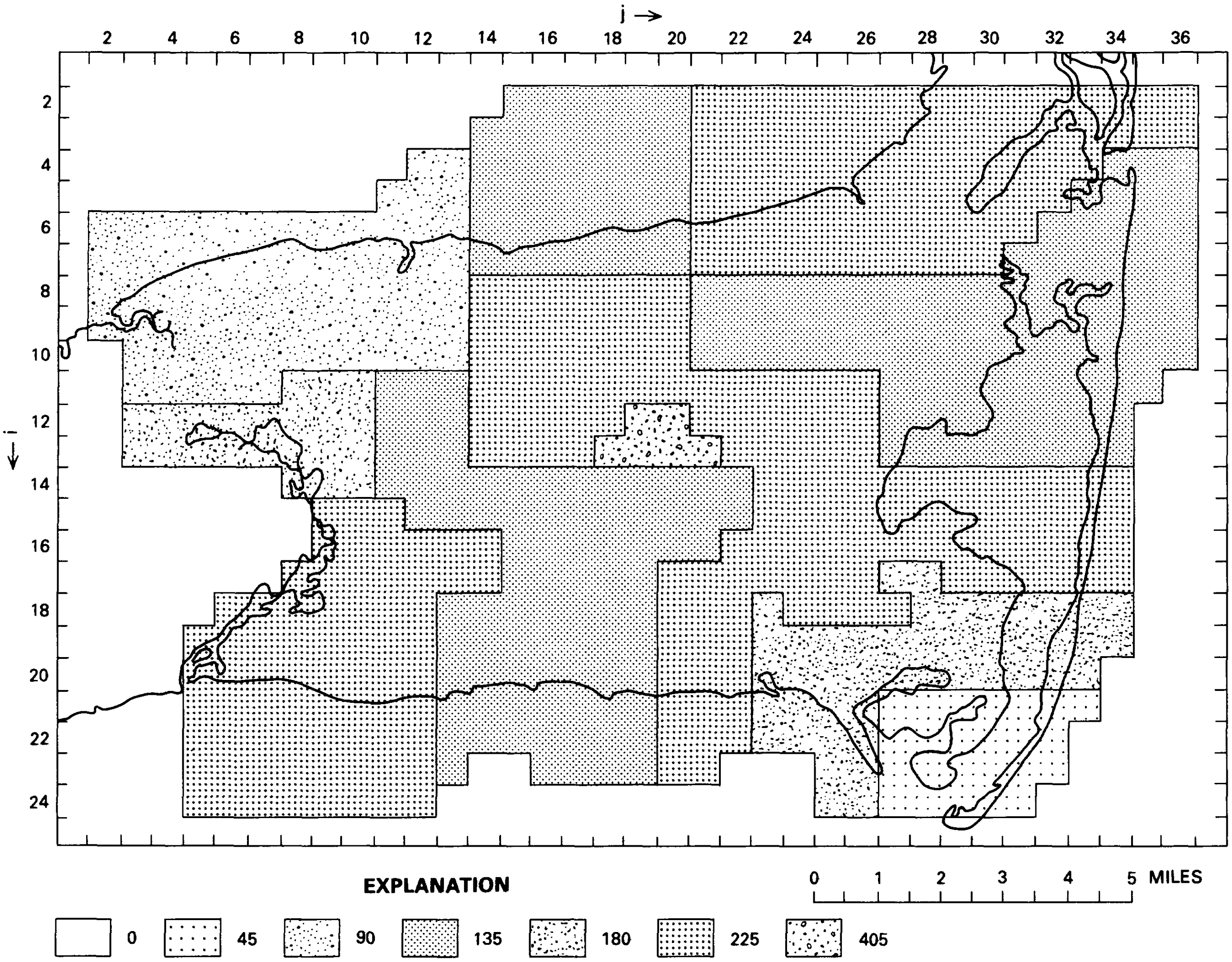


Figure 33. Hydraulic conductivity, in feet per day, ECAPE model, layer 5.

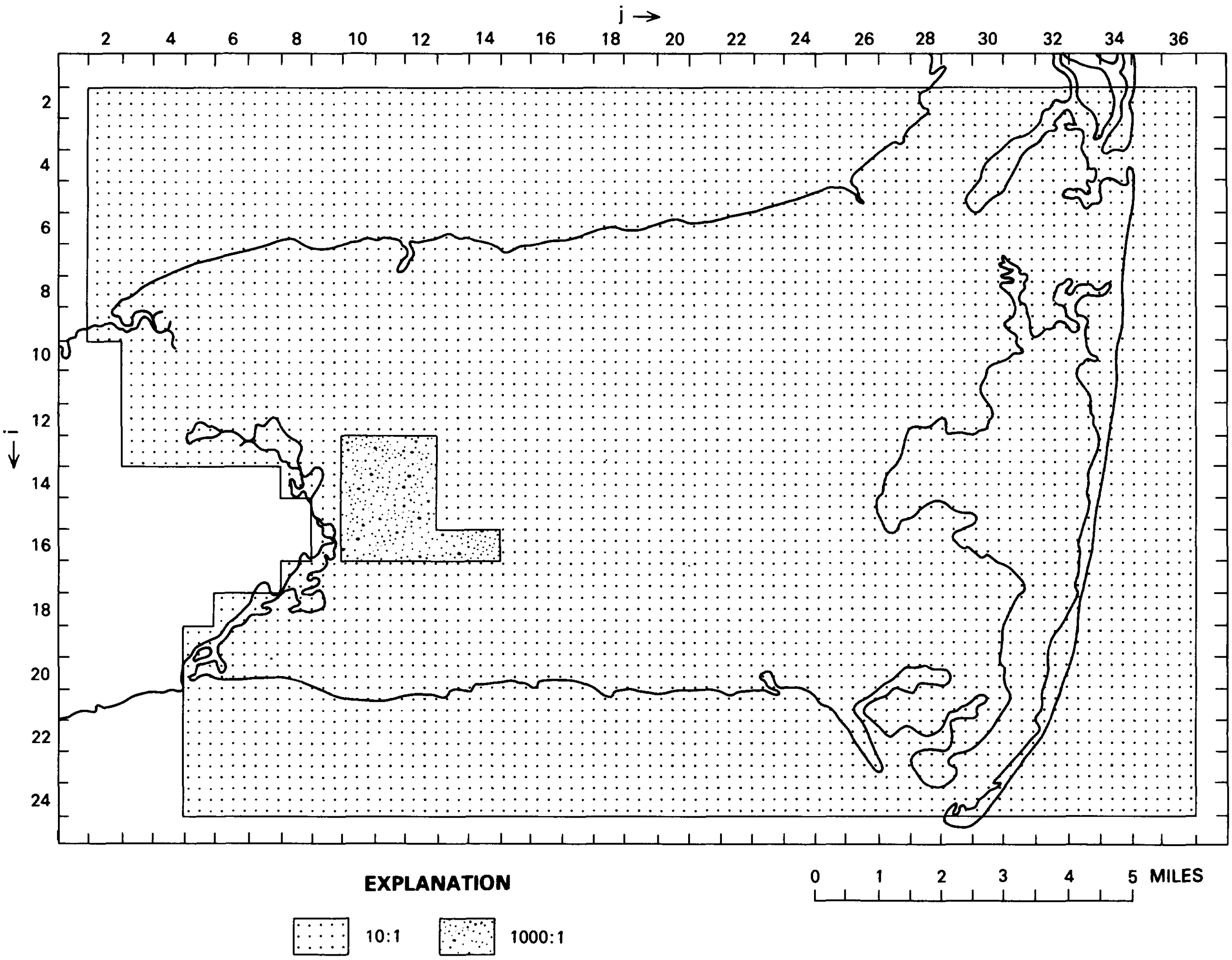


Figure 34. Ratio of lateral to vertical hydraulic conductivity, ECAPE model, layer 2.

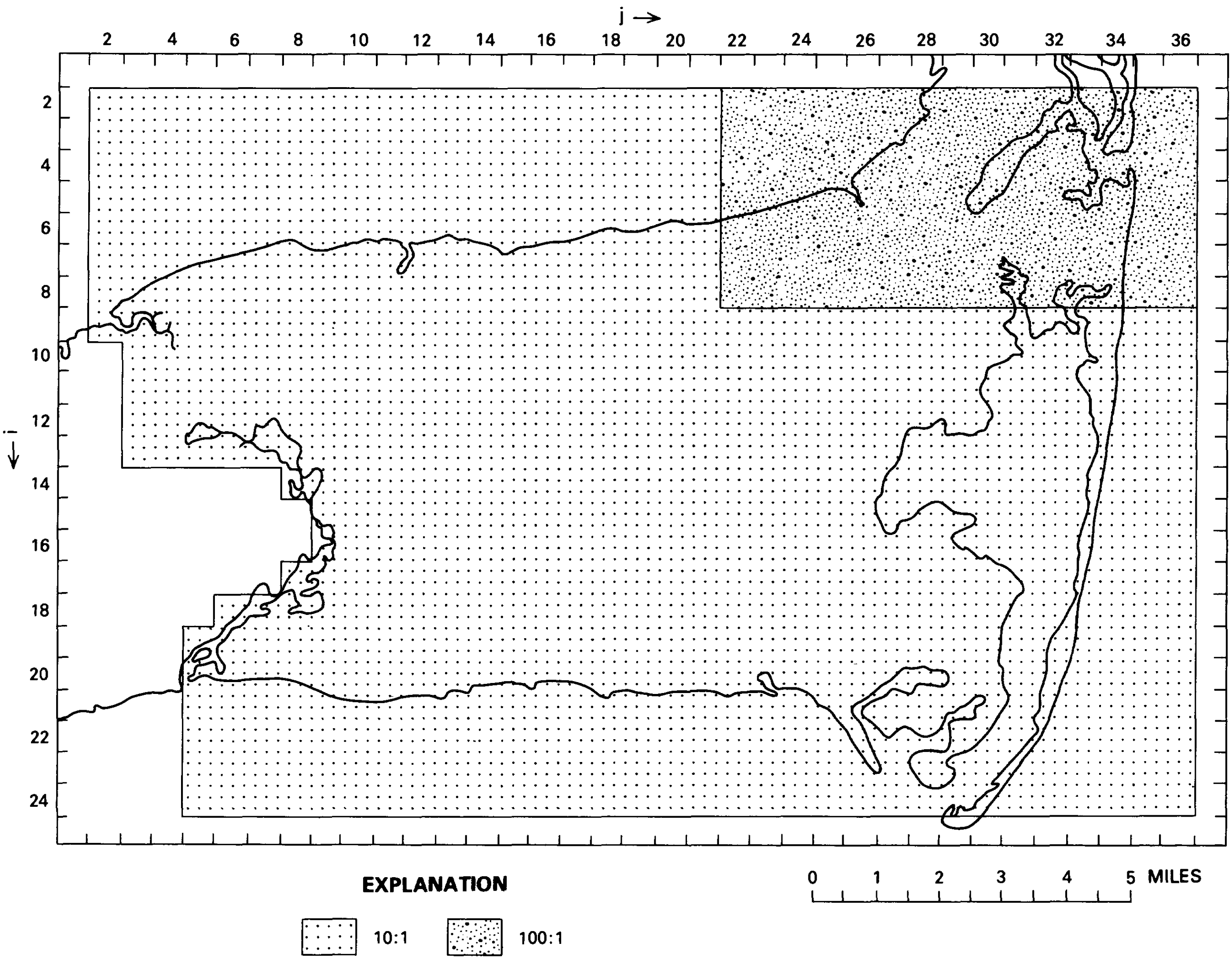


Figure 35. Ratio of lateral to vertical hydraulic conductivity, ECAPE model, layer 3.

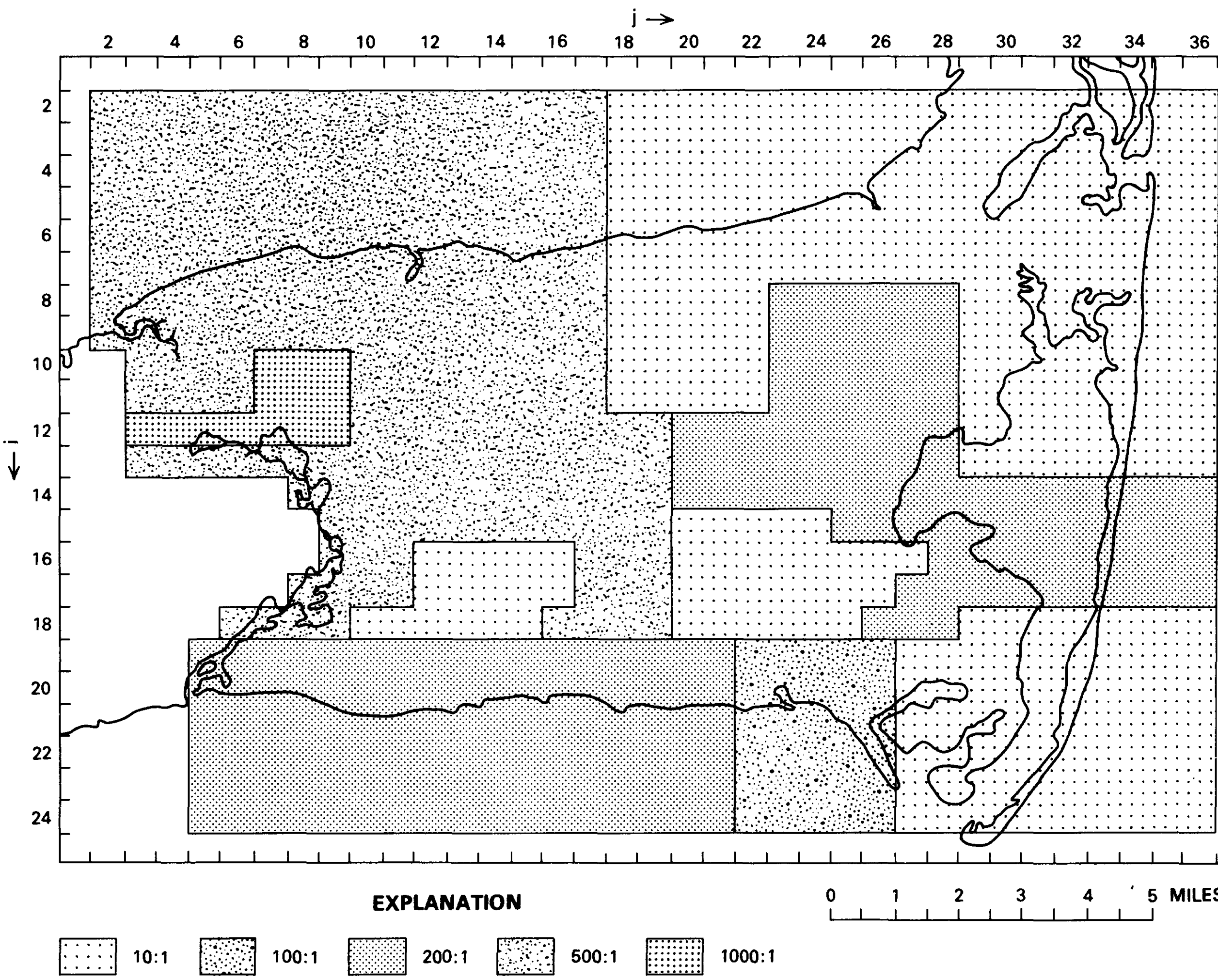


Figure 36. Ratio of lateral to vertical hydraulic conductivity, ECAPE model, layer 4.

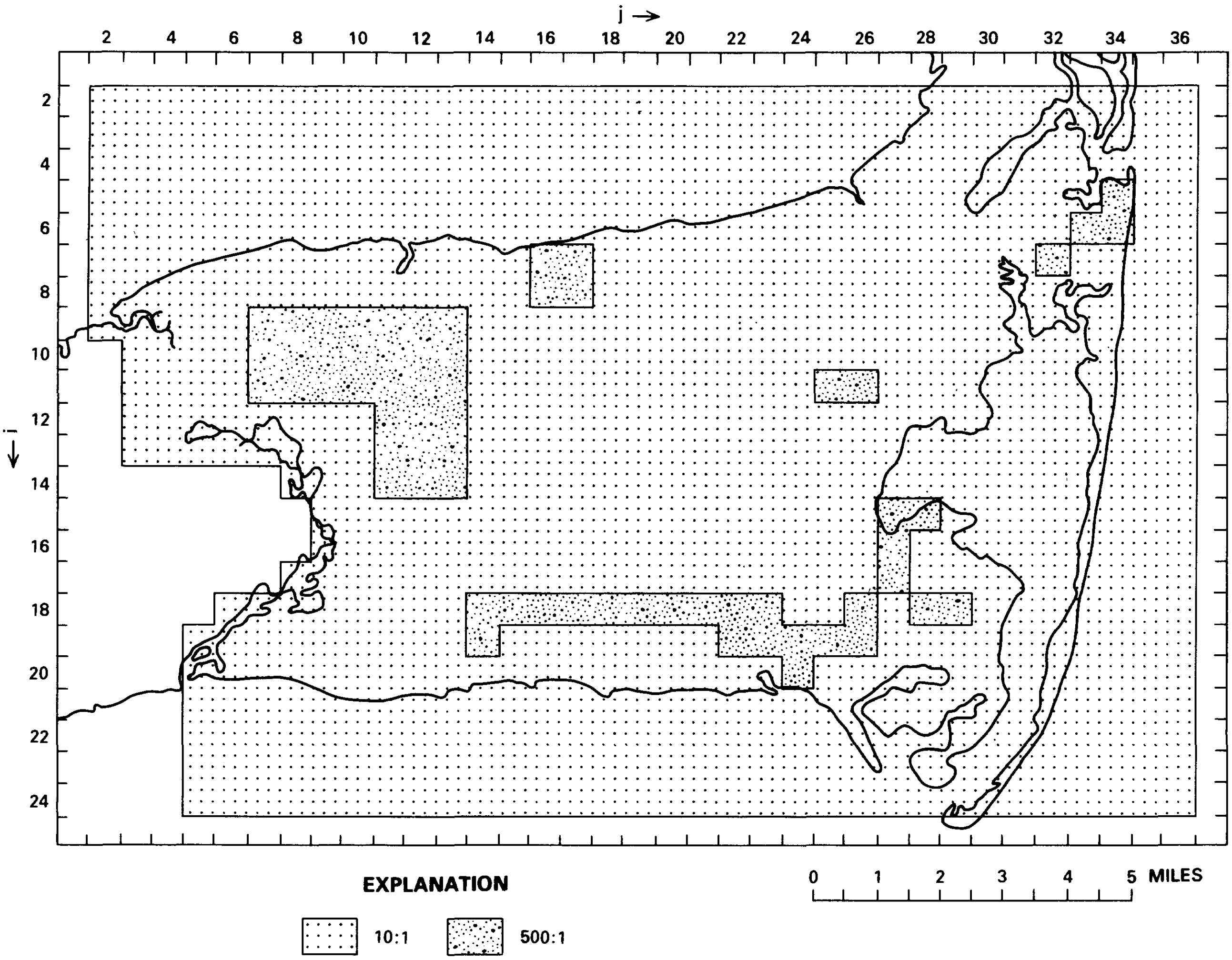


Figure 37. Ratio of lateral to vertical hydraulic conductivity, ECAPE model, layer 5.

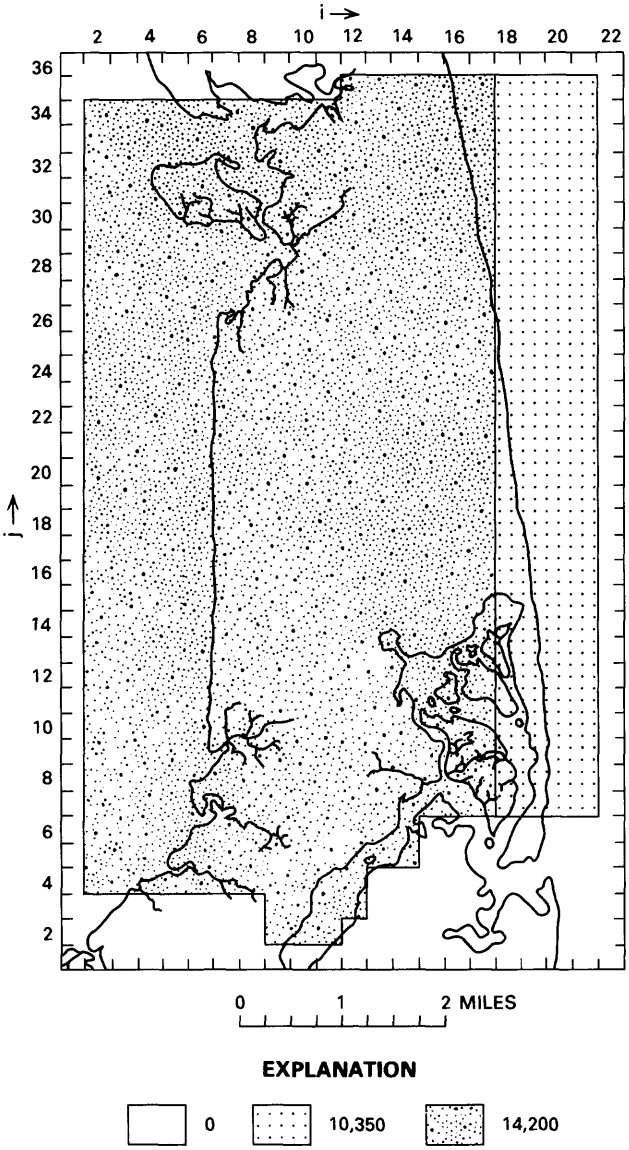


Figure 38. Transmissivity, in feet squared per day, ESTHM model, layer 5.

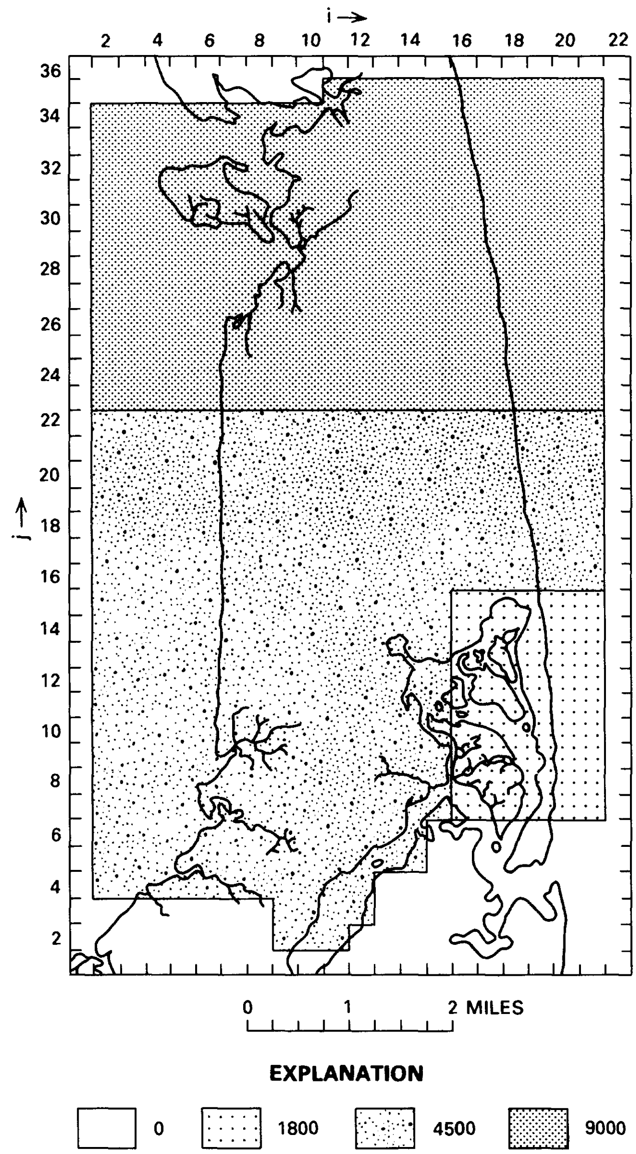


Figure 39. Transmissivity, in feet squared per day, ESTHM model, layer 6.

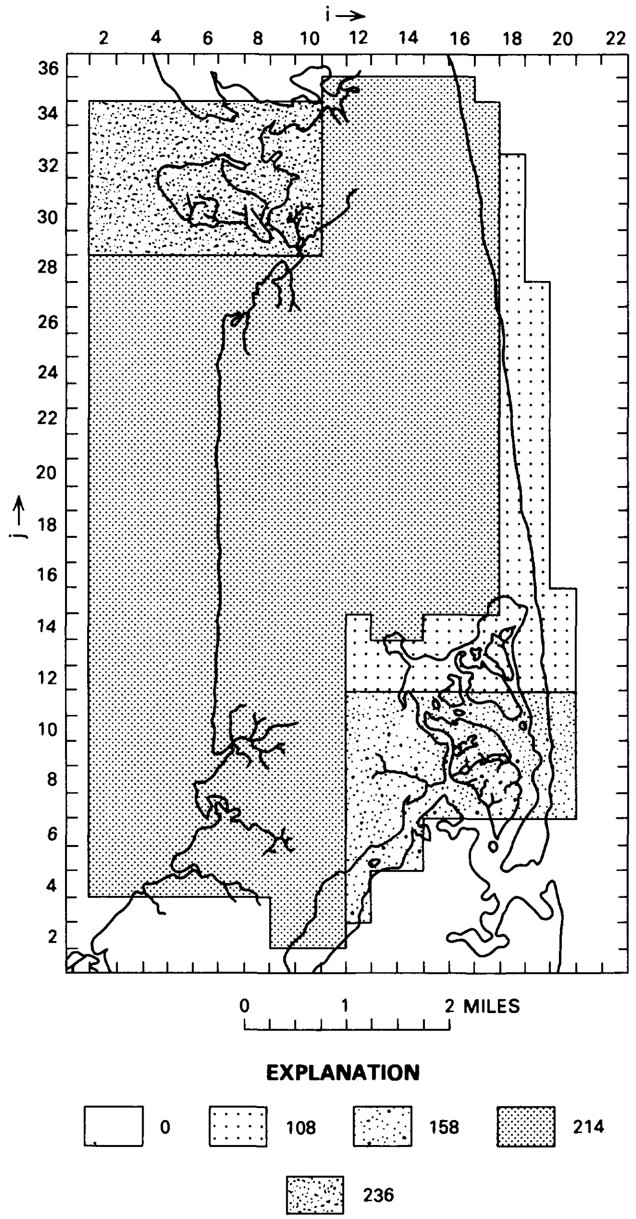


Figure 40. Hydraulic conductivity, in feet per day, ESTHM model, layer 7.

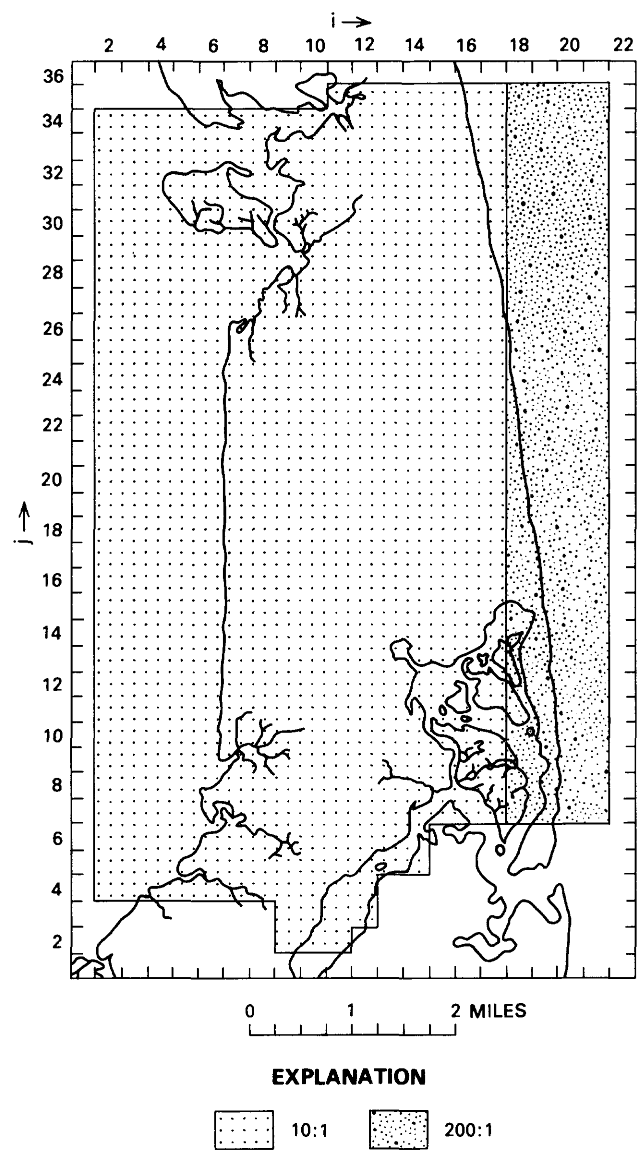
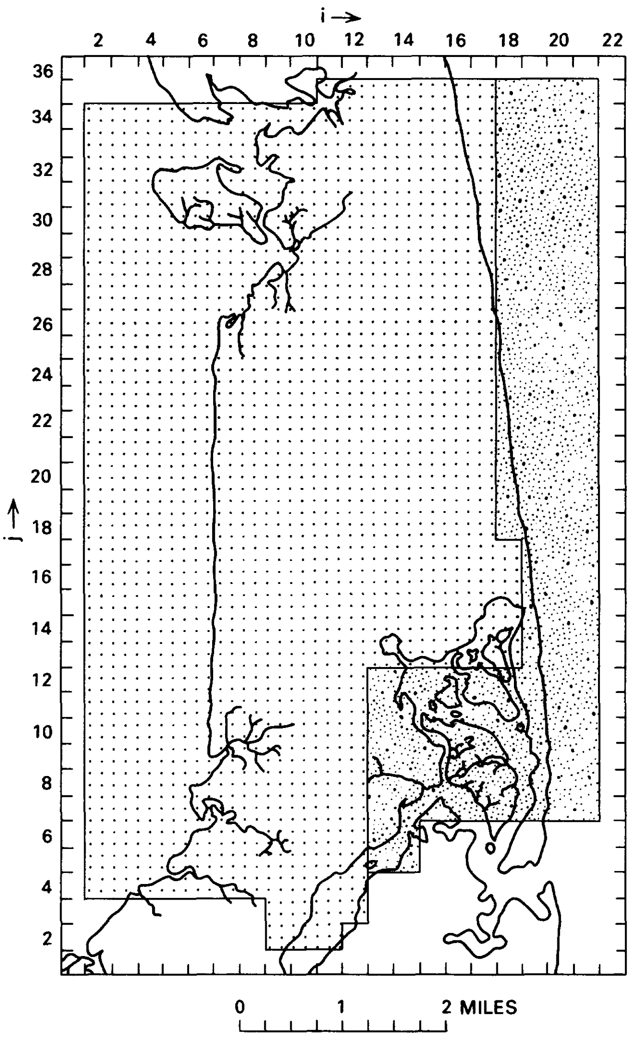


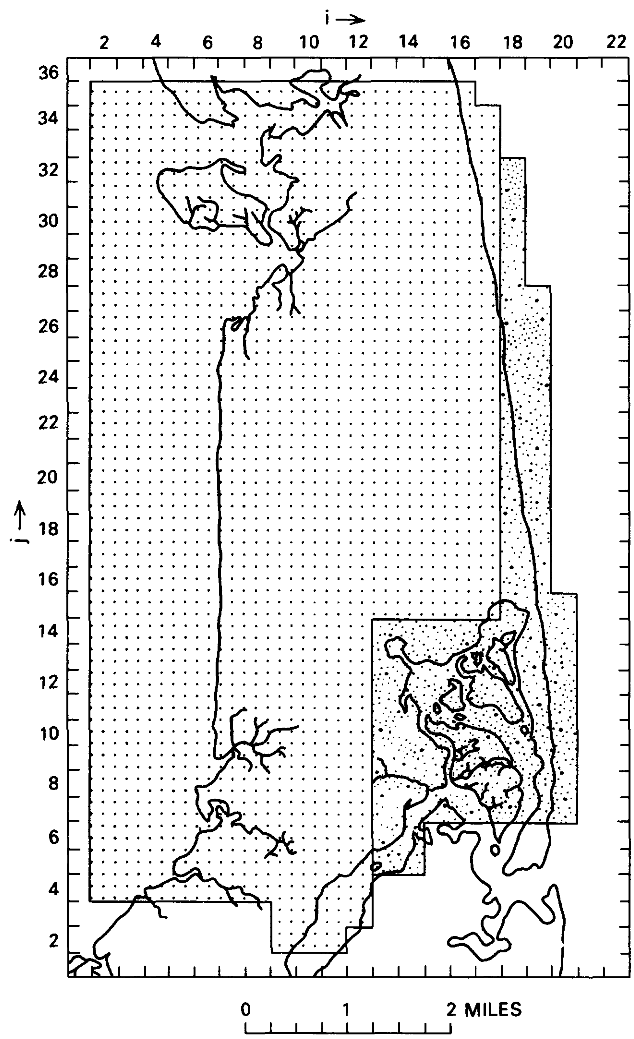
Figure 41. Ratio of lateral to vertical hydraulic conductivity, ESTHM model, layer 5.



EXPLANATION

10:1 20:1

Figure 42. Ratio of lateral to vertical hydraulic conductivity, ESTHM model, layer 6.



EXPLANATION

10:1 20:1

Figure 43. Ratio of lateral to vertical hydraulic conductivity, ESTHM model, layer 7.

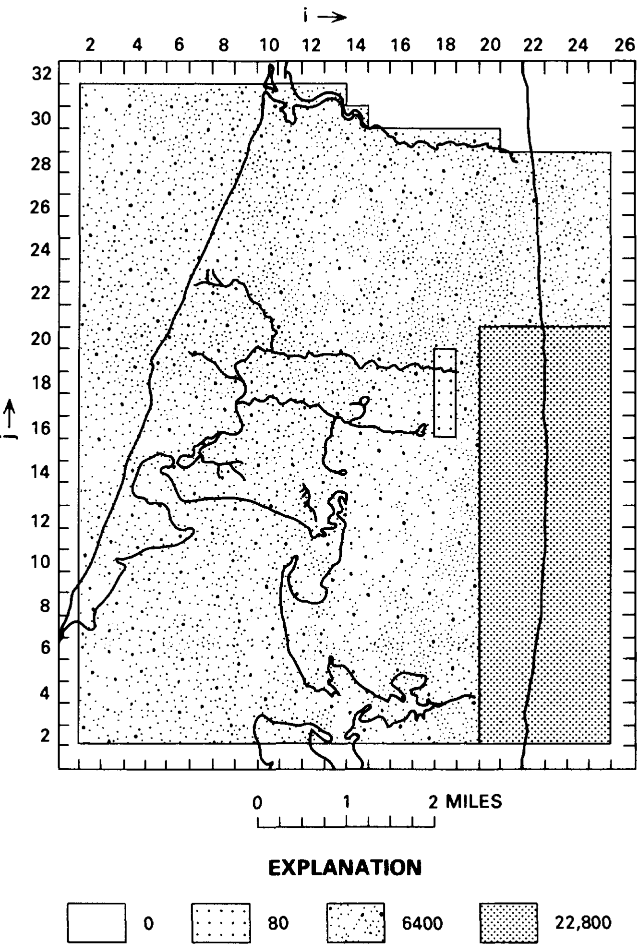


Figure 44. Transmissivity, in feet squared per day, WLFLT model, layer 4.

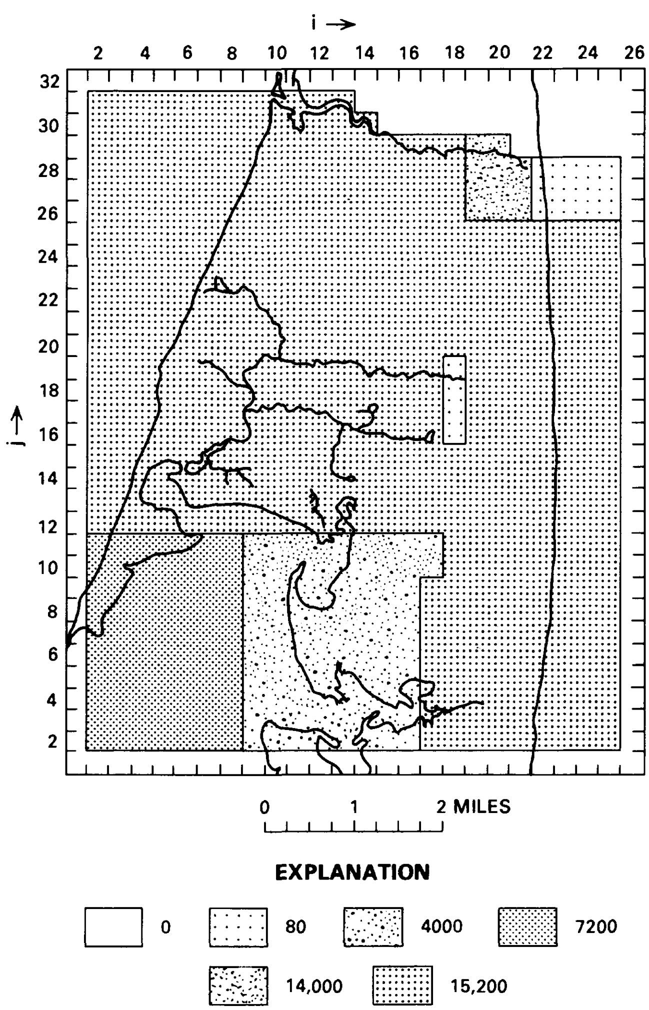


Figure 45. Transmissivity, in feet squared per day, WLFLT model, layer 5.

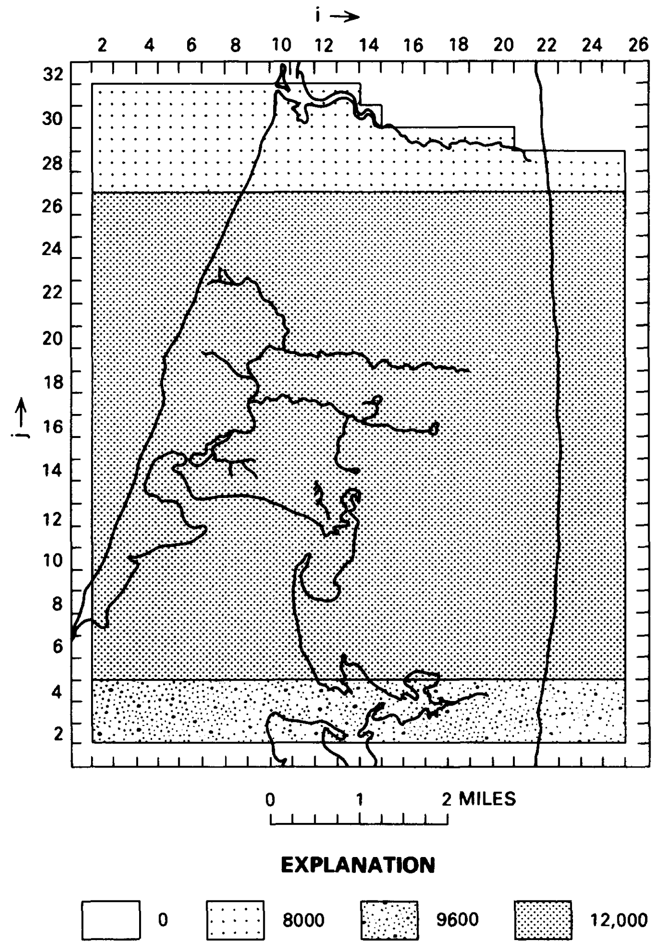


Figure 46. Transmissivity, in feet squared per day, WLFLT model, layer 6.

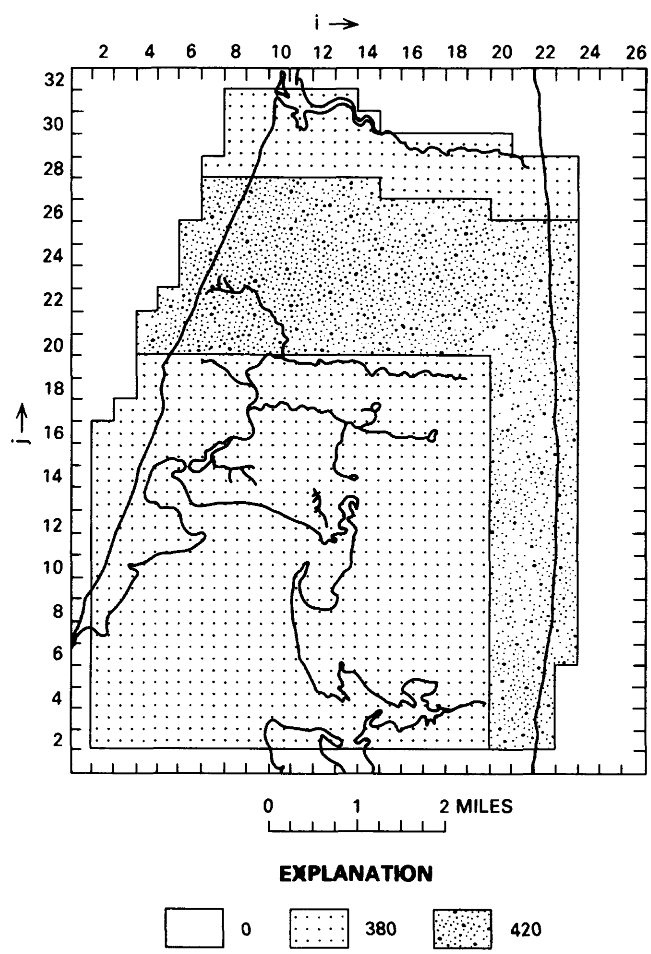


Figure 47. Hydraulic conductivity, in feet per day, WLFLT model, layer 7.

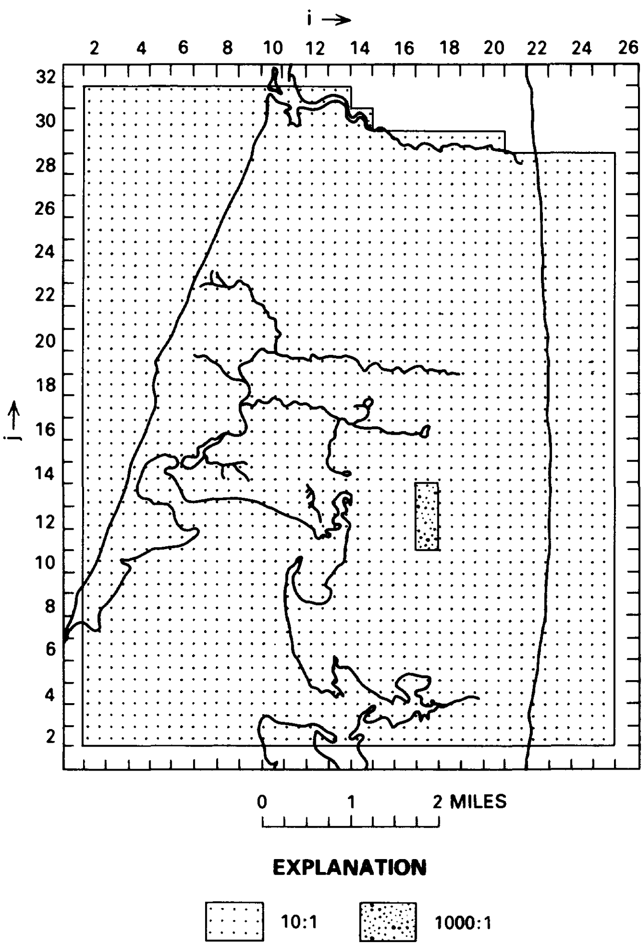


Figure 48. Ratio of lateral to vertical hydraulic conductivity, WLFLT model, layer 5.

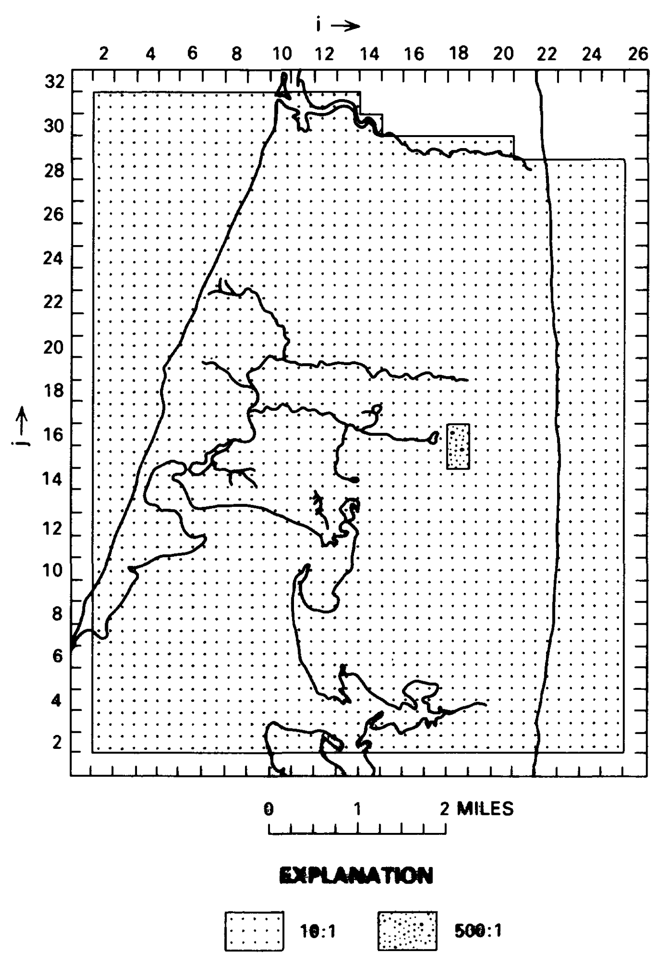


Figure 49. Ratio of lateral to vertical hydraulic conductivity, WLFLT model, layer 6.

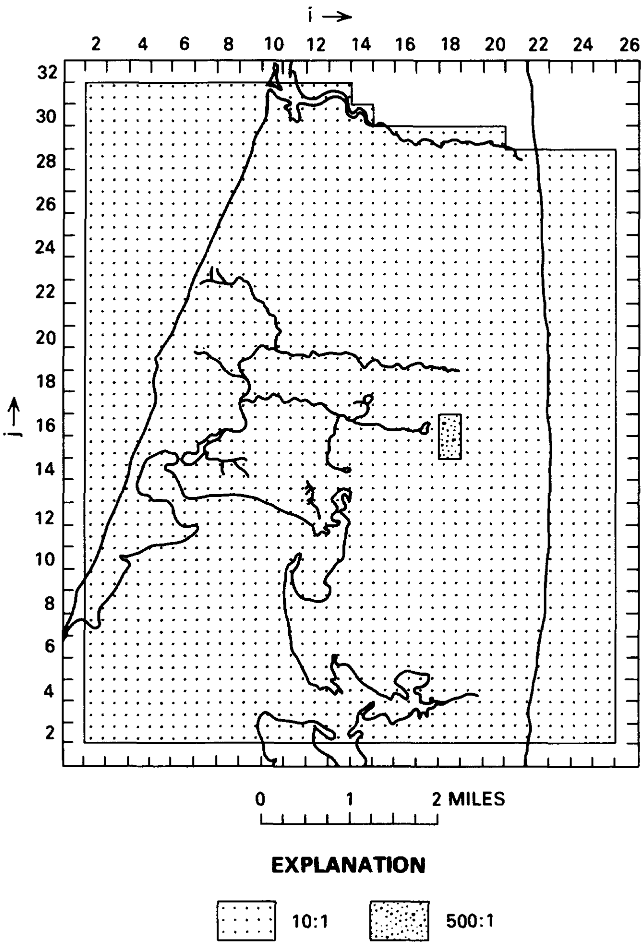


Figure 50. Ratio of lateral to vertical hydraulic conductivity, WLFLT model, layer 7.

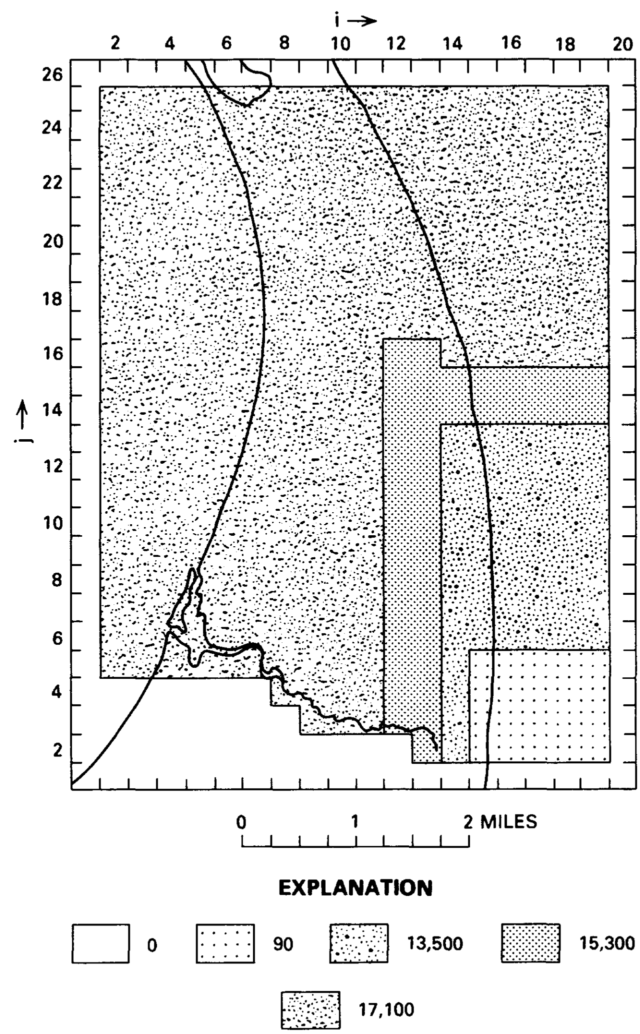


Figure 51. Transmissivity, in feet squared per day, TRURO model, layer 5.

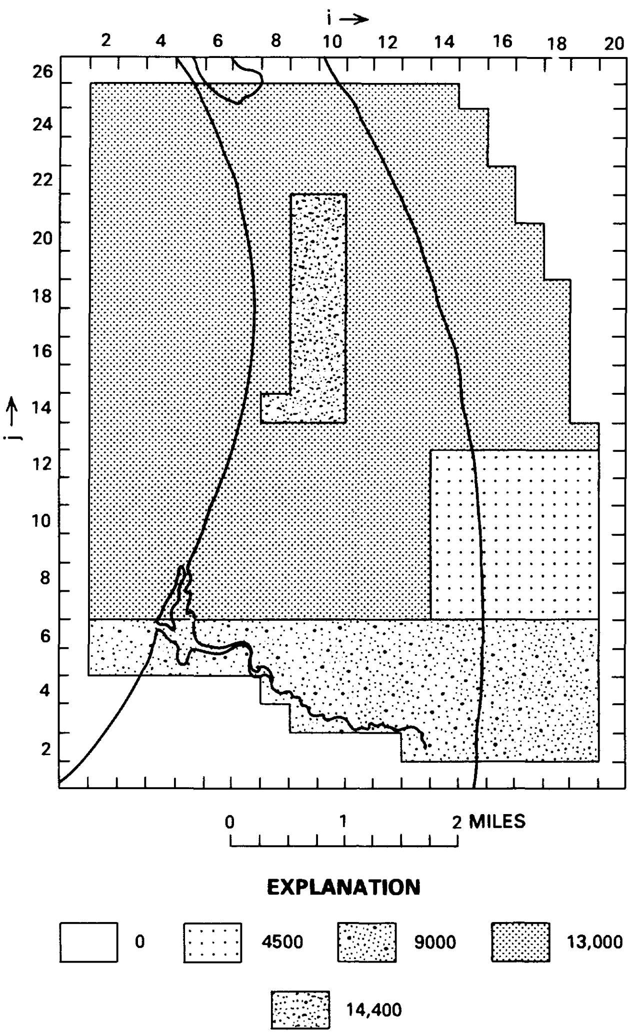


Figure 52. Transmissivity, in feet squared per day, TRURO model, layer 6.

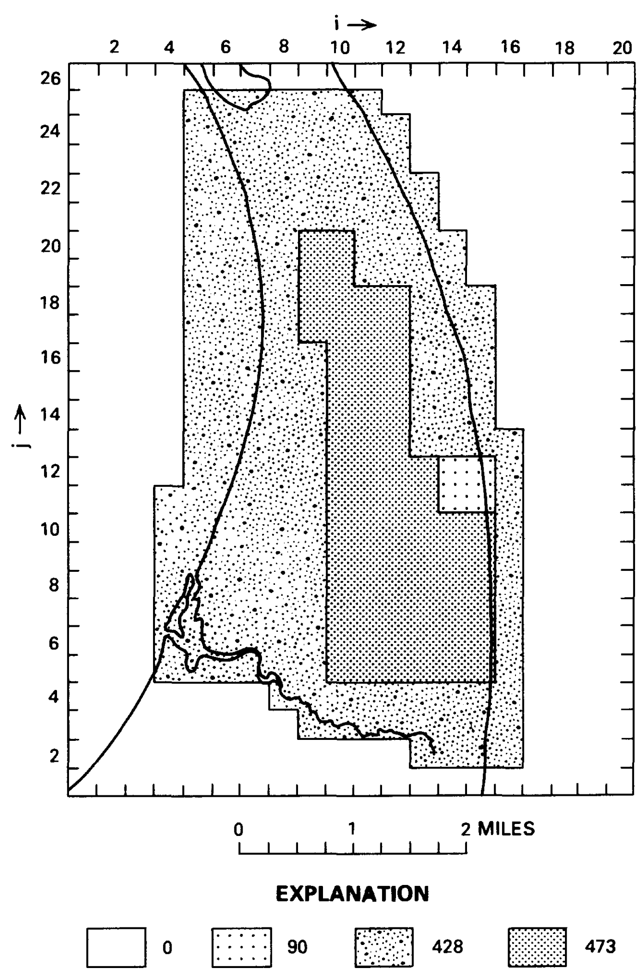


Figure 53. Hydraulic conductivity, in feet per day, TRURO model, layer 7.

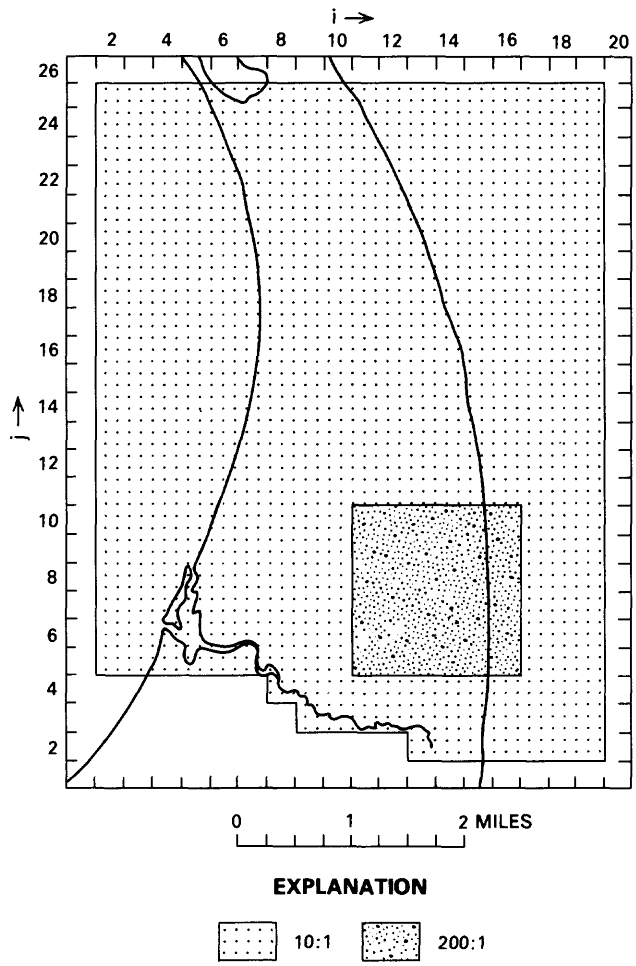


Figure 54. Ratio of lateral to vertical hydraulic conductivity, TRURO model, layer 5.

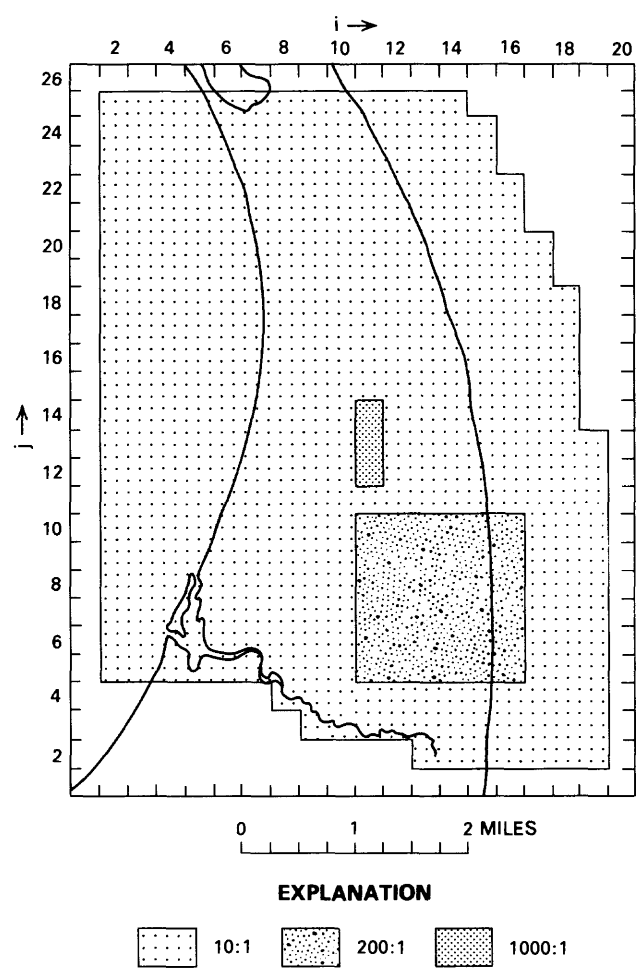


Figure 55. Ratio of lateral to vertical hydraulic conductivity, TRURO model, layer 6.

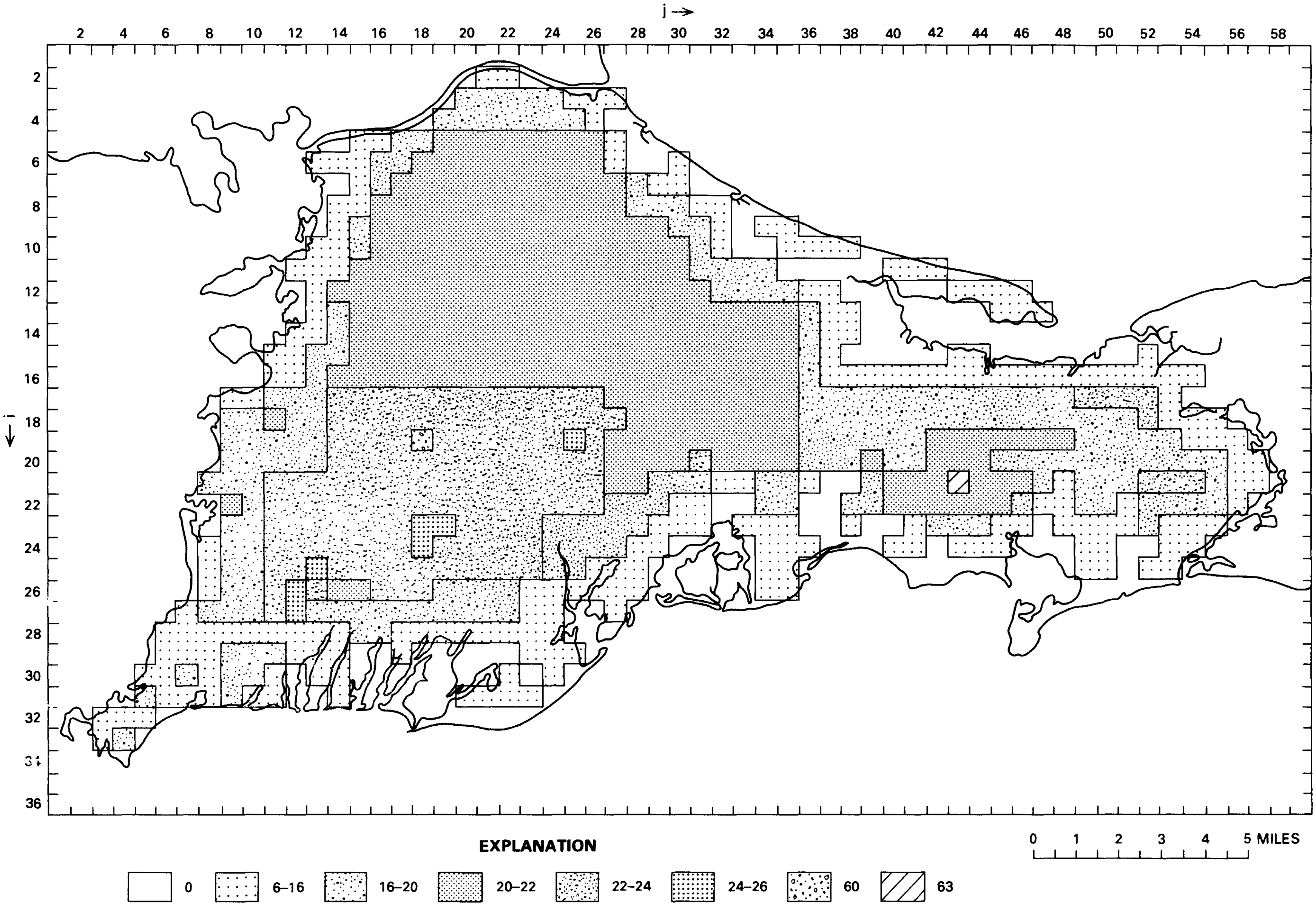


Figure 56. Steady-state recharge, in inches per year, WCAPE model.

Table 10. Summary of well discharges represented in models
(An asterisk indicates multiple well. A dash indicates well not pumped during 1975-76.)

Node (i,j,k)	U.S. Geological Survey well No.	Discharge (ft ^{3}/s)	Total node discharge (ft ^{3}/s)	Node (i,j,k)	U.S. Geological Survey well No.	Discharge (ft ^{3}/s)	Total node discharge (ft ^{3}/s)
			Model:	WCAPE			
3,24,4	BHW 22,232*	0.124	0.124	21,56,4	YAW 53	0.223	
4,26,4	SDW 249,250	.062	.062		YAW 144	.256	
7,28,4	SDW 27	.104	.104		YAW 146	.238	0.717
7,28,5	SDW 37	.320	.320	22,29,5	A1W 369	.113	.113
9,15,4	BHW 233	.079	.079	22,46,4	A1W 377	1.165	
10,14,4	BHW 1-3,136	.336	.336		A1W 385,386	.109	1.274
10,15,5	BHW 199	.239	.239	22,49,4	YAW 128	.125	
15,3,4	BHW 137	.288	.288		YAW 195	--	.125
16,20,5	SDW 155	.433	.433	22,50,4	YAW 193	--	
17,16,5	BHW 23	.507	.507		YAW 194	--	0
17,43,5	A1W 228	.172	.172	22,51,4	YAW 130	--	0
18,52,4	YAW 103*	.223	.223	22,53,4	YAW 64	.345	
18,52,5	YAW 103*	.223	.223		YAW 65	.309	.654
19,35,5	A1W 371	.170	.170	23,17,5	FSW 214	--	0
19,36,5	A1W 372	.156	.156	23,28,4	A1W 251	.112	.112
19,45,5	A1W 370	.269	.269	23,34,5	A1W 159	.133	
19,46,5	A1W 402	.072	.072		A1W 107	.400	.533
19,49,4	YAW 41	--	0	23,34,5	A1W 249	.371	
19,49,5	YAW 42	.248			A1W 158	.133	
	YAW 43	.266	.514		A1W 160	.133	.637
19,55,4	YAW 126	.046		23,41,4	A1W 226	.071	.071
	YAW 127	.027	.073	23,41,5	A1W 227	.048	
20,46,5	A1W 403	.065			A1W 368	.119	.167
	A1W 383	.305		23,42,4	A1W 376	.453	.453
	A1W 387	.173	.543	24,29,5	A1W 224	.046	.046
20,56,4	YAW 54	.467	.467	24,42,4	A1W 229	.210	
21,29,5	A1W 59	.007	.007		A1W 384	.966	1.176
21,36,4	A1W 373	.031	.031	26,9,5	Long Pond	1.04	1.04
21,36,5	A1W 259	.471	.471	26,10,5	Long Pond	1.45	1.45
21,50,4	YAW 58	.310	.310	27,10,5	Long Pond	1.66	1.66
21,52,4	YAW 61	.279		29,24,5	MIW 32	.048	.048
	YAW 63	.208	.487	30,23,4	MIW 35	.150	.150

Table 10. Summary of well discharges represented in models—Continued

Node (i,j,k)	U.S. Geological Survey well No.	Discharge (ft^3/s)	Total node discharge (ft^3/s)	Node (i,j,k)	U.S. Geological Survey well No.	Discharge (ft^3/s)	Total node discharge (ft^3/s)
			Model: ECAPE				
8,28,4	OSW 11	0.322		13,12,5	DGW 77	0.273	
	OSW 14	.206			DGW 87	.150	0.423
	OSW 15	.277	0.805	14,9,5	DGW 1-5	.246	.246
9,28,4	OSW 42	.041		14,10,5	DGW 66	.161	.161
	OSW 43	.088	.129	16,12,4	DGW 205	--	0
11,9,5	DGW 232	.015		18,20,4	HJW 160	.120	.120
	DGW 67	.059	.074	18,20,5	HJW 49	.529	
11,11,5	DGW 244	--	0		HJW 55	.148	
11,24,4	BMW 37	.287	.287		HJW 56	.373	
12,9,5	DGW 112	.390	.390		HJW 1-4*	.023	1.073
12,10,4	DGW 79	.527	.527	18,22,4	HJW 162	.118	
12,11,5	DGW 85	.232	.232		HJW 163	.126	.244
12,12,5	DGW 86	.139	.139	18,22,5	HJW 161	.032	.132
12,24,4	BMW 41	.231	.231	18,26,4	CGW 211	.776	.776
13,11,5	DGW 56	.226		19,22,4	CGW 153	.085	.085
	DGW 57	.013		19,22,5	CGW 1,2	.171	.171
	DGW 58	.123	0.362				
			Model: TRURO				
8,21,6	TSW 115*	0.696	0.696	11,14,6	TSW 78*	0.358	0.358
10,14,6	TSW 78*	.358	.358				

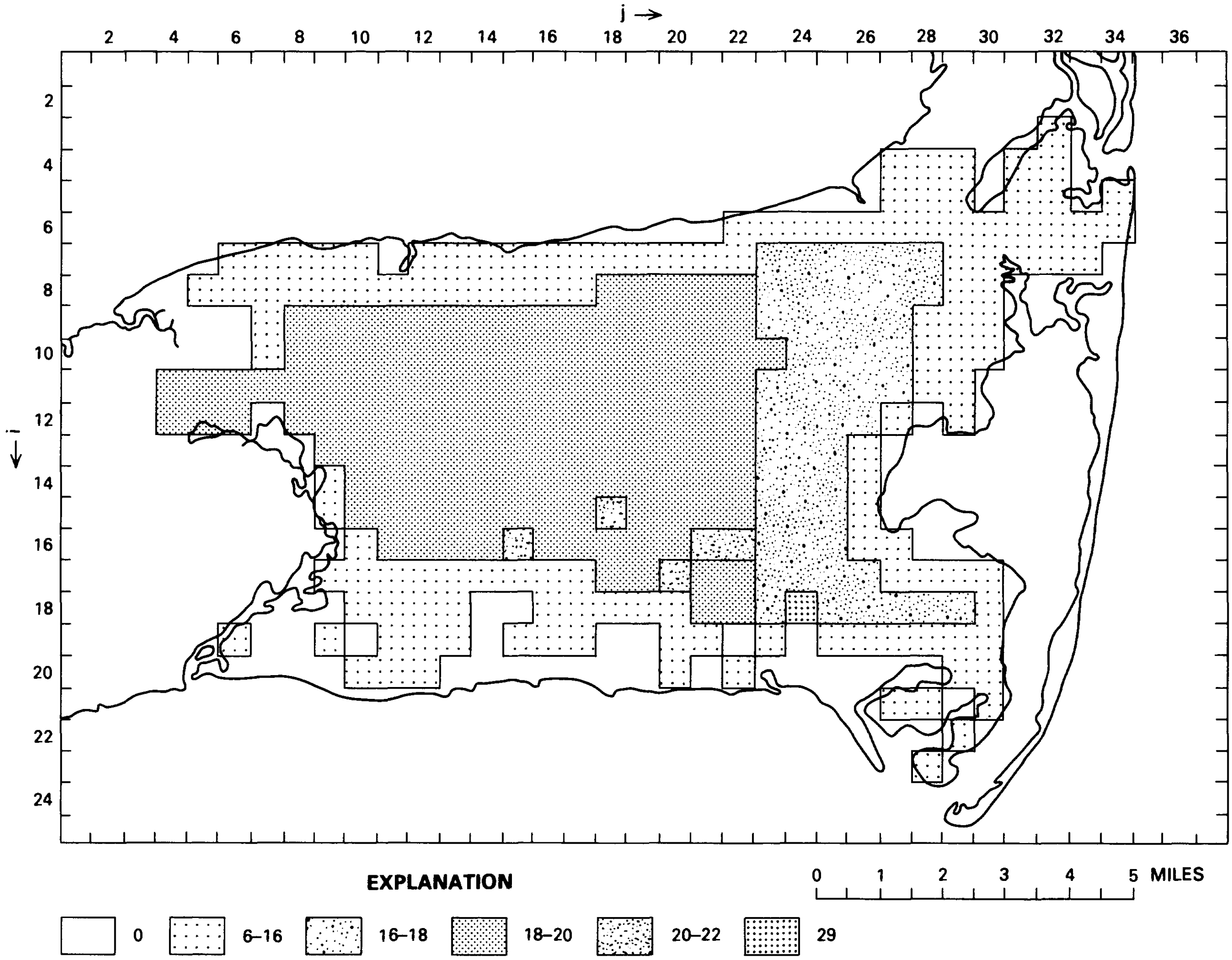


Figure 57. Steady-state recharge, in inches per year, ECAPE model.

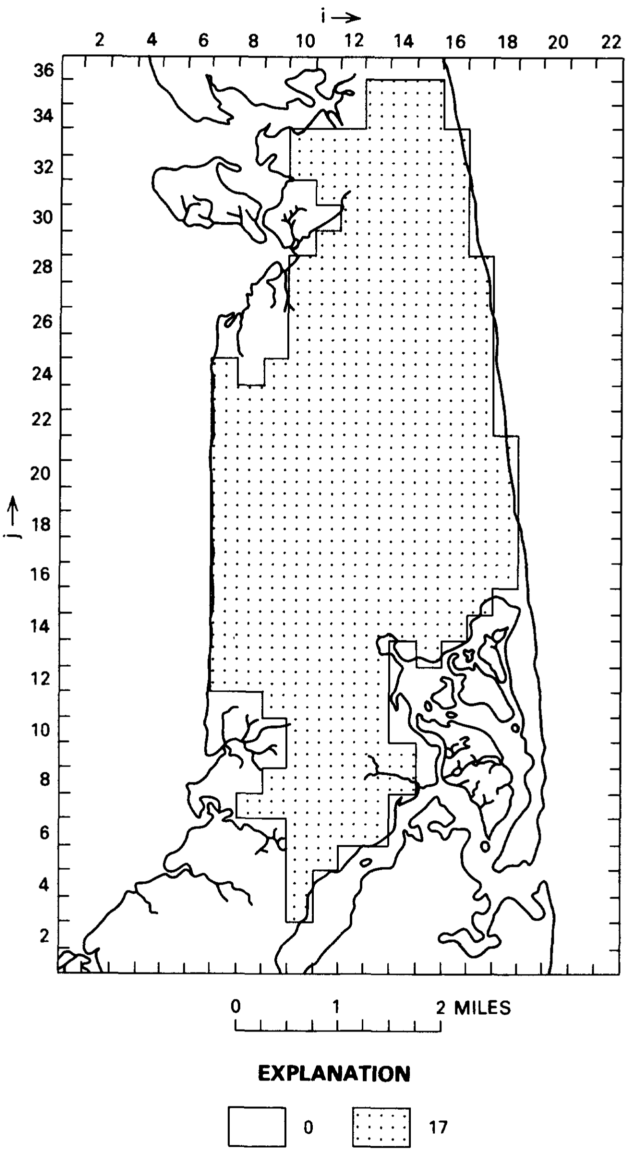


Figure 58. Steady-state recharge, in inches per year, ESTHM model.

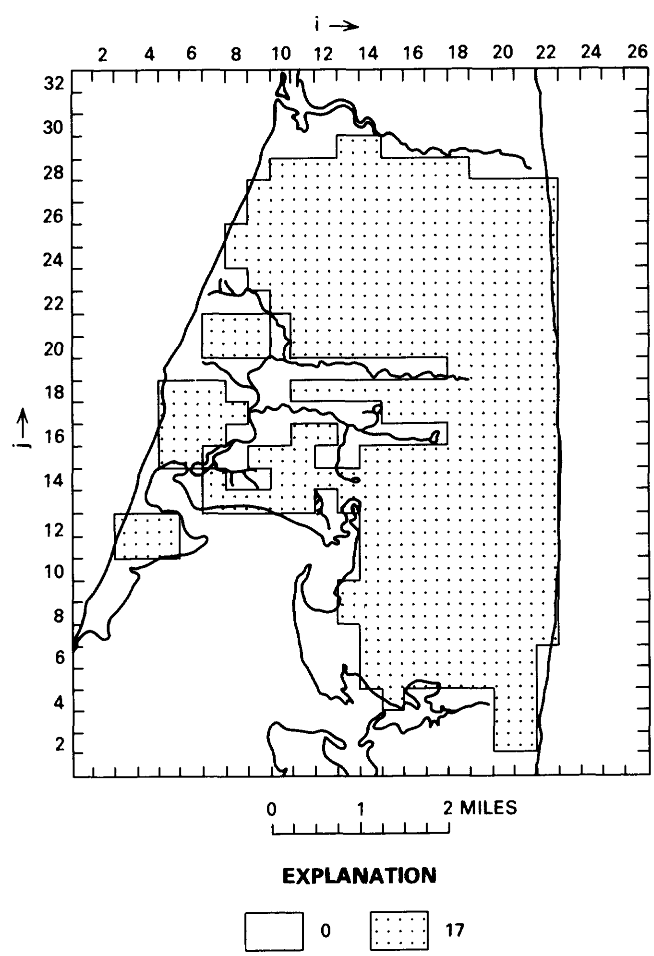


Figure 59. Steady-state recharge, in inches per year, WLFLT model.

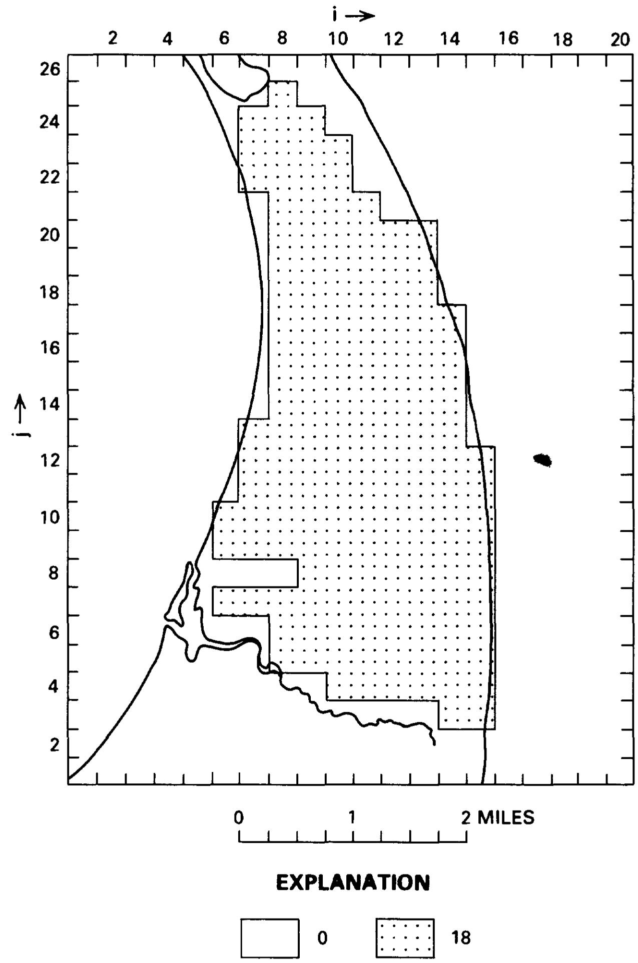


Figure 60. Steady-state recharge, in inches per year, TRURO model.

Table 11. Input documentation

Group I: Title, Simulation Options, and Problem Dimensions

This group of cards, which are read by the main program, contains data required to dimension the model. To specify an option on card 4 punch the characters underlined in the definition. For an option not used, that section of the card 4 can be left blank.

NOTE: Default typing of variables applies for all data input.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
1	1-80	20A4	HEADING	Any title the user wishes to print on one line at the start of output
2	1-52	13A4	HEADING	
3	1-10	I10	IO	Number of rows
	11-20	I10	JO	Number of columns
	21-30	I10	KO	Number of layers
	31-40	I10	ITMAX	Maximum number of iterations per time step
	41-50	I10	NCH	Number of constant head nodes
	51-60	I10	NKODE	= 1 for interface solution; ≠ 1 if freshwater/saline water boundary is not to be considered
	61-70	G10.0	BETA1	Parameter to dampen numerical oscillation (usually 0.6 to 1.0)
	71-75	I5	KFLOW	= 1 to print flow rates to individual constant head and leaky nodes
	76-80	I5	NRIV	Number of leaky "river" nodes
4	1-4	A4	IDRAW	DRAW to print drawdown
	6-9	A4	IHEAD	HEAD to print hydraulic head
	11-14	A4	IFLOW	MASS to compute a mass balance
	16-18	A3	IDK1	DK1 to read initial head, elapsed time, and mass balance parameters from unit 4 on disk
	21-23	A3	IDK2	DK2 to write computed head, elapsed time and mass balance parameters on unit 4 (disk)
	26-29	A4	IWATER	WATE if the upper hydrologic unit is unconfined
	31-34	A4	IQRE	RECH for a constant recharge that may be a function of space
	36-39	A4	IPU1	PUN1 to read initial head, elapsed time, and mass balance parameters from cards
	41-44	A4	IPU2	PUN2 to punch computed head, elapsed time, and mass balance parameters on cards
	46-49	A4	ITK	ITKR to read the values of TK (I,J,K); for simulations in which confining layers are not represented by layers of nodes, TK (I,J,K) = K_{zz}/b; where variation in aquifer anistrophy is to be approximated, TK (I,J,K) = 2.0*K_{zz}(K+1)*K_{zz} (K)/(K_{zz}(K+1)*DELZ(K)+K_{zz}(K)*DELZ(K+1))(Trescott and Larson, 1976)
	51-54	A4	IEQN	EQN3 if equation 3 is being solved; otherwise it is assumed that equation 4 is being solved

Group II: Scalar Parameters

The parameters required in every problem are underlined. The other parameters are required as noted; when not required, their location on the card can be left blank. The G format is used to read E, F, and I format data. Minimize mistakes by always right-justifying data in the field. If F format data do not contain significant figures to the right of the decimal point, the decimal point can be omitted.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
1	1-10	G10.0	NPER	Number of pumping periods for the simulation
	11-20	G10.0	KTH	Number of times steps between printouts
	NOTE: To print only the results for the final time step in a pumping period, make KTH greater than the expected number of time steps. The program always prints the results for the final time step.			
	21-30	G10.0	ERR	Error criterion for closure (L)
	NOTE: When the head change at all nodes on subsequent iterations is less than this value (for example, 0.01 foot), the program has converged to a solution for the time step.			
	31-40	G10.0	LENGTH	Number of iteration parameters
2	1-10	G10.0	XSCALE	Factor to convert model length unit to unit used in X direction on maps (e.g. to convert from feet to miles, XSCALE = 5280). For no maps, card 2 is blank)
	11-20	G10.0	YSCALE	Factor to convert model length unit used in Y direction on maps
	21-30	G10.0	DINCH	Number of map units per inch
	31-40	G10.0	FACT1	Factor to adjust value of drawdown printed*
	41-49	9I1	LEVEL1(I)	Layers for which drawdown maps are to be printed. List layers starting in column 41; the first zero entry terminates the printing of drawdown maps
	51-60	G10.D	FACT2	Factor to adjust value of head printed*
	61-69	9I1	LEVEL2(I)	Layers for which head maps are to be printed. List layers starting in column 61; the first zero entry terminates the printing of head maps.
	71-78	A8	MESUR	Name of map length unit

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
3	1-20	G20.10	SUM	Parameters in which elapsed time and cumulative volumes for mass balance are stored. For the start of a simulation, insert three blank cards. For continuation of a previous run using cards as input, replace the three blank cards with the first three cards of punched output from the previous run. Using data from disk for input, leave the three blank cards in the data deck.
	21-40	G20.10	SUMP	
	41-60	G20.10	PUMPT	
	61-80	G20.10	CFLUXT	
4	1-20	G20.10	QRET	
	21-40	G20.10	CHST	
	41-60	G20.10	CHDT	
	61-80	G20.10	FLUXT	
5	1-20	G20.10	STORT	
	21-40	G20.10	ETFLXT	
	41-60	G20.10	FLXNT	

* Value of

drawdown or head	FACT 1 or FACT 2	Printed value
	0.01	1
	0.1	5
52.57	1.0	53
	10.0	526
	100.0	***

Group III: Array Data

Each of the following data sets (except data set 1) consists of a parameter card, and if the data set contains variable data, a set of data cards. If the data set requires data for each layer, a parameter card and data cards (for layers with variable data) are required for each layer. Each parameter card contains at least five variables.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
Every Parameter Card	1-10	G10.0	FAC	If IVAR = 0, FAC is the value assigned to every element of the matrix for this layer If IVAR = 1, FAC is the multiplication factor for the following sets of data cards for this layer
	11-20	G10.0	IVAR	= 0 if no data cards are to be read for this layer = 1 if data cards for this layer follow
	21-30	G10.0	IPRN	= 0 if input data for this layer are to be printed = 1 if input data for the layer are not to be printed

Transmissivity Parameter Cards also have these Variables:

31-40	G10.0	FACT(K,1)	Multiplication factor for transmissivity in x direction
41-50	G10.0	FACT(K,2)	Multiplication factor for transmissivity in y direction
51-60	G10.0	FACT(K,3)	Multiplication factor for hydraulic conductivity in the z direction (Not used when confining bed nodes are eliminated and TK values are read)

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
Every Parameter Card	61-70	G10.0	IRECS	= 0 if the matrix is being read from cards or if each element is being set equal to FAC =1 if the matrix is to be read from disk (unit 2)
Every Parameter Card	71-80	G10.0	IRECD	= 0 if the matrix is not to be stored on disk = 1 if the matrix being read from cards or set equal to FAC is to be stored on disk (unit 2) for later retrieval

When data cards are included, start each row on a new card. To prepare a set of data cards for an array that is a function of space, the general procedure is to overlay the finite-difference grid on a contoured map of the parameter and record the average value of the parameter for each finite-difference block on coding forms according to the appropriate format. In general, record only significant digits and no decimal points (except for data set 2); use the multiplication factor to convert the data to their appropriate values. For example, if DELX ranges from 1000 to 15000 feet, coded values should range from 1-15; the multiplication factor (FAC) would be 1000.

DATA SET	COLUMNS	FORMAT	VARIABLE	DEFINITION
1	1-80	8F 10.4	PHI(I,J,K)	Head values for continuation of a previous run (L)

NOTE: For a new simulation this data set is omitted. Do not include a parameter card with this data set.

2	1-80	8F 10.4	STRT(I,J,K)	Starting head matrix (L)
3	1-80	20F 4.0	S (I,J,K)	Storage coefficient (dimensionless). If equation 3 is to be solved, read Ss instead of storage coefficient

NOTE: This matrix is also used to locate constant head boundaries by coding a negative number at constant head nodes. At these nodes, T must be greater than zero.

DATA SET	COLUMNS	FORMAT	VARIABLE	DEFINITION
4	1-80	8F 10.4	T(I,J,K)	Transmissivity (L^2/T). If equation 3 is to be solved, read hydraulic conductivity instead of transmissivity

NOTE: 1) Zero values are required around the perimeter of the T matrix for each layer for reasons inherent in the computational scheme. This is done automatically by the program.

2) See the previous page for additional requirements on the parameter cards for this data set.

3) If the upper active layer is unconfined and PERM and BOTTOM are to be read for this layer, insert a parameter card for this layer with only the values for FACT on it.

5	1-80	8F 10.4	TK(I,J,K)	K_{zz}/b

NOTE: This data set is read only if specified in the options. The number of layers of TK values = KO-1. See the discussion of the treatment of confining layers.

6	1-80	8F 10.4	PERM(I,J)	Hydraulic conductivity (L/T) (see note 1 for data set 4)
7	1-80	20F 4.0	BOTTOM(I,J)	Elevation of bottom of water-table unit (L)

NOTE: Data sets 6 and 7 are required only for simulating unconfined conditions in the upper hydrologic unit.

8	1-80	20F 4.0	QRE(I,J)	Recharge rate (L/T)

NOTE: Omit if not used

DATA SET	COLUMNS	FORMAT	VARIABLE	DEFINITION
9	1-80	8F10.0	DELX(J)	Grid spacing in x direction (L)
10	1-80	8G10.0	DELY(I)	Grid spacing in y direction (L)
11	1-80	8G10.0	DELZ(K)	Grid spacing in z direction (L)
12	1-80	80I1	IDR(I,J,K)	Flag for leaky "river" nodes. = 1 for leaky nodes
13	1-10	G10.0	FAC	Multiplication factor for head on leaky layer
	1-80	20F 4.0	RH(I)	Head on leaky layer
14	1-10	G10.0	FAC	Multiplication factor for elevation of bottom of leaky layer
	1-80	20F 4.0	RB(I)	Elevation of bottom of leaky layer
15	1-10	G10.0	FAC	Multiplication factor for leakance coefficient
	1-80	10F 8.0	RC(I)	Leakance coefficient (K_z/M)

Group IV: Parameters that Change with the Pumping Period

The program has two options for the simulation period:

1. To simulate a given number of time steps, set TMAX to a value larger than the expected simulation period. The program will use NUMT, CDLT, and DELT as coded.
2. To simulate a given pumping period, set NUMT larger than the number required for the simulation period (for example, 50). The program will compute the exact DELT (which will be $\leq$ DELT coded) and NUMT to arrive exactly at TMAX on the last step.

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
1	1-10	G10.0	KP	Number of the pumping period
	11-20	G10.0	KPM1	Number of the previous pumping period
NOTE: KPM1 is currently not used				
	21-30	G10.0	NWEL	Number of wells for this pumping period
	31-40	G10.0	TMAX	Number of days in this pumping period
	41-50	G10.0	NUMT	Number of time steps
	51-60	G10.0	CDLT	Multiplying factor for DELT
NOTE: 1.5 is commonly used				
	61-70	G10.0	DELT	Initial time step in hours

If NWEL = 0 the following set of cards is omitted

DATA SET 1 (NWEL cards)

COLUMNS	FORMAT	VARIABLE	DEFINITION
1-10	G10.0	K	Layer in which well is located
11-20	G10.0	I	Row location of well
21-30	G10.0	J	Column location of well
31-40	G10.0	WELL (I,J,K)	Pumping rate (L^3/T), negative for a pumping well

For each additional pumping period, another set of group IV cards is required (that is, NPER sets of group IV cards are required).

Group V: Interface and Optional Print Parameters

(These data cards follow the first set of Group IV cards)

CARD	COLUMNS	FORMAT	VARIABLE	DEFINITION
1	1-10	I10	NTAD	Transmissivity adjustment code. Specifies frequency of transmissivity adjustment (e.g. = 1 every iteration; = 4 every fourth iteration)
	11-20	G10.0	RF	Density factor
	21-30	I10	KODCHP	= 1 Transmissivity adjustment proceeds across grid from north to south; otherwise adjustment proceeds from both north and south to mid-point of grid
	31-40	I1D	KPRNTT	= 1 write to file 9 final PHI, T, PERM and TK matrices
DATA SET				
1	1-80	16F 5.0	DBOT(I,J)	Elevation of bottom of aquifer (positive downward). Read similar to array data.
2	1-80	80I1	ISEA(I,J)	Flag for "ocean" nodes. Assign value equal to node layer. Ocean nodes which "leak" into aquifer are treated as saline-water nodes
3	1-80	8G10.0	DTOP(K)	Elevation of top of layers (positive downward)

Group VI: Additional Print Parameters

CARD				
1	1-10	I10	NODKOD	= 1 to print head values at each iteration for selected nodes
	11-20	I10	NMBRND	Number of nodes (10 maximum) for which head values are to be printed
2	1-80	10(I3, I3, I2)	AA(I)	I, J, K location of nodes for which head values are to be printed. If NODKOD not equal to 1 omit this card
3	1-10	I10	IHEDKD	= 1 write to file 8 final head values for selected nodes

DATA SET	COLUMNS	FORMAT	VARIABLE	DEFINITION
1	1-80	80I1	IFNLHD (I,J,K)	Index array to identify those nodes for which final head values are to be printed in table form. Two passes are made through the array. The first pass is made only on the top layer and prints head values for those nodes where IFNLHD = 1. The second pass is made through the entire array and prints head values for those nodes where IFNLHD = 2. If IHEDKD is not equal to 1, omit this data set

Table 12. Computer source code

(*Indicates a change from Trescott, 1975, and Trescott and Larson, 1976.)

```
C     ------------------------------------------------------------------MAN0010
C     FINITE-DIFFERENCE MODEL FOR SIMULATION OF GROUND-WATER FLOW IN    MAN0020
C     THREE DIMENSIONS, SEPTEMBER 1975 BY P.C. TRESCOTT, U. S. G. S.    MAN0030
C     WITH CONTRIBUTIONS TO MAIN, DATAI AND SOLVE BY S.P. LARSON        MAN0040
C     AND MODIFICATIONS NOTED BY ASTERISKS BY J.H.GUSWA AND D.R.LEBLANC MAN0045*
C     ------------------------------------------------------------------MAN0050
C                                                                       MAN0060
C     SPECIFICATIONS:                                                   MAN0070
      REAL *8YSTR                                                       MAN0080
C                                                                       MAN0090
      DIMENSION Y(177500), L(29), HEADNG(33), NAME(42), INFT(2,2), IOFT(MAN0100*
     19,4), DUM(3)                                                      MAN0110
C                                                                       MAN0120
      EQUIVALENCE (YSTR,Y(1))                                           MAN0130
C                                                                       MAN0140
      COMMON /INTEGR/ I0,J0,K0,I1,J1,K1,I,J,K,NPER,KTH,ITMAX,LENGTH,KP,NMAN0150
     1WEL,NUMT,IFINAL,IT,KT,IHEAD,IDRAW,IFLO,IERR,I2,J2,K2,IMAX,ITMX1,NCMAN0160
     2H,IDK1,IDK2,IWATER,IQRE,IP,JP,IQ,JQ,IK,JK,K5,IPU1,IPU2,ITK,IEQN   MAN0170
     3,NKODE,KFLOW                                                      MAN0180*
      COMMON /SPARAM/ TMAX,CDLT,DELT,ERR,TEST,SUM,SUMP,QR,BETA1         MAN0190*
      COMMON /SARRAY/ ICHK(13),LEVEL1(9),LEVEL2(9)                      MAN0200
C                                                                       MAN0210
      DATA NAME/2*4H    ,4H   S,4HTART,4HING ,4HHEAD,4H    ,4H STO,4HRAGMAN0220
     1E,4H COE,4HFFIC,4HIENT,2*4H    ,4H  TR,4HANSM,4HISSI,4HVITY,5*4H  MAN0230
     2  ,4H  TK,4H  HY,4HDRAU,4HLIC ,4HCOND,4HUCTI,4HVITY,2*4H    ,4HBOTMAN0240
     3T,4HOM E,4HLEVA,4HTION,2*4H    ,4H   R,4HECHA,4HRGE ,4HRATE/      MAN0250
      DATA INFT/4H(20F,4H4.0),4H(8F1,4H0.4)/                            MAN0260
      DATA IOFT'4H(1H0,4H,I2,,4H2X,2,4H0F6.,4H1/(5,4HX,20,4HF6.1,4H))  ,MAN0270
     14H    ,4H(1H0,4H,I5,,4H14F9,4H.5/(,4H1H ,,4H5X,1,4H4F9.,4H5)) ,4H MAN0280
     2   ,4H(1H0,4H,I5,,4H10E1,4H2.5/,4H(1H ,4H,5X,,4H10E1,4H2.5),4H)   MAN0290
     3,4H(1H0,4H,I5,,4H10E1,4H1.3/,4H(1H ,4H,5X,,4H10E1,4H1.3),4H)   /  MAN0300
C                                                                       MAN0310
      DEFINE FILE 2(8,1520,U,KKK)                                       MAN0320
C     ..................................................................MAN0330
C                                                                       MAN0340
C     ---READ TITLE, PROGRAM SIZE AND OPTIONS---                        MAN0350
      READ (5,200) HEADNG                                               MAN0360
      WRITE (6,190) HEADNG                                              MAN0370
      READ (5,160) I0,J0,K0,ITMAX,NCH,NKODE,BETA1,KFLOW,NRIV            MAN0380*
      WRITE (6,180) I0,J0,K0,ITMAX,NCH,NKODE,BETA1,KFLOW,NRIV           MAN0390*
      READ (5,210) IDRAW,IHEAD,IFLO,IDK1,IDK2,IWATER,IQRE,IPU1,IPU2,ITK MAN0400
     1,IEQN                                                             MAN0410
      WRITE (6,220) IDRAW,IHEAD,IFLO,IDK1,IDK2,IWATER,IQRE,IPU1,IPU2,ITKMAN0420
     1,IEQN                                                             MAN0430
      IERR=0                                                            MAN0440
C                                                                       MAN0450
C     ---COMPUTE DIMENSIONS FOR ARRAYS---                               MAN0460
      J1=J0-1                                                           MAN0470
      I1=I0-1                                                           MAN0480
      K1=K0-1                                                           MAN0490
      I2=I0-2                                                           MAN0500
      J2=J0-2                                                           MAN0510
      K2=K0-2                                                           MAN0520
      IMAX=MAX0(I0,J0)                                                  MAN0530
      NCD=MAX0(1,NCH)                                                   MAN0540
      ITMX1=ITMAX+1                                                     MAN0550
      ISIZ=I0*J0*K0                                                     MAN0560
      IK1=I0*J0                                                         MAN0570
```

```
      IK2=MAX0(IK1*K1,1)                                            MAN0580
      ISUM=2*ISIZ+1                                                 MAN0590
      L(1)=1                                                        MAN0600
      DO 30 I=2,14                                                  MAN0610
      IF (I.NE.8) GO TO 20                                          MAN0620
      L(8)=ISUM                                                     MAN0630
      ISUM=ISUM+IK2                                                 MAN0640
      IF (IK2.EQ.1) GO TO 10                                        MAN0650
      IK=I0                                                         MAN0660
      JK=J0                                                         MAN0670
      K5=K1                                                         MAN0680
      GO TO 30                                                      MAN0690
   10 IK=1                                                          MAN0700
      JK=1                                                          MAN0710
      K5=1                                                          MAN0720
      GO TO 30                                                      MAN0730
   20 L(I)=ISUM                                                     MAN0740
      ISUM=ISUM+ISIZ                                                MAN0750
   30 CONTINUE                                                      MAN0760
      L(15)=ISUM                                                    MAN0770
      ISUM=ISUM+J0                                                  MAN0780
      L(16)=ISUM                                                    MAN0790
      ISUM=ISUM+I0                                                  MAN0800
      L(17)=ISUM                                                    MAN0810
      ISUM=ISUM+K0                                                  MAN0820
      L(18)=ISUM                                                    MAN0830
      ISUM=ISUM+IMAX                                                MAN0840
      L(19)=ISUM                                                    MAN0850
      ISUM=ISUM+K0*3                                                MAN0860
      L(20)=ISUM                                                    MAN0870
      ISUM=ISUM+ITMX1                                               MAN0880
      L(21)=ISUM                                                    MAN0890
      ISUM=ISUM+3*NCD                                               MAN0900
      L(22)=ISUM                                                    MAN0910
      ISUM=ISUM+NCD                                                 MAN0920
      L(23)=ISUM                                                    MAN0930
      IF (IWATER.NE.ICHK(6)) GO TO 40                               MAN0940
      ISUM=ISUM+IK1                                                 MAN0950
      L(24)=ISUM                                                    MAN0960
      ISUM=ISUM+IK1                                                 MAN0970
      IP=I0                                                         MAN0980
      JP=J0                                                         MAN0990
      GO TO 50                                                      MAN1000
   40 ISUM=ISUM+1                                                   MAN1010
      L(24)=ISUM                                                    MAN1020
      ISUM=ISUM+1                                                   MAN1030
      IP=1                                                          MAN1040
      JP=1                                                          MAN1050
   50 L(25)=ISUM                                                    MAN1060
      IF (IQRE.NE.ICHK(7)) GO TO 60                                 MAN1070
      ISUM=ISUM+IK1                                                 MAN1080
      IQ=I0                                                         MAN1090
      JQ=J0                                                         MAN1100
      GO TO 70                                                      MAN1110
   60 ISUM=ISUM+1                                                   MAN1120
      IQ=1                                                          MAN1130
      JQ=1                                                          MAN1140
```

```
   70 IF(NRIV.EQ.0) GO TO 75                                            MAN1150*
      L(26)=ISUM                                                        MAN1160*
      ISUM=ISUM+ISIZ                                                    MAN1170*
      L(27)=ISUM                                                        MAN1180*
      ISUM=ISUM+NRIV                                                    MAN1190*
      L(28)=ISUM                                                        MAN1200*
      ISUM=ISUM+NRIV                                                    MAN1210*
      L(29)=ISUM                                                        MAN1220*
      ISUM=ISUM+NRIV                                                    MAN1230*
      GO TO 79                                                          MAN1240*
   75 L(26)=ISUM                                                        MAN1250*
      L(27)=ISUM+1                                                      MAN1260*
      L(28)=ISUM+2                                                      MAN1270*
      L(29)=ISUM+3                                                      MAN1280*
      ISUM=ISUM+4                                                       MAN1290*
   79 WRITE (6,170) ISUM                                                MAN1300*
C                                                                       MAN1310
C     ---PASS INITIAL ADDRESSES OF ARRAYS TO SUBROUTINES---             MAN1320
      CALL DATAI(Y(L(1)),Y(L(2)),Y(L(3)),Y(L(4)),Y(L(5)),Y(L(6)),Y(L(7))MAN1330
     1,Y(L(8)),Y(L(9)),Y(L(15)),Y(L(16)),Y(L(17)),Y(L(19)),Y(L(23)),Y(L(MAN1340
     224)),Y(L(25)),                                                    MAN1350
     3Y(L(26)),Y(L(27)),Y(L(28)),Y(L(29)))                              MAN1360*
      CALL STEP(Y(L(1)),Y(L(2)),Y(L(3)),Y(L(4)),Y(L(5)),Y(L(6)),Y(L(7)),MAN1370
     1Y(L(8)),Y(L(9)),Y(L(15)),Y(L(16)),Y(L(17)),Y(L(19)),Y(L(18)),Y(L(2MAN1380
     20)))                                                              MAN1390
      CALL SOLVE(Y(L(1)),Y(L(2)),Y(L(3)),Y(L(4)),Y(L(5)),Y(L(6)),Y(L(7))MAN1400
     1,Y(L(8)),Y(L(9)),Y(L(15)),Y(L(16)),Y(L(17)),Y(L(19)),Y(L(10)),Y(L(MAN1410
     211)),Y(L(12)),Y(L(13)),Y(L(14)),Y(L(20)),Y(L(25)),Y(L(24)),       MAN1420
     3Y(L(23)),Y(L(26)),Y(L(27)),Y(L(28)),Y(L(29)),NRIV)                MAN1430*
      CALL COEF(Y(L(1)),Y(L(2)),Y(L(3)),Y(L(4)),Y(L(5)),Y(L(6)),Y(L(7)),MAN1440
     1Y(L(8)),Y(L(9)),Y(L(15)),Y(L(16)),Y(L(17)),Y(L(19)),Y(L(23)),Y(L(2MAN1450
     24)),Y(L(25)))                                                     MAN1460
      CALL CHECKI(Y(L(1)),Y(L(2)),Y(L(3)),Y(L(4)),Y(L(5)),Y(L(6)),Y(L(7)MAN1470
     1),Y(L(8)),Y(L(9)),Y(L(15)),Y(L(16)),Y(L(17)),Y(L(19)),Y(L(21)),Y(LMAN1480
     2(22)),Y(L(25)),                                                   MAN1490
     3Y(L(26)).Y(L(27)),Y(L(28)),Y(L(29)),NRIV)                         MAN1500*
      CALL PRNTAI(Y(L(1)),Y(L(2)),Y(L(4)),Y(L(5)),Y(L(9)),Y(L(15)),Y(L(1MAN1510
     16)))                                                              MAN1520
C                                                                       MAN1530
C     ---START COMPUTATIONS---                                          MAN1540
C     ************************                                          MAN1550
C     ---READ AND WRITE DATA FOR GROUPS II AND III---                   MAN1560
      CALL DATAIN                                                       MAN1570
      IRN=1                                                             MAN1580
      NIJ=IO*JO                                                         MAN1590
      DO 80 K=1,KO                                                      MAN1600
      LOC=L(2)+(K-1)*NIJ                                                MAN1610
   80 CALL ARRAY(Y(LOC),INFT(1,2),IOFT(1,1),NAME(1),IRN,DUM)            MAN1620
      DO 90 K=1,KO                                                      MAN1630
      LOC=L(5)+(K-1)*NIJ                                                MAN1640
   90 CALL ARRAY(Y(LOC),INFT(1,1),IOFT(1,2),NAME(7),IRN,DUM)            MAN1650
      DO 100 K=1,KO                                                     MAN1660
      LOC=L(4)+(K-1)*NIJ                                                MAN1670
      L1=L(19)+K-1                                                      MAN1680
      L2=L(19)+KO+K-1                                                   MAN1690
      L3=L(19)+2*KO+K-1                                                 MAN1700
      CALL ARRAY(Y(LOC),INFT(1,2),IOFT(1,2),NAME(13),IRN,DUM)           MAN1710
```

```
      Y(L1)=DUM(1)
      Y(L2)=DUM(2)
      Y(L3)=DUM(3)
  100 WRITE (6,230) K,Y(L1),Y(L2),Y(L3)
      IF (ITK.NE.ICHK(10)) GO TO 120
      DO 110 K=1,K1
      LOC=L(8)+(K-1)*NIJ
  110 CALL ARRAY(Y(LOC),INFT(1,2),IOFT(1,3),NAME(19),IRN,DUM)
  120 IF (IWATER.NE.ICHK(6)) GO TO 130
      K = K0
      CALL ARRAY(Y(L(23)),INFT(1,2),IOFT(1,4),NAME(25),IRN,DUM)
      CALL ARRAY(Y(L(24)),INFT(1,1),IOFT(1,1),NAME(31),IRN,DUM)
  130 IF (IQRE.EQ.ICHK(7)) CALL ARRAY(Y(L(25)),INFT(1,1),IOFT(1,4),NAME(
     137),IRN,DUM)
      CALL MDAT
      IF(NRIV.NE.0) CALL DDAT2(NRIV)
C
C     ---COMPUTE TRANSMISSIVITY FOR UNCONFINED LAYER---
      IF (IWATER.EQ.ICHK(6)) CALL TRANS(1)
C
C     ---COMPUTE T COEFFICIENTS---
      CALL TCOF
C
C     ---COMPUTE ITERATION PARAMETERS---
      CALL ITER
C
C     ---READ TIME PARAMETERS AND PUMPING DATA FOR A NEW PUMPING PERIOD-
  140 CALL NEWPER
C
      KT=0
      IFINAL=0
C
C     ---START NEW TIME STEP COMPUTATIONS---
  150 CALL NEWSTP
C
C     ---START NEW ITERATION IF MAXIMUM NO. ITERATIONS NOT EXCEEDED---
      CALL NEWITA
C
C     ---PRINT OUTPUT AT DESIGNATED TIME STEPS---
      CALL OUTPUT
C
C     ---LAST TIME STEP IN PUMPING PERIOD ?---
      IF (IFINAL.NE.1) GO TO 150
C
C     ---CHECK FOR NEW PUMPING PERIOD---
      IF (KP.LT.NPER) GO TO 140
C
      STOP
C
C     ---FORMATS---
C
C
C
  160 FORMAT (6I10,G10.0,2I5)
  170 FORMAT ('0',54X,'WORDS OF VECTOR Y USED =',I7)
  180 FORMAT ('0',62X,'NUMBER OF ROWS =',I5/60X,'NUMBER OF COLUMNS =',I5
     1/61X,'NUMBER OF LAYERS =',I5//39X,'MAXIMUM PERMITTED NUMBER OF ITE
```

```
   2RATIONS =',I5//48X,'NUMBER OF CONSTANT HEAD NODES =',I5//72X,       MAN2290
   3'NKODE =',I5//53X,'BETA ITERATION PARAMETER =',F5.2,                MAN2300*
   4//72X,'KFLOW =',I5,                                                 MAN2310*
   5//56X,'NUMBER OF LEAKY NODES =',I5)                                 MAN2320*
190 FORMAT ('1',33A4)                                                   MAN2330
200 FORMAT (20A4)                                                       MAN2340
210 FORMAT (16(A4,1X))                                                  MAN2350
220 FORMAT ('-SIMULATION OPTIONS:  ',11(A4,4X))                         MAN2360
230 FORMAT (1H0,44X,'DIRECTIONAL TRANSMISSIVITY MULTIPLICATION FACTORSMAN2370
   1 FOR LAYER',I3,/,76X,'X =',G15.7/76X,'Y =',G15.7/76X,'Z =',G15.7) MAN2380
    END                                                                 MAN2390
```

```
      SUBROUTINE DATAI(PHI,STRT,OLD,T,S,TR,TC,TK,WELL,DELX,DELY,DELZ,FAC
     1T,PERM,BOTTOM,QRE,
     2IDR,RH,RC,RB)
C     ------------------------------------------------------------------
C     READ AND WRITE DATA
C     ------------------------------------------------------------------
C
C     SPECIFICATIONS:
      REAL *8PHI
      REAL *8XLABEL,YLABEL,TITLE,XN1,MESUR
C
      DIMENSION PHI(I0,J0,K0), STRT(I0,J0,K0), OLD(I0,J0,K0), T(I0,J0,K0
     1), S(I0,J0,K0), TR(I0,J0,K0), TC(I0,J0,K0), TK(IK,JK,K5), WELL(I0,
     2J0,K0), DELX(J0), DELY(I0), DELZ(K0), FACT(K0,3), PERM(IP,JP), BOT
     3TOM(IP,JP), QRE(IQ,JQ), TF(3), A(I0,J0), IN(6), IOFT(9), INFT(2)
     4,IDR(I0,J0,K0),RH(1),RC(1),RB(1)
C
      COMMON /INTEGR/ I0,J0,K0,I1,J1,K1,I,J,K,NPER,KTH,ITMAX,LENGTH,KP,N
     1WEL,NUMT,IFINAL,IT,KT,IHEAD,IDRAW,IFLO,IERR,I2,J2,K2,IMAX,ITMX1,NC
     2H,IDK1,IDK2,IWATER,IQRE,IP,JP,IQ,JQ,IK,JK,K5,IPU1,IPU2,ITK,IEQN
     3,NKODE,KFLOW
      COMMON /SPARAM/ TMAX,CDLT,DELT,ERR,TEST,SUM,SUMP,QR,BETA1
      COMMON /SARRAY/ ICHK(13),LEVEL1(9),LEVEL2(9)
      COMMON /CK/ RLFLXT,STORT,QRET,CHST,CHDT,FLUXT,PUMPT,CFLUXT,FLXNT
      COMMON /PR/ XLABEL(3),YLABEL(6),TITLE(6),XN1,MESUR,PRNT(122),BLANK
     1(60),DIGIT(122),VF1(6),VF2(6),VF3(7),XSCALE,DINCH,SYM(17),XN(100),
     2YN(13),NA(4),N1,N2,N3,YSCALE,FACT1,FACT2
      RETURN
C     ..................................................................
C     ********************
      ENTRY DATAIN
C     ********************
C
C     ---READ AND WRITE SCALAR PARAMETERS---
      READ (5,330) NPER,KTH,ERR,LENGTH
      WRITE (6,340) NPER,KTH,ERR
      READ (5,460) XSCALE,YSCALE,DINCH,FACT1,(LEVEL1(I),I=1,9),FACT2,(LE
     1VEL2(I),I=1,9),MESUR
      IF (XSCALE.NE.0.) WRITE (6,470) XSCALE,YSCALE,MESUR,MESUR,DINCH,FA
     1CT1,LEVEL1,FACT2,LEVEL2
C
C     ---READ CUMULATIVE MASS BALANCE PARAMETERS---
      READ (5,450) SUM,SUMP,PUMPT,CFLUXT,QRET,CHST,CHDT,FLUXT,STORT,ETFL
     1XT,FLXNT
      IF (IDK1.EQ.ICHK(4)) GO TO 20
      IF (IPU1.NE.ICHK(8)) GO TO 50
C
C     ---READ INITIAL HEAD VALUES FROM CARDS---
      DO 10 K=1,K0
      DO 10 I=1,I0
   10 READ (5,360) (PHI(I,J,K),J=1,J0)
      GO TO 30
C
C     ---READ INITIAL HEAD AND MASS BALANCE PARAMETERS FROM DISK---
   20 READ (4) PHI,SUM,SUMP,PUMPT,CFLUXT,QRET,CHST,CHDT,FLUXT,STORT,ETFL
     1XT,FLXNT
      REWIND 4
```

```
   30 WRITE (6,430) SUM
      DO 40 K=1,KO
      WRITE (6,440) K
      DO 40 I=1,IO
   40 WRITE (6,350) I,(PHI(I,J,K),J=1,JO)
C
   50 DO 60 K=1,KO
      DO 60 I=1,IO
      DO 60 J=1,JO
      WELL(I,J,K)=0.
      TR(I,J,K)=0.
      TC(I,J,K)=0.
      IF (K.NE.KO) TK(I,J,K)=0.
   60 CONTINUE
      RETURN
C     ********************
      ENTRY ARRAY(A,INFT,IOFT,IN,IRN,TF)
C     ********************
      READ (5,330) FAC,IVAR,IPRN,TF,IRECS,IRECD
      IC=4*IRECS+2*IVAR+IPRN+1
      GO TO (70,70,90,90,120,120), IC
   70 DO 80 I=1,IO
      DO 80 J=1,JO
   80 A(I,J)=FAC
      WRITE (6,280) IN,FAC,K
      GO TO 140
   90 IF (IC.EQ.3) WRITE (6,290) IN,K
      DO 110 I=1,IO
      READ (5,INFT) (A(I,J),J=1,JO)
      DO 100 J=1,JO
  100 A(I,J)=A(I,J)*FAC
  110 IF (IC.EQ.3) WRITE (6,IOFT) I,(A(I,J),J=1,JO)
      GO TO 140
  120 READ (2'IRN) A
      IF (IC.EQ.6) GO TO 140
      WRITE (6,290) IN,K
      DO 130 I=1,IO
  130 WRITE (6,IOFT) I,(A(I,J),J=1,JO)
  140 IF (IRECD.EQ.1) WRITE (2'IRN) A
      IRN=IRN+1
      RETURN
C     ********************
      ENTRY MDAT
C     ********************
      DO 150 K=1,KO
      DO 150 I=1,IO
      DO 150 J=1,JO
      IF (I.EQ.1.OR.I.EQ.IO.OR.J.EQ.1.OR.J.EQ.JO) T(I,J,K)=0.
      IF (IDK1.NE.ICHK(4).AND.IPU1.NE.ICHK(8)) PHI(I,J,K)=STRT(I,J,K)
      IF (K.NE.KO.OR.IWATER.NE.ICHK(6)) GO TO 150
      IF (I.EQ.1.OR.I.EQ.IO.OR.J.EQ.1.OR.J.EQ.JO) PERM(I,J)=0.
  150 CONTINUE
      READ (5,330) FAC,IVAR,IPRN
C     ............................ DELX ............................
      IF (IVAR.EQ.1) READ (5,330) (DELX(J),J=1,JO)
      DO 170 J=1,JO
      IF (IVAR.NE.1) GO TO 160
```

```
      DELX(J)=DELX(J)*FAC                                               DAT1150
      GO TO 170                                                         DAT1160
  160 DELX(J)=FAC                                                       DAT1170
  170 CONTINUE                                                          DAT1180
      IF (IVAR.EQ.1.AND.IPRN.NE.1) WRITE (6,370) (DELX(J),J=1,J0)       DAT1190
      IF (IVAR.EQ.0) WRITE (6,300) FAC                                  DAT1200
C     ............................. DELY ..............................DAT1210
      READ (5,330) FAC,IVAR,IPRN                                        DAT1220
      IF (IVAR.EQ.1) READ (5,330) (DELY(I),I=1,I0)                      DAT1230
      DO 190 I=1,I0                                                     DAT1240
      IF (IVAR.NE.1) GO TO 180                                          DAT1250
      DELY(I)=DELY(I)*FAC                                               DAT1260
      GO TO 190                                                         DAT1270
  180 DELY(I)=FAC                                                       DAT1280
  190 CONTINUE                                                          DAT1290
      IF (IVAR.EQ.1.AND.IPRN.NE.1) WRITE (6,380) (DELY(I),I=1,I0)       DAT1300
      IF (IVAR.EQ.0) WRITE (6,310) FAC                                  DAT1310
C     ............................. DELZ ..............................DAT1320
      READ (5,330) FAC,IVAR,IPRN                                        DAT1330
      IF (IVAR.EQ.1) READ (5,330) (DELZ(K),K=1,K0)                      DAT1340
      DO 210 K=1,K0                                                     DAT1350
      IF (IVAR.NE.1) GO TO 200                                          DAT1360
      DELZ(K)=DELZ(K)*FAC                                               DAT1370
      GO TO 210                                                         DAT1380
  200 DELZ(K)=FAC                                                       DAT1390
  210 CONTINUE                                                          DAT1400
      IF (IVAR.EQ.1.AND.IPRN.NE.1) WRITE (6,390) (DELZ(K),K=1,K0)       DAT1410
      IF (IVAR.EQ.0) WRITE (6,320) FAC                                  DAT1420
C                                                                       DAT1430
C     ---INITIALIZE VARIABLES---                                        DAT1440
      B=0.                                                              DAT1450
      D=0.                                                              DAT1460
      F=0.                                                              DAT1470
      H=0.                                                              DAT1480
      SU=0.                                                             DAT1490
      Z=0.                                                              DAT1500
      IF (XSCALE.NE.0.) CALL MAP                                        DAT1510
      RETURN                                                            DAT1520
      ENTRY DDAT2(NRIV)                                                 DAT1530*
      NK=1                                                              DAT1540*
      DO 580 K=1,K0                                                     DAT1550*
      DO 580 I=1,I0                                                     DAT1560*
      READ(5,510) (IDR(I,J,K),J=1,J0)                                   DAT1570*
      DO 580 J=1,J0                                                     DAT1580*
      IF(IDR(I,J,K).EQ.0) GO TO 580                                     DAT1590*
      IDR(I,J,K)=NK                                                     DAT1600*
      NK=NK+1                                                           DAT1610*
  580 CONTINUE                                                          DAT1620*
      NK=NK-1                                                           DAT1630*
      IF(NK.EQ.NRIV) GO TO 600                                          DAT1640*
      WRITE(6,520) NK,NRIV                                              DAT1650*
      STOP                                                              DAT1660*
  600 READ(5,330) FAC                                                   DAT1670*
      READ(5,560) (RH(I),I=1,NRIV)                                      DAT1680*
      DO 610 I=1,NRIV                                                   DAT1690*
  610 RH(I)=RH(I)*FAC                                                   DAT1700*
      WRITE(6,540) (RH(I),I=1,NRIV)                                     DAT1710*
```

```
      READ(5,330) FAC                                                   DAT1720*
      READ(5,560) (RB(I),I=1,NRIV)                                      DAT1730*
      DO 620 I=1,NRIV                                                   DAT1740*
  620 RB(I)=RB(I)*FAC                                                   DAT1750*
      WRITE(6,545) (RB(I),I=1,NRIV)                                     DAT1760*
      READ(5,330) FAC                                                   DAT1770*
      READ(5,625) (RC(I),I=1,NRIV)                                      DAT1780*
      DO 630 I=1,NRIV                                                   DAT1790*
  630 RC(I)=RC(I)*FAC                                                   DAT1800*
      WRITE(6,635) (RC(I),I=1,NRIV)                                     DAT1810*
      RETURN                                                            DAT1820*
C     ..................................................................DAT1830
C     ---READ TIME PARAMETERS AND PUMPING DATA FOR A NEW PUMPING PERIOD-DAT1840
C     *********************                                             DAT1850
      ENTRY NEWPER                                                      DAT1860
C     *********************                                             DAT1870
C                                                                       DAT1880
      READ (5,330) KP,KPM1,NWEL,TMAX,NUMT,CDLT,DELT                     DAT1890
C                                                                       DAT1900
C     ---COMPUTE ACTUAL DELT AND NUMT---                                DAT1910
      DT=DELT/24.                                                       DAT1920
      TM=0.0                                                            DAT1930
      DO 220 I=1,NUMT                                                   DAT1940
      DT=CDLT*DT                                                        DAT1950
      TM=TM+DT                                                          DAT1960
      IF (TM.GE.TMAX) GO TO 230                                         DAT1970
  220 CONTINUE                                                          DAT1980
      GO TO 240                                                         DAT1990
  230 DELT=TMAX/TM*DELT                                                 DAT2000
      NUMT=I                                                            DAT2010
  240 WRITE (6,400) KP,TMAX,NUMT,DELT,CDLT                              DAT2020
      DELT=DELT*3600.                                                   DAT2030
      TMAX=TMAX*86400.                                                  DAT2040
      SUMP=0.0                                                          DAT2050
C                                                                       DAT2060
C     ---READ AND WRITE WELL PUMPING RATES---                           DAT2070
      WRITE (6,410) NWEL                                                DAT2080
      IF (NWEL.EQ.0) GO TO 260                                          DAT2090
      DO 245 K = 1,KO                                                   DAT2100
      DO 245 I = 1,IO                                                   DAT2110
      DO 245 J = 1,JO                                                   DAT2120
  245 WELL(I,J,K) = 0.0                                                 DAT2130
      DO 250 II=1,NWEL                                                  DAT2140
      READ (5,330) K,I,J,WELL(I,J,K)                                    DAT2150
      WRITE (6,420) K,I,J,WELL(I,J,K)                                   DAT2160
  250 WELL(I,J,K)=WELL(I,J,K)/(DELX(J)*DELY(I))                         DAT2170
  260 RETURN                                                            DAT2180
C                                                                       DAT2190
C     ---FORMATS---                                                     DAT2200
C                                                                       DAT2210
C                                                                       DAT2220
C                                                                       DAT2230
  280 FORMAT (1H0,52X,6A4,' =',G15.7,' FOR LAYER',I3)                   DAT2240
  290 FORMAT (1H1,45X,6A4,' MATRIX, LAYER',I3/46X,41('-'))              DAT2250
  300 FORMAT ('0',72X,'DELX =',G15.7)                                   DAT2260
  310 FORMAT ('0',72X,'DELY =',G15.7)                                   DAT2270
  320 FORMAT ('0',72X,'DELZ =',G15.7)                                   DAT2280
```

```
330 FORMAT (8G10.0)                                                     DAT2290
340 FORMAT ('0',51X,'NUMBER OF PUMPING PERIODS =',I5/49X,'TIME STEPS BDAT2300
   1ETWEEN PRINTOUTS =',I5//51X,'ERROR CRITERIA FOR CLOSURE =',G15.7/)DAT2310
350 FORMAT ('0',I2,2X,20F6.1/(5X,20F6.1))                               DAT2320
360 FORMAT (8F10.4)                                                     DAT2330
370 FORMAT (1H1,46X,40HGRID SPACING IN PROTOTYPE IN X DIRECTION/47X,40DAT2340
   1('-')//('0',12F10.0))                                               DAT2350
380 FORMAT (1H-,46X,40HGRID SPACING IN PROTOTYPE IN Y DIRECTION/47X,40DAT2360
   1('-')//('0',12F10.0))                                               DAT2370
390 FORMAT (1H-,46X,40HGRID SPACING IN PROTOTYPE IN Z DIRECTION/47X,40DAT2380
   1('-')//('0',12F10.0))                                               DAT2390
400 FORMAT ('-',50X,'PUMPING PERIOD NO.',I4,':',F10.2,' DAYS'/51X,38('DAT2400
   1-')//53X,'NUMBER OF TIME STEPS=',I6//59X,'DELT IN HOURS =',F10.3//DAT2410
   253X,'MULTIPLIER FOR DELT =',F10.3)                                  DAT2420
410 FORMAT ('-',63X,I4,' WELLS'/65X,9('-')//50X,'K',9X,'I',9X,'J    PUDAT2430
   1MPING RATE'/)                                                       DAT2440
420 FORMAT (41X,3I10,2F13.2)                                            DAT2450
430 FORMAT ('-',40X,' CONTINUATION - HEAD AFTER ',G20.7,' SEC PUMPING DAT2460
   1'/42X,58('-'))                                                      DAT2470
440 FORMAT ('1',55X,'INITIAL  HEAD MATRIX, LAYER',I3/56X,30('-'))      DAT2480
450 FORMAT (4G20.10)                                                    DAT2490
460 FORMAT (3G10.0,2(G10.0,9I1,1X),A8)                                  DAT2500
470 FORMAT ('0',30X,'ON ALPHAMERIC MAP:'/40X,'MULTIPLICATION FACTOR FODAT2510
   1R X DIMENSION =',G15.7/40X,'MULTIPLICATION FACTOR FOR Y DIMENSION DAT2520
   2=',G15.7/55X,'MAP SCALE IN UNITS OF  ',A11/50X,'NUMBER OF ',A8,' PDAT2530
   3ER INCH =',G15.7/43X,'MULTIPLICATION FACTOR FOR DRAWDOWN =',G15.7,DAT2540
   4' PRINTED FOR LAYERS',9I2/47X,'MULTIPLICATION FACTOR FOR HEAD =',GDAT2550
   515.7,' PRINTED FOR LAYERS',9I2)                                     DAT2560
510 FORMAT(80I1)                                                        DAT2570*
520 FORMAT('     ERROR****   NK.NE.NRIV    NK=',I5,5X,'NRIV=',I5)      DAT2580*
540 FORMAT('1',54X,'RH MATRIX',/'0',54X,9('-'),//(' ',10(E10.4,2X)))  DAT2590*
545 FORMAT('1',54X,'RB MATRIX',/'0',54X,9('-'),//(' ',10(E10.4,2X)))  DAT2600*
560 FORMAT(20F4.0)                                                      DAT2610*
625 FORMAT(10F8.0)                                                      DAT2620*
635 FORMAT('1',54X,'RC MATRIX',/'0',54X,9('-'),//(' ',10(E10.4,2X)))  DAT2630*
    END                                                                 DAT2640
```

```
      SUBROUTINE STEP(PHI,STRT,OLD,T,S,TR,TC,TK,WELL,DELX,DELY,DELZ,FACT
     1,DDN,TEST3)
C     -----------------------------------------------------------------
C     INITIALIZE DATA FOR A NEW TIME STEP AND PRINT RESULTS
C     -----------------------------------------------------------------
C
C     SPECIFICATIONS:
      REAL *8PHI
      REAL *8XLABEL,YLABEL,TITLE,XN1,MESUR
C
      DIMENSION PHI(I0,J0,K0), STRT(I0,J0,K0), OLD(I0,J0,K0), T(I0,J0,K0
     1), S(I0,J0,K0), TR(I0,J0,K0), TC(I0,J0,K0), TK(IK,JK,K5), WELL(I0,
     2J0,K0), DELX(J0), DELY(I0), DELZ(K0), FACT(K0,3), DDN(IMAX), TEST3
     3(ITMX1), ITTO(50)
C
      COMMON /INTEGR/ I0,J0,K0,I1,J1,K1,I,J,K,NPER,KTH,ITMAX,LENGTH,KP,N
     1WEL,NUMT,IFINAL,IT,KT,IHEAD,IDRAW,IFLO,IERR,I2,J2,K2,IMAX,ITMX1,NC
     2H,IDK1,IDK2,IWATER,IQRE,IP,JP,IQ,JQ,IK,JK,K5,IPU1,IPU2,ITK,IEQN
     3,NKODE,KFLOW
      COMMON /SPARAM/ TMAX,CDLT,DELT,ERR,TEST,SUM,SUMP,QR,BETA1
      COMMON /SARRAY/ ICHK(13),LEVEL1(9),LEVEL2(9)
      COMMON /CK/ RLFLXT,STORT,QRET,CHST,CHDT,FLUXT,PUMPT,CFLUXT,FLXNT
      COMMON /PR/ XLABEL(3),YLABEL(6),TITLE(6),XN1,MESUR,PRNT(122),BLANK
     1(60),DIGIT(122),VF1(6),VF2(6),VF3(7),XSCALE,DINCH,SYM(17),XN(100),
     2YN(13),NA(4),N1,N2,N3,YSCALE,FACT1,FACT2
      RETURN
C     .................................................................
C     ********************
      ENTRY NEWSTP
C     ********************
      KT=KT+1
      IT=0
      DO 10 K=1,K0
      DO 10 I=1,I0
      DO 10 J=1,J0
   10 OLD(I,J,K)=PHI(I,J,K)
      DELT=CDLT*DELT
      SUM=SUM+DELT
      SUMP=SUMP+DELT
      DAYSP=SUMP/86400.
      YRSP=DAYSP/365.
      HRS=SUM/3600.
      SMIN=HRS*60.
      DAYS=HRS/24.
      YRS=DAYS/365.
      RETURN
C
C     ---PRINT OUTPUT AT DESIGNATED TIME STEPS---
C     *********************
      ENTRY OUTPUT
C     *********************
      IF (KT.EQ.NUMT) IFINAL=1
      ITTO(KT)=IT
      IF (IT.LE.ITMAX) GO TO 20
      IT=IT-1
      ITTO(KT)=IT
      IERR=2
```

```
C
C     ---IF MAXIMUM ITERATIONS EXCEEDED,WRITE RESULTS ON DISK OR CARDS--
      IF (IDK2.EQ.ICHK(5)) WRITE (4) PHI,SUM,SUMP,PUMPT,CFLUXT,QRET,CHST
     1,CHDT,FLUXT,STORT,RLFLXT,FLXNT
      IF (IPU2.EQ.ICHK(9)) WRITE (7,230) SUM,SUMP,PUMPT,CFLUXT,QRET,CHST
     1,CHDT,FLUXT,STORT,RLFLXT,FLXNT
C
   20 IF (IFLO.EQ.ICHK(3)) CALL CHECK
      IF (IERR.EQ.2) GO TO 30
      IF (MOD(KT,KTH).NE.0.AND.IFINAL.NE.1) RETURN
   30 WRITE (6,210) KT,DELT,SUM,SMIN,HRS,DAYS,YRS,DAYSP,YRSP
      IF (IFLO.EQ.ICHK(3)) CALL CWRITE
      IT=IT+1
      WRITE (6,180) (TEST3(J),J=1,IT)
      I3=1
      I5=0
  352 I5=I5+40
      I4=MIN0(KT,I5)
      WRITE (6,240) (I,I=I3,I4)
      WRITE (6,260)
      WRITE (6,250) (ITTO(I),I=I3,I4)
      WRITE (6,260)
      IF(KT.LE.I5) GO TO 353
      I3=I3+40
      GO TO 352
C
C     ---PRINT MAPS---
  353 IF (XSCALE.EQ.0.) GO TO 70
      IF (FACT1.EQ.0.) GO TO 50
      DO 40 IA=1,9
      II=LEVEL1(IA)
      IF (II.EQ.0) GO TO 50
   40 CALL PRNTA(1,II)
   50 IF (FACT2.EQ.0.) GO TO 70
      DO 60 IA=1,9
      II=LEVEL2(IA)
      IF (II.EQ.0) GO TO 70
   60 CALL PRNTA(2,II)
   70 IF (IDRAW.NE.ICHK(1)) GO TO 100
C
C     ---PRINT DRAWDOWN---
      DO 90 K=1,KO
      WRITE (6,200) K
      DO 90 I=1,IO
      DO 80 J=1,JO
   80 DDN(J)=STRT(I,J,K)-PHI(I,J,K)
   90 WRITE (6,170) I,(DDN(J),J=1,JO)
  100 IF (IHEAD.NE.ICHK(2)) GO TO 120
C
C     ---PRINT HEAD MATRIX---
      DO 110 K=1,KO
      WRITE (6,190) K
      DO 110 I=1,IO
  110 WRITE (6,170) I,(PHI(I,J,K),J=1,JO)
C
C     ---WRITE ON DISK---
  120 IF (IERR.EQ.2) GO TO 130
```

```
      IF (KP.LT.NPER.OR.IFINAL.NE.1) RETURN
      IF (IDK2.EQ.ICHK(5)) WRITE (4) PHI,SUM,SUMP,PUMPT,CFLUXT,QRET,CHST
     1,CHDT,FLUXT,STORT,RLFLXT,FLXNT
C
C     ---PUNCHED OUTPUT---
  130 IF (IPU2.NE.ICHK(9)) GO TO 160
      IF (IERR.EQ.2) GO TO 140
      WRITE (7,230) SUM,SUMP,PUMPT,CFLUXT,QRET,CHST,CHDT,FLUXT,STORT,ETF
     1LXT,FLXNT
  140 DO 150 K=1,KO
      DO 150 I=1,IO
  150 WRITE (7,220)  (PHI(I,J,K),J=1,JO)
  160 IF (IERR.EQ.2) STOP
      RETURN
C
C     ---FORMATS---
C
C
C
  170 FORMAT ('0',I4,18F7.2/(5X,18F7.2))
  180 FORMAT ('0MAXIMUM HEAD CHANGE FOR EACH ITERATION:'/' ',39('-')/('0
     1',10F12.4))
  190 FORMAT ('1',55X,'HEAD MATRIX, LAYER',I3/56X,21('-'))
  200 FORMAT ('1',55X,'   DRAWDOWN, LAYER',I3/59X,18('-'))
  210 FORMAT (1H1,44X,57('-')/45X,'|',14X,'TIME STEP NUMBER =',I9,14X,'|
     1'/45X,57('-')//50X,29HSIZE OF TIME STEP IN SECONDS=,F14.2//55X,'TO
     2TAL SIMULATION TIME IN SECONDS=',F14.2/80X,8HMINUTES=,F14.2/82X,6H
     3HOURS=,F14.2/83X,5HDAYS=,F14.2/82X,'YEARS=',F14.2///45X,'DURATION 
     4OF CURRENT PUMPING PERIOD IN DAYS=',F14.2/82X,'YEARS=',F14.2//)
  220 FORMAT (10F8.4)
  230 FORMAT (4G20.10)
  240 FORMAT ('OTIME STEP :',40I3)
  250 FORMAT ('OITERATIONS:',40I3)
  260 FORMAT (' ',10('-'))
      END
```

```
      SUBROUTINE SOLVE(PHI,STRT,OLD,T,S,TR,TC,TK,WELL,DELX,DELY,DELZ,FACSP30010
     1T,EL,FL,GL,V,XI,TEST3,QRE,BOTTOM,                                 SP30020
     2PERM,IDR,RH,RC,RB,IRIV)                                           SP30030*
C     ------------------------------------------------------------------SP30040
C     SOLUTION BY THE STRONGLY IMPLICIT PROCEDURE                       SP30050
C     ------------------------------------------------------------------SP30060
C                                                                       SP30070
C     SPECIFICATIONS:                                                   SP30080
      REAL *8PHI,RHO,B,D,F,H,Z,SU,RHOP,W,WMIN,RHO1,RHO2,RHO3,XPART,YPARTSP30090
     1,ZPART,DMIN1,WMAX,XT,YT,ZT,DABS,DMAX1,DEN,TXM,TYM,TDEN            SP30100
      REAL *8E,AL,BL,CL,A,C,G,WU,TU,U,DL,RES,SUPH,GLXI,ZPHI             SP30110
C                                                                       SP30120
      DIMENSION PHI(1), STRT(1), OLD(1), T(1), S(1), TR(1), TC(1), TK(1)SP30130
     1, WELL(1), DELX(1), DELY(1), DELZ(1), FACT(K0,3), RHOP(20), TEST3(SP30140
     21), EL(1), FL(1), GL(1), V(1), XI(1), QRE(1),BOTTOM(1)            SP30150
     3,PERM(1),IDR(1),RH(1),RC(1),RB(1)                                 SP30160*
      DIMENSION  IN(36,59,7),TT(36,59,6),DBOT(36,59),ISEA(36,59)        SP30170*
     2,DTOP(9),IAA(30),HEDSTR(10),BTM(36,59),KCHOP2(9),IFNLHD(36,59,7),TSP30180*
     3DEN(8)                                                            SP30183*
C                                                                       SP30190
      COMMON /INTEGR/ I0,J0,K0,I1,J1,K1,I,J,K,NPER,KTH,ITMAX,LENGTH,KP,NSP30200
     1WEL,NUMT,IFINAL,IT,KT,IHEAD,IDRAW,IFLO,IERR,I2,J2,K2,IMAX,ITMX1,NCSP30210
     2H,IDK1,IDK2,IWATER,IQRE,IP,JP,IQ,JQ,IK,JK,K5,IPU1,IPU2,ITK,IEQN   SP30220
     3,NKODE,KFLOW                                                      SP30230*
      COMMON /SPARAM/ TMAX,CDLT,DELT,ERR,TEST,SUM,SUMP,QR,BETA1         SP30240*
      COMMON /SARRAY/ ICHK(13),LEVEL1(9),LEVEL2(9)                      SP30250
      RETURN                                                            SP30260
C     ..................................................................SP30270
C     *********************                                             SP30280
      ENTRY ITER                                                        SP30290
C     *********************                                             SP30300
C     ---COMPUTE AND PRINT ITERATION PARAMETERS---                      SP30310
      WRITE (6,240)                                                     SP30320
      WMIN=1.D0                                                         SP30330
      DELT=1.                                                           SP30340
      P2=LENGTH-1                                                       SP30350
      NT=I0*J0*K0                                                       SP30360
      NIJ=I0*J0                                                         SP30370
      XT=3.141593**2/(2.*J2*J2)                                         SP30380
      YT=3.141593**2/(2.*I2*I2)                                         SP30390
      ZT=3.141593**2/(2.*K0*K0)                                         SP30400
      RHO1=0.D0                                                         SP30410
      RHO2=0.D0                                                         SP30420
      RHO3=0.D0                                                         SP30430
      DO 40 K=1,K0                                                      SP30440
      DO 40 I=2,I1                                                      SP30450
      DO 40 J=2,J1                                                      SP30460
      N=I+(J-1)*I0+(K-1)*NIJ                                            SP30470
      IF (T(N).EQ.0.) GO TO 40                                          SP30480
      D=TR(N-I0)/DELX(J)                                                SP30490
      F=TR(N)/DELX(J)                                                   SP30500
      B=TC(N-1)/DELY(I)                                                 SP30510
      H=TC(N)/DELY(I)                                                   SP30520
      SU=0.D0                                                           SP30530
      Z=0.D0                                                            SP30540
      IF (K.NE.1) Z=TK(N-NIJ)/DELZ(K)                                   SP30550
      IF (K.NE.K0) SU=TK(N)/DELZ(K)                                     SP30560
```

```
   10 CONTINUE
      TXM=DMAX1(D,F)
      TYM=DMAX1(B,H)
      TZM=DMAX1(SU,Z)
      DEN=DMIN1(D,F)
      IF (DEN.EQ.0.D0) DEN=TXM
      IF (DEN.EQ.0.D0) GO TO 20
      RHO1=DMAX1(RHO1,TYM/DEN)
   20 DEN=DMIN1(B,H)
      IF (DEN.EQ.0.D0) DEN=TYM
      IF (DEN.EQ.0.D0) GO TO 30
      RHO2=DMAX1(RHO2,TXM/DEN)
   30 DEN=DMIN1(SU,Z)
      IF (DEN.EQ.0.D0) DEN=TZM
      IF (DEN.EQ.0.D0) GO TO 40
      RHO3=DMAX1(RHO3,TXM/DEN)
   40 CONTINUE
      XPART=XT/(1.D0+RHO1)
      YPART=YT/(1.D0+RHO2)
      ZPART=ZT/(1.D0+RHO3)
      WMIN=DMIN1(WMIN,XPART,YPART,ZPART)
      WMAX=1.D0-WMIN
      PJ=-1.
      DO 50 I=1,LENGTH
      PJ=PJ+1.
   50 RHOP(I)=1.D0-(1.D0-WMAX)**(PJ/P2)
      WRITE (6,230) LENGTH,(RHOP(J),J=1,LENGTH)
      RETURN
C     ..........................................................
C
C     ---INITIALIZE DATA FOR A NEW ITERATION---
   60 IT=IT+1
C
C  PRINT HEAD VALUES FOR SELECTED NODES
C
      ITT=IT-1
      IF(NODKOD.NE.1) GO TO 68
      IF(ITT.LT.1) WRITE(6,2970) (IAA(I),I=1,MNOD)
      J=1
      DO 64 I=3,MNOD,3
      N=IAA(I-2)+(IAA(I-1)-1)*I0+(IAA(I)-1)*NIJ
      HEDSTR(J)=PHI(N)
      J=J+1
   64 CONTINUE
      WRITE(6,2980) ITT,(HEDSTR(J),J=1,NMBRND)
C
   68 IF (IT.LE.ITMAX) GO TO 70
      WRITE (6,220)
      CALL OUTPUT
   70 IF (MOD(IT,LENGTH)) 80,80,90
C     ********************
      ENTRY NEWITA
C     ********************
   80 NTH=0
   90 NTH=NTH+1
      W=RHOP(NTH)
      TEST3(IT+1)=0.
```

```
      TEST=0.0
      BIG=0.
      DO 100 I=1,NT
      EL(I)=0.
      FL(I)=0.
      GL(I)=0.
      V(I)=0.
  100 XI(I)=0.
C
C     ---COMPUTE TRANSMISSIVITY AND T COEFFICIENTS FOR UPPER
C     HYDROLOGIC UNIT WHEN IT IS UNCONFINED---
      IF (IWATER.NE.ICHK(6)) GO TO 110
      CALL TRANS(0)
      CALL TCOF
C
C     ---CHOOSE SIP NORMAL OR REVERSE ALGORITHM---
  110 IF (MOD(IT,2)) 120,120,170
  120 DO 150 K=1,K0
      DO 150 I=2,I1
      DO 150 J=2,J1
      N=I+(J-1)*I0+(K-1)*NIJ
      NIA=N+1
      NIB=N-1
      NJA=N+I0
      NJB=N-I0
      NKA=N+NIJ
      NKB=N-NIJ
C
C     ---SKIP COMPUTATIONS IF NODE OUTSIDE MODEL---
      IF (T(N).EQ.0.     .OR.S(N).LT.0.) GO TO 150
C
C     ---COMPUTE COEFFICIENTS---
      D=TR(NJB)/DELX(J)
      F=TR(N)/DELX(J)
      B=TC(NIB)/DELY(I)
      H=TC(N)/DELY(I)
      SU=0.D0
      Z=0.D0
      IF(K.EQ.1) GO TO 124
      Z=TK(NKB)
      IF(IEQN.EQ.ICHK(11)) Z=Z/DELZ(K)
  124 IF(K.EQ.K0) GO TO 125
      SU=TK(N)
      IF(IEQN.EQ.ICHK(11))SU=SU/DELZ(K)
  125 RHO=S(N)/DELT
      QR=0.
      UXR=0.
      IF (K.NE.K0) GO TO 127
      IF (IQRE.EQ.ICHK(7)) QR=QRE(I+(J-1)*I0)
  127 IF(IRIV.LE.0) GO TO 130
      ND=IDR(N)
      IF(ND.EQ.0) GO TO 130
      IF(PHI(N).GT.RB(ND)) GO TO 128
      QR=QR+RC(ND)*(RH(ND)-RB(ND))
      GO TO 130
  128 UXR=RC(ND)
      QR=QR+RC(ND)*RH(ND)
```

```
C
C     ---SIP NORMAL ALGORITHM---
C     ---FORWARD SUBSTITUTE, COMPUTING INTERMEDIATE VECTOR V---
  130 E=-B-D-F-H-SU-Z-RHO-UXR
      BL=B/(1.+W*(EL(NIB)+GL(NIB)))
      CL=D/(1.+W*(FL(NJB)+GL(NJB)))
      C=BL*EL(NIB)
      G=CL*FL(NJB)
      WU=CL*GL(NJB)
      U=BL*GL(NIB)
      IF (K.EQ.1) GO TO 140
      AL=Z/(1.+W*(EL(NKB)+FL(NKB)))
      A=AL*EL(NKB)
      TU=AL*FL(NKB)
      DL=E+W*(A+C+G+WU+TU+U)-CL*EL(NJB)-BL*FL(NIB)-AL*GL(NKB)
      IF(DABS(DL).LT.1.E-10) WRITE(8,8100) I,J,K,DL,E,W,CL,BL,AL
      IF(DABS(DL).LT.1.0E-10) WRITE(8,8125) B,D,F,H,SU,Z,RHO,UXR
      IF(DABS(DL).LT.1.0E-10) WRITE(8,8130) A,C,G,WU,TU,U
      IF(DABS(DL).LT.1.0E-10) WRITE(8,8150) T(N),T(NIB),T(NIA),T(NJB),
     1T(NJA),T(NKB),T(NKA)
      EL(N)=(F-W*(A+C))/DL
      FL(N)=(H-W*(G+TU))/DL
      GL(N)=(SU-W*(WU+U))/DL
      SUPH=0.D0
      IF (K.NE.K0) SUPH=SU*PHI(NKA)
      RES=-B*PHI(NIB)-D*PHI(NJB)-E*PHI(N)-F*PHI(NJA)-H*PHI(NIA)-SUPH-Z*P
     1HI(NKB)-WELL(N)-RHO*OLD(N)-QR
      RES=RES*BETA1
      V(N)=(RES-AL*V(NKB)-BL*V(NIB)-CL*V(NJB))/DL
      GO TO 150
  140 DL=E+W*(C+G+WU+U)-CL*EL(NJB)-BL*FL(NIB)
      EL(N)=(F-W*C)/DL
      FL(N)=(H-W*G)/DL
      GL(N)=(SU-W*(WU+U))/DL
      SUPH=0.D0
      IF (K.NE.K0) SUPH=SU*PHI(NKA)
      RES=-B*PHI(NIB)-D*PHI(NJB)-E*PHI(N)-F*PHI(NJA)-H*PHI(NIA)-SUPH-WEL
     1L(N)-RHO*OLD(N)-QR
      RES=RES*BETA1
      V(N)=(RES-BL*V(NIB)-CL*V(NJB))/DL
  150 CONTINUE
C
C     ---BACK SUBSTITUTE FOR VECTOR XI---
      DO 160 K=1,K0
      K3=K0-K+1
      DO 160 I=1,I2
      I3=I0-I
      DO 160 J=1,J2
      J3=J0-J
      N=I3+(J3-1)*I0+(K3-1)*NIJ+I-I
      IF (T(N).EQ.0.      .OR.S(N).LT.0.) GO TO 160
      GLXI=0.D0
      IF (K3.NE.K0) GLXI=GL(N)*XI(N+NIJ)
      XI(N)=V(N)-EL(N)*XI(N+I0)-FL(N)*XI(N+1)-GLXI
C
C     ---COMPARE MAGNITUDE OF CHANGE WITH CLOSURE CRITERIA---
      TCHK=ABS(XI(N))
```

```
      IF (TCHK.GT.BIG) BIG=TCHK                                         SP32280
      PHI(N)=PHI(N)+XI(N)                                               SP32290
  160 CONTINUE                                                          SP32300
      IF (BIG.GT.ERR) TEST=1.                                           SP32310
      TEST3(IT+1)=BIG                                                   SP32320
C                                                                       SP32330*
C  ---GROUP I CARDS---                                                  SP32340*
C                                                                       SP32350*
C                                                                       SP32360*
C  CHECK IF THIS IS FIRST TIME STEP OF FIRST PUMPING PERIOD.            SP32370*
C   IF NOT SKIP THIS SECTION.                                           SP32371*
      IF(KP.GT.1) GO TO 2090                                            SP32380*
      IF(KT.GT.1) GO TO 2090                                            SP32385*
C                                                                       SP32390*
C  CHECK IF THIS IS FIRST SOLUIION.  IF IT IS PROCEED WITH INITIAL CHECKSP32400*
      IF(IT.NE.0) GO TO 2090                                            SP32410*
C                                                                       SP32420*
C  INTERFACE DEFINITION:  DEFINE PARAMETERS, READ DATA, INITIALIZE IN   SP32430*
C     ARRAY AND PERFORM FIRST CHOP                                      SP32440*
C                                                                       SP32450*
C                                                                       SP32460*
C  DEFINITION OF PARAMETERS                                             SP32470*
C                                                                       SP32480*
C     NTAD = NUMBER OF ITERATIONS BETWEEM T ADJUSTMENTS                 SP32490*
C     DBOT(I,J) = BASE OF AQUIFER SYSTEM IN TERMS OF + NUMBERS BECAUSE  SP32500*
C      Z IS + DOWNWARD                                                  SP32510*
C     RF = COEFFICIENT OF PHI TO GIVE INTERFACE DEPTH BELOW SEA LEVEL   SP32520*
C        = ((RHOS-RHOF)/RHOF)                                           SP32530*
C     DTOP(K) = DEPTH BELOW SEA LEVEL TO TOP OF LAYERS IN TERMS OF +    SP32540*
C      NUMBERS BECAUSE Z IS + DOWNWARD                                  SP32550*
C     Z2 = DISTANCE TO TOP OF F.D. BLOCK BELOW SEA LEVEL                SP32560*
C     IH = I0/2 FOR TWO-SIDED CHOP, = I1 FOR ONE-SIDED CHOP             SP32570*
C          (E.G. CROSS SECTIONS)                                        SP32580*
C     Z = DEPTH BELOW S.L. TO INTERFACE                                 SP32590*
C     IN = 4 = BLOCK WITH ADJUSTED T                                    SP32600*
C        = 6 = BLOCK DROPPED DURING CHOP                                SP32610*
C        = 7 = BLOCK DROPPED BECAUSE OCEAN LEAKAGE INTO AQUIFER         SP32620*
C        = 8 = INITIAL T = 0                                            SP32630*
C     KODCHP = CODE TO INDICATE WHETHER CHOPPING SHOULD PROCEED FROM    SP32640*
C              TWO SIDES OR ONE (1=ONE-SIDED, NE 1=TWO-SIDED)           SP32650*
C  READ AND WRITE INPUT DATA                                            SP32660*
      READ(5,2901) NTAD,RF,KODCHP,KPRNTT                                SP32670*
      WRITE(6,2902) NTAD,RF,KODCHP,KPRNTT                               SP32680*
C                                                                       SP32690*
C     READ AND PRINT AQUIFER BASE                                       SP32700*
      READ(5,2909) FAC,IVAR,IPRN                                        SP32710*
      IF(IVAR.NE.1) GO TO 2002                                          SP32720*
      DO 2001 I=1,I0                                                    SP32730*
 2001 READ(5,2903)  (DBOT(I,J),J=1,J0)                                  SP32740*
 2002 DO 2004 I=1,I0                                                    SP32750*
      DO 2004 J=1,J0                                                    SP32760*
      IF(IVAR.NE.1) GO TO 2003                                          SP32770*
      DBOT(I,J)=DBOT(I,J)*FAC                                           SP32780*
      GO TO 2004                                                        SP32790*
 2003 DBOT(I,J)=FAC                                                     SP32800*
 2004 CONTINUE                                                          SP32810*
      IF(IPRN.EQ.1) GO TO 2006                                          SP32820*
```

```
      WRITE(6,2904)
      DO 2005 I=1,IO
 2005 WRITE(6,2905) I,(DBOT(I,J),J=1,J0)
C
C     READ AND PRINT CONSTANT HEAD OCEAN NODES
 2006 DO 2007 I=1,IO
 2007 READ(5,2906) (ISEA(I,J),J=1,J0)
      WRITE(6,2907)
      DO 2008 I=1,IO
 2008 WRITE(6,2908) I,(ISEA(I,J),J=1,J0)
C
C  READ AND PRINT DEPTH TO LAYER TOPS
      READ(5,29J9) (DTOP(K),K=1,K0)
      DO 2010 K=1,K0
 2010 WRITE(6,2910) K,DTOP(K)
C  READ NODES FOR WHICH HEAD VALUES AT EACH ITERATION ARE TO BE PRINTED
      READ(5,2909) NODKOD,NMBRND
      IF(NODKOD.NE.1) GO  TO 2012
      MNOD=3*NMBRND
      READ(5,2960) (IAA(I),I=1,MNOD)
C
C  READ NODES FOR WHICH FINAL HEADS ARE TO BE PRINTED INDIVIDUALLY
C
 2012 READ(5,2909) IHEDKD
      IF(IHEDKD.NE.1) GO TO 2014
      DO 2013 K=1,K0
      DO 2013 I=1,IO
 2013 READ(5,2906) (IFNLHD(I,J,K),J=1,J0)
C  CHECK IF INTERFACE SOLUTION.  IF NOT SKIP THIS SECTION
 2014 IF(NKODE.NE.1) GO TO 2095
C
C
C  INITIALIZE IN,TT, AND SCALARS
      IH=IO/2
      IF(KODCHP.EQ.1) IH=I1
      NIJ=IO*J0
      KCHOP2(K0)=0
      DO 2015 K=1,K1
      KCHOP2(K)=0
      DO 2015 I=1,IO
      DO 2015 J=1,J0
      N=I+(J-1)*IO+(K-1)*NIJ
      IN(I,J,K)=0
 2015 TT(I,J,K)=T(N)
C
      DO 2017 I=1,IO
      DO 2017 J=1,J0
      IN(I,J,K0)=0
      M=I+(J-1)*IO
 2017 BTM(I,J)=BOTTOM(M)
C
C   NORTH SIDE CHECK
C
      DO 2026 K=1,K0
      DO 2025 J=2,J1
      DO 2020 I=2,IH
      N=I+(J-1)*IO+(K-1)*NIJ
```

```
      IF(T(N).EQ.0.) IN(I,J,K)=8                                        SP33400*
 2020 CONTINUE                                                          SP33410*
 2025 CONTINUE                                                          SP33420*
 2026 CONTINUE                                                          SP33430*
C                                                                       SP33440*
      IF(KODCHP.EQ.1) GO TO 2038                                        SP33450*
C                                                                       SP33460*
C  SOUTH SIDE CHECK:                                                    SP33470*
C                                                                       SP33480*
      DO 2036 K=1,K0                                                    SP33490*
      DO 2035 J=2,J1                                                    SP33500*
      DO 2030 II=1,IH                                                   SP33510*
      I=I0-II                                                           SP33520*
      N=I+(J-1)*I0+(K-1)*NIJ                                            SP33530*
      IF(T(N).EQ.0.) IN(I,J,K)=8                                        SP33540*
 2030 CONTINUE                                                          SP33550*
 2035 CONTINUE                                                          SP33560*
 2036 CONTINUE                                                          SP33570*
C                                                                       SP33580*
C  PRINT IN ARRAY FOR EACH LAYER                                        SP33590*
 2038 DO 2042 K=1,K0                                                    SP33600*
      WRITE(6,2911) K                                                   SP33610*
      DO 2040 I=1,I0                                                    SP33620*
 2040 WRITE(6,2912) I,(IN(I,J,K),J=1,J0)                                SP33630*
 2042 CONTINUE                                                          SP33640*
C                                                                       SP33650*
C  ADJUST T?                                                            SP33660*
 2090 IF(NKODE.LT.1) GO TO 2095                                         SP33670*
      IF(TEST.EQ.0.) GO TO 2700                                         SP33680*
      IF(IT.EQ.0) GO TO 60                                              SP33690*
      IF(MOD(IT,NTAD).EQ.0) GO TO 2100                                  SP33700*
 2095 IF(TEST.EQ.0.) RETURN                                             SP33710
      GO TO 60                                                          SP33720
C     ..................................................................SP33730
  170 DO 200 KK=1,K0                                                    SP33740
      K=K0-KK+1                                                         SP33750
      DO 200 II=1,I2                                                    SP33760
      I=I0-II                                                           SP33770
      DO 200 J=2,J1                                                     SP33780
      N=I+(J-1)*I0+(K-1)*NIJ                                            SP33790
      NIA=N+1                                                           SP33800
      NIB=N-1                                                           SP33810
      NJA=N+I0                                                          SP33820
      NJB=N-I0                                                          SP33830
      NKA=N+NIJ                                                         SP33840
      NKB=N-NIJ                                                         SP33850
C                                                                       SP33860
C     ---SKIP COMPUTATIONS IF NODE OUTSIDE AQUIFER---                   SP33870
      IF (T(N).EQ.0.     .OR.S(N).LT.0.) GO TO 200                      SP33880
C                                                                       SP33890
C     ---COMPUTE COEFFICIENTS---                                        SP33900
      D=TR(NJB)/DELX(J)                                                 SP33910
      F=TR(N)/DELX(J)                                                   SP33920
      B=TC(NIB)/DELY(I)                                                 SP33930
      H=TC(N)/DELY(I)                                                   SP33940
      SU=0.D0                                                           SP33950
      Z=0.D0                                                            SP33960
```

```
      IF(K.EQ.1) GO TO 174
      Z=TK(NKB)
      IF(IEQN.EQ.ICHK(11)) Z=Z/DELZ(K)
  174 IF(K.EQ.KO) GO TO 175
      SU=TK(N)
      IF(IEQN.EQ.ICHK(11))SU=SU/DELZ(K)
  175 RHO=S(N)/DELT
      QR=0.
      UXR=0.
      IF (K.NE.KO) GO TO 177
      IF (IQRE.EQ.ICHK(7)) QR=QRE(I+(J-1)*IO)
  177 IF(IRIV.LE.0) GO TO 180
      ND=IDR(N)
      IF(ND.EQ.0) GO TO 180
      IF(PHI(N).GT.RB(ND)) GO TO 178
      QR=QR+RC(ND)*(RH(ND)-RB(ND))
      GO TO 180
  178 UXR=RC(ND)
      QR=QR+RC(ND)*RH(ND)
C
C     ---SIP REVERSE ALGORITHM---
C     ---FORWARD SUBSTITUTE, COMPUTING INTERMEDIATE VECTOR V---
  180 E=-B-D-F-H-SU-Z-RHO-UXR
      BL=H/(1.+W*(EL(NIA)+GL(NIA)))
      CL=D/(1.+W*(FL(NJB)+GL(NJB)))
      C=BL*EL(NIA)
      G=CL*FL(NJB)
      WU=CL*GL(NJB)
      U=BL*GL(NIA)
      IF (K.EQ.KO) GO TO 190
      AL=SU/(1.+W*(EL(NKA)+FL(NKA)))
      A=AL*EL(NKA)
      TU=AL*FL(NKA)
      DL=E+W*(C+G+A+WU+TU+U)-AL*GL(NKA)-BL*FL(NIA)-CL*EL(NJB)
      IF(DABS(DL).LT.1.E-10) WRITE(8,8200) I,J,K,DL,E,W,CL,BL,AL
      IF(DABS(DL).LT.1.0E-10) WRITE(8,8125) B,D,F,H,SU,Z,RHO,UXR
      IF(DABS(DL).LT.1.0E-10) WRITE(8,8130) A,C,G,WU,TU,U
      IF(DABS(DL).LT.1.0E-10) WRITE(8,8150) T(N),T(NIB),T(NIA),T(NJB),
     1T(NJA),T(NKB),T(NKA)
      EL(N)=(F-W*(C+A))/DL
      FL(N)=(B-W*(G+TU))/DL
      GL(N)=(Z-W*(WU+U))/DL
      ZPHI=0.D0
      IF (K.NE.1) ZPHI=Z*PHI(NKB)
      RES=-B*PHI(NIB)-D*PHI(NJB)-E*PHI(N)-F*PHI(NJA)-H*PHI(NIA)-SU*PHI(N
     1KA)-ZPHI-WELL(N)-RHO*OLD(N)-QR
      RES=RES*BETA1
      V(N)=(RES-AL*V(NKA)-BL*V(NIA)-CL*V(NJB))/DL
      GO TO 200
  190 DL=E+W*(C+G+WU+U)-BL*FL(NIA)-CL*EL(NJB)
      EL(N)=(F-W*C)/DL
      FL(N)=(B-W*G)/DL
      GL(N)=(Z-W*(WU+U))/DL
      ZPHI=0.D0
      IF (K.NE.1) ZPHI=Z*PHI(NKB)
      RES=-B*PHI(NIB)-D*PHI(NJB)-E*PHI(N)-F*PHI(NJA)-H*PHI(NIA)-ZPHI-WEL
     1L(N)-RHO*OLD(N)-QR
```

```
      RES=RES*BETA1
      V(N)=(RES-BL*V(NIA)-CL*V(NJB))/DL
  200 CONTINUE
C
C     ---BACK SUBSTITUTE FOR VECTOR XI---
      DO 210 K=1,K0
      DO 210 I=2,I1
      DO 210 J=1,J2
      J3=J0-J
      N=I+(J3-1)*I0+(K-1)*NIJ
      IF (T(N).EQ.0.      .OR.S(N).LT.0.) GO TO 210
      GLXI=0.D0
      IF (K.NE.1) GLXI=GL(N)*XI(N-NIJ)
      XI(N)=V(N)-EL(N)*XI(N+I0)-FL(N)*XI(N-1)-GLXI
C
C     ---COMPARE MAGNITUDE OF CHANGE WITH CLOSURE CRITERIA---
      TCHK=ABS(XI(N))
      IF (TCHK.GT.BIG) BIG=TCHK
      PHI(N)=PHI(N)+XI(N)
  210 CONTINUE
      IF (BIG.GT.ERR) TEST=1.
      TEST3(IT+1)=BIG
C
C  ---GROUP II CARDS---
C
      IF(NKODE.LT.1) GO TO 2900
      IF(TEST.EQ.0.) GO TO 2700
      IF(MOD(IT,NTAD).GT.0) GO TO  60
C
C  ADJUSTMENT OF T:
C    NORTH SIDE
C
 2100 WRITE(6,2920) IT
      DO 2300 K=1,K0
      DO 2280 J=2,J1
      DO 2260 I=2,IH
      N=I+(J-1)*I0+(K-1)*NIJ
      M=I+(J-1)*I0
      Z2=DTOP(K)
      IF(T(N).EQ.0.) GO TO 2260
C
C  COMPUTE INTERFACE DEPTH BELOW SEA LEVEL
      Z=RF*PHI(N)
C
C  DROP A BLOCK IF OCEAN LEAKAGE IS INTO AQUIFER
C
C  CHECK LOWER LAYER
      IF(IN(I,J,K-1).EQ.0.OR.IN(I,J,K-1).EQ.4) GO TO 2105
      ND=IDR(N)
      IF(ND.EQ.0) GO TO 2105
      IF(ISEA(I,J).EQ.0) GO TO 2105
      IF(PHI(N).GE.RH(ND)) GO TO 2105
      T(N)=0.
      IN(I,J,K)=7
      RC(ND)=0.
      IF(K.EQ.K0) PERM(M)=0.
C
```

```
      GO TO 2260
C        NEXT FEW STATEMENTS ARE LAYER DEPENDENT
C
 2105 IF(K.EQ.1) GO TO 2110
      IF(K.EQ.K0) GO TO 2130
      GO TO 2120
C
C   LAYER 1
C   DROP A BLOCK WHERE INTERFACE IS ABOVE THE TOP
 2110 IF(Z.LE.Z2) GO TO 2210
C
C  INTERFACE BELOW BLOCK BOTTOM IN LAYER 1
      IF(Z.GE.DBOT(I,J)) GO TO 2240
C
C  ADJUST T IN BLOCK OF LAYER 1
      T(N)=TT(I,J,K)*(Z-Z2)/(DBOT(I,J)-Z2)
      IF(ND.NE.0) T(N)=TT(I,J,K)
      GO TO 2160
C
C
C   MIDDLE LAYERS
C
C   CHECK LOWER LAYER
 2120 IF(IN(I,J,K-1).EQ.0.OR.IN(I,J,K-1).EQ.4) GO TO 2260
C
C   DROP A BLOCK WHERE INTERFACE IS ABOVE THE TOP
      IF(Z.LE.Z2) GO TO 2210
C
C  INTERFACE BELOW BLOCK BOTTOM
      IF(Z.GE.DTOP(K-1)) GO TO 2240
C
C  ADJUST T FOR LAYER K
      T(N)=TT(I,J,K)*(Z-Z2)/(DTOP(K-1)-DTOP(K))
      IF(ND.NE.0) T(N)=TT(I,J,K)
      GO TO 2160
C
C   TOP LAYER
C
C   CHECK LOWER LAYER
 2130 IF(IN(I,J,K-1).EQ.0.OR.IN(I,J,K-1).EQ.4) GO TO 2260
C
C   DROP A BLOCK WHERE INTERFACE IS ABOVE THE TOP
      IF(Z.LE.Z2) GO TO 2210
C
C   INTERFACE BELOW BLOCK BOTTOM
      IF(Z.GE.DTOP(K-1)) GO TO 2240
C
C   ADJUST T FOR LAYER K0
      BOTTOM(M)=Z*(-1.0)
      IF(ND.NE.0) BOTTOM(M)=BTM(I,J)
      T(N)=PERM(M)*(PHI(N)-BOTTOM(M))
C  CHOP NODES A'ID NOTE AN ADJUSTED T
C
 2160 IN(I,J,K)=4
      GO TO 2260
 2210 T(N)=0.
      IN(I,J,K)=6
```

```
      GO TO 2260
 2240 IF(K.NE.K0) T(N)=TT(I,J,K)
      IF(K.EQ.K0) T(N)=PERM(M)*(PHI(N)-BTM(I,J))
      IF(K.EQ.K0) BOTTOM(M)=BTM(I,J)
      IN(I,J,K)=0
 2260 CONTINUE
 2280 CONTINUE
 2300 CONTINUE
C
      IF (KODCHP.EQ.1) GO TO 60
C
C     SOUTH SIDE
C
      DO 2500 K=1,K0
      DO 2480 J=2,J1
      DO 2460 II=1,IH
      I=I0-II
      N=I+(J-1)*I0+(K-1)*NIJ
      M=I+(J-1)*I0
      Z2=DTOP(K)
      IF(T(N).EQ.0.) GO TO 2460
C
C   COMPUTE INTERFACE DEPTH BELOW SEA LEVEL
      Z=RF*PHI(N)
C
C    DROP A BLOCK IF OCEAN LAYER IS LEAKING INTO AQUIFER
C
C    CHECK LOWER LAYER
      IF(IN(I,J,K-1).EQ.0.OR.IN(I,J,K-1).EQ.4) GO TO 2305
      ND=IDR(N)
      IF(ND.EQ.0) GO TO 2305
      IF(ISEA(I,J).EQ.0) GO TO 2305
      IF(PHI(N).GE.RH(ND)) GO TO 2305
      T(N)=0.
      IN(I,J,K)=7
      RC(ND)=0.
      IF(K.EQ.K0) PERM(M)=0.
      GO TO 2460
C
C   THE NEXT FEW STATEMENTS ARE LAYER DEPENDENT
C
 2305 IF(K.EQ.1) GO TO 2310
      IF(K.EQ.K0) GO TO 2330
      GO TO 2320
C
C    LAYER 1
C    DROP A BLOCK WHERE INTERFACE IS ABOVE THE TOP
 2310 IF(Z.LE.Z2) GO TO 2410
C
C   INTERFACE BELOW BLOCK BOTTOM IN LAYER 1
      IF(Z.GE.DBOT(I,J)) GO TO 2440
C
C   ADJUST T IN BLOCK OF LAYER 1
      T(N)=TT(I,J,K)*(Z-Z2)/(DBOT(I,J)-Z2)
      IF(ND.NE.0) T(N)=TT(I,J,K)
      GO TO 2360
C
```

```
C  MIDDLE LAYERS
C
C   CHECK LOWER LAYER
 2320 IF(IN(I,J,K-1).EQ.0.OR.IN(I,J,K-1).EQ.4) GO TO 2460
C
C   DROP A BLOCK WHERE INTERFACE IS ABOVE THE TOP
      IF(Z.LE.Z2) GO TO 2410
C
C  INTERFACE BELOW BLOCK BOTTOM
      IF(Z.GE.DTOP(K-1))GO TO 2440
C
C  ADJUST T FOR LAYER K
      T(N)=TT(I,J,K)*(Z-Z2)/(DTOP(K-1)-DTOP(K))
      IF(ND.NE.0) T(N)=TT(I,J,K)
      GO TO 2360
C
C   TOP LAYER
C
C   CHECK LOWER LAYER
 2330 IF(IN(I,J,K-1).EQ.0.OR.IN(I,J,K-1).EQ.4) GO TO 2460
C
C   DROP A BLOCK WHERE INTERFACE IS ABOVE THE TOP
      IF(Z.LE.Z2) GO TO 2410
C
C   INTERFACE BELOW BLOCK BOTTOM
      IF(Z.GE.DTOP(K-1)) GO TO 2440
C
C   ADJUST T FOR LAYER K0
      BOTTOM(M)=Z*(-1.)
      IF(ND.NE.0) BOTTOM(M)=BTM(I,J)
      T(N)=PERM(M)*(PHI(N)-BOTTOM(M))
C
C  CHOP NODES AND NOTE AN ADJUSTED T
C
 2360 IN(I,J,K)=4
      GO TO 2460
 2410 T(N)=0.
      IN(I,J,K)=6
      GO TO 2460
 2440 IF(K.NE.K0) T(N)=TT(I,J,K)
      IF(K.EQ.K0) T(N)=PERM(M)*(PHI(N)-BTM(I,J))
      IF(K.EQ.K0) BOTTOM(M)=BTM(I,J)
      IN(I,J,K)=0
 2460 CONTINUE
 2480 CONTINUE
 2500 CONTINUE
C
 2600 GO TO 60
C
C
C  PRINT CONTENTS OF IN ARRAY AND LOCATION OF ADJUSTED T
 2700 DO 2710 K=1,K0
      DO 2710 I=2,I1
      DO 2710 J=2,J1
 2710 IF(IN(I,J,K).EQ.6) KCHOP2(K)=KCHOP2(K)+1
      DO 2740 K=1,K0
      WRITE(6,2925) K,KCHOP2(K)
```

```
      DO 2720 I=1,I0                                                    SP36820*
 2720 WRITE(6,2930) I,(IN(I,J,K),J=1,J0)                               SP36830*
 2740 CONTINUE                                                          SP36840*
C                                                                       SP36850*
      WRITE(6,2940)                                                     SP36860*
      DO 2800 K=1,K0                                                    SP36870*
      DO 2800 I=1,I0                                                    SP36880*
      DO 2800 J=1,J0                                                    SP36890*
      N=I+(J-1)*I0+(K-1)*NIJ                                            SP36900*
      IF(IN(I,J,K).NE.4) GO TO 2800                                     SP36910*
      Z=RF*PHI(N)                                                       SP36920*
      WRITE(6,2950) K,I,J,Z,T(N)                                        SP36930*
 2800 CONTINUE                                                          SP36940*
C                                                                       SP36950*
C  PRINT FINAL HEAD VALUES FOR SELECTED NODES                           SP36960*
      IF(IHEDKD.NE.1) GO TO 2825                                        SP36970*
      WRITE(8,2985)                                                     SP36980*
C   UPPER LAYER ONLY                                                    SP36990*
      K=K0                                                              SP37000*
      DO 2820 I=2,I1                                                    SP37010*
      DO 2820 J=2,J1                                                    SP37020*
      IF(IFNLHD(I,J,K).NE.1) GO TO 2820                                 SP37030*
      N=I+(J-1)*I0+(K-1)*NIJ                                            SP37040*
      WRITE(8,2990) I,J,K,PHI(N)                                        SP37050*
 2820 CONTINUE                                                          SP37060*
C   ALL LAYERS                                                          SP37070*
      DO 2823 K=1,K0                                                    SP37080*
      DO 2823 I=2,I1                                                    SP37090*
      DO 2823 J=2,J1                                                    SP37100*
      IF(IFNLHD(I,J,K).NE.2) GO TO 2823                                 SP37110*
      N=I+(J-1)*I0+(K-1)*NIJ                                            SP37120*
      WRITE(8,2990) I,J,K,PHI(N)                                        SP37130*
 2823 CONTINUE                                                          SP37140*
C                                                                       SP37145*
C  PRINT FINAL PHI, T, PERM, AND TK MATRICES                            SP37150*
 2825 IF(KPRNTT.NE.1) GO TO 2895                                        SP37160*
      I0J0=I0*J0                                                        SP37170*
      L1=1                                                              SP37175*
      DO 2893 N1=1,3                                                    SP37180*
      K0A=K0                                                            SP37185*
      IF(N1.GT.2) K0A=K0A-1                                             SP37190*
      DO 2890 K=1,K0A                                                   SP37195*
      WRITE(9,2830)                                                     SP37200*
 2830 FORMAT(9X,'1',9X,'1',9X,'0',9X,'1',9X,'1',9X,'1',9X,'0',9X,'0')  SP37205*
      DO 2885 I=1,I0                                                    SP37210*
      DO 2880 J=1,J0                                                    SP37215*
      N=I+(J-1)*I0+(K-1)*I0J0                                           SP37220*
      GO TO (2835,2840,2845),N1                                         SP37225*
 2835 TDEN(L1)=PHI(N)                                                   SP37230*
      GO TO 2850                                                        SP37235*
 2840 IF(K.LT.K0A) GO TO 2843                                           SP37240*
      M=I+(J-1)*I0                                                      SP37243*
      TDEN(L1)=PERM(M)                                                  SP37246*
      GO TO 2850                                                        SP37250*
 2843 TDEN(L1)=T(N)                                                     SP37253*
      GO TO 2850                                                        SP37256*
 2845 TDEN(L1)=TK(N)                                                    SP37260*
```

```
 2850 IF(L1.LT.8)GO TO 2855                                               SP37263*
      GO TO 2860                                                          SP37266*
 2855 IF(J.EQ.J0)GO TO 2860                                               SP37270*
      L1=L1+1                                                             SP37273*
      GO TO 2880                                                          SP37276*
 2860 WRITE(9,2865)(TDEN(N2), N2=1,8)                                     SP37280*
 2865 FORMAT(8E10.4)                                                      SP37283*
      DO 2870 N2=1,8                                                      SP37286*
      TDEN(N2)=0.0                                                        SP37290*
 2870 CONTINUE                                                            SP37293*
      L1=1                                                                SP37296*
 2880 CONTINUE                                                            SP37300*
 2885 CONTINUE                                                            SP37303*
      IF(N1.NE.2)GO TO 2888                                               SP37306*
      IF(IWATER NE.ICHK(6))GO TO 2888                                     SP37310*
      K0A1=K0A-1                                                          SP37313*
      IF(K.NE.K0A1)GO TO 2888                                             SP37316*
      WRITE(9,2887)                                                       SP37320*
 2887 FORMAT(39X,'1',9X,'1',9X,'1'/'ENDOFMEMBER')                         SP37323*
      GO TO 2890                                                          SP37326*
 2888 IF(N1.EQ.1.AND.K.NE.K0A)GO TO 2890                                  SP37330*
      WRITZ(9,2889)                                                       SP37333*
 2889 FORMAT('ENDOFMEMBER')                                               SP37336*
 2890 CONTINUE                                                            SP37340*
 2893 CONTINUE                                                            SP37345*
 2895 RETURN                                                              SP37350*
 2900 CONTINUE                                                            SP37360*
      IF(TEST.EQ.0.) RETURN                                               SP37370*
      GO TO 60                                                            SP37380
C     ....................................................................SP37390
C                                                                         SP37400
C     ---FORMATS---                                                       SP37410
C                                                                         SP37420
C                                                                         SP37430
C                                                                         SP37440
  220 FORMAT ('0EXCEEDED PERMITTED NUMBER OF ITERATIONS'/' ',39('*'))     SP37450
  230 FORMAT (///1H0,I5,22H ITERATION PARAMETERS:,6E15.7/(/28X,6E15.7/))SP37460
  240 FORMAT ('-',44X,'SOLUTION BY THE STRONGLY IMPLICIT PROCEDURE'/45X,SP37470
     143('_'))                                                            SP37480
 2901 FORMAT(I10,G10.0,2I10)                                              SP37490*
 2902 FORMAT('0','NTAD =',I10,/,3X,'RF =',F10.2,/,1X,'KODCHP =',I10,      SP37500*
     1/1X,'KPRNTT =',I10)                                                 SP37510*
 2903 FORMAT(16F5.0)                                                      SP37520*
 2904 FORMAT('1','BOTTOM OF AQUIFER SYSTEM:'//)                           SP37530*
 2905 FORMAT('0',I4,2X,20F6.1/(7X,20F6.1))                                SP37540*
 2906 FORMAT(80I1)                                                        SP37550*
 2907 FORMAT('1','SPECIFIED-HEAD OCEAN NODES'//5X,'NUMBERS REPRESENT LAYSP37560*
     1ER CONTAINING THE SPECIFIED HEAD OCEAN NODE'/7X,'0 = NO OCEAN NODESP37563*
     2 AT THIS LOCATION'///)                                              SP37566*
 2908 FORMAT('0',I4,2X,60(I1,1X)/(7X,60(I1,1X)))                          SP37570*
 2909 FORMAT(8G10.0)                                                      SP37580*
 2910 FORMAT('0',5X,'DEPTH TO TOP OF LAYER',I5,'  =  ',F6.1)              SP37590*
 2911 FORMAT('1','PRELIMINARY IN ARRAY FOR LAYER',I5//'   INITIAL TRANSMSP37600*
     1ISSIVITY SET EQUAL TO ZERO ALONG PERIMETER'/'     AND WHERE CODE =SP37610*
     2 8.'///)                                                            SP37615*
 2912 FORMAT('0',I4,2X,60I2/(7X,60I2))                                    SP37620*
 2920 FORMAT('0','T ADJUSTED ON ITERATION ',I4)                           SP37630*
```

```
2925 FORMAT('1','IN ARRAY FOR LAYER',I3,5X,I5,' BLOCKS CHOPPED'//'   8=SP37640*
    1 INITIAL TRANSMISSIVITY EQUALS ZERO '/'   6= BLOCK DROPPED DURING SP37643*
    2CHOP'/'   7= BLOCK DROPPED BECAUSE OF OCEAN LEAKAGE INTO AQUIFER'/SP37646*
    3'   4= BLOCK WITH ADJUSTED TRANSMISSIVITY'///)                    SP37648*
2930 FORMAT('0',61I2/(3X,60I2))                                         SP37650*
2940 FORMAT('1','SUMMARY OF NODES WITH ADJUSTED TRANSMISSIVITY'/'    LASP37660*
    1YER       ROW      COLUMN        Z        T(I,J,K)')               SP37670*
2950 FORMAT('0',5X,I2,2(8X,I2),6X,F6.2,6X,F8.4)                         SP37680*
2960 FORMAT(10(I3,I3,I2))                                               SP37690*
2970 FORMAT('1','HEAD VALUES AT SELECTED NODES AFTER EACH ITERATION'//1SP37700*
    14X,'NODE(I,J,K):',5X,10('(',I2,',',I2,',',I1,')'))                 SP37705*
2980 FORMAT('0',' HEAD AT ITERATION',2X,I5,5X,10(2X,F6.2,2X))           SP37710*
2985 FORMAT('1',10X,'FINAL HEAD VALUES FOR SELECTED NODES')             SP37720*
2990 FORMAT(' ',3I5,5X,F7.2)                                            SP37730*
8100 FORMAT(' ','I=',I3,2X,'J=',I3,2X,'K=',I3,2X,'DL=',E10.4,' E=',     SP37740*
    1E10.4,' W=',E10.4,' CL=',E10.4,' BL=',E10.4,' AL=',E10.4,'SIPNORM'SP37750*
    2)                                                                  SP37760*
8125 FORMAT(' ','B,D,F,H,SU,Z,RHO,UXR=',8(E10.4,2X))                    SP37770*
8130 FORMAT(' ','A,C,G,WU,TU,U=',6(E10.4,2X))                           SP37780*
8150 FORMAT(' ',5X,'T(N)=',E10.4,' T(NIB)=',E10.4,' T(NIA)=',E10.4,     SP37790*
    1' T(NJB)=',E10.4,' T(NJA)=',E10.4,' T(NKB)=',E10.4,' T(NKA)='      SP37800*
    2,E10.4,/)                                                          SP37810*
8200 FORMAT(' ','I=',I3,2X,'J=',I3,2X,'K=',I3,2X,'DL=',E10.4,' E=',     SP37820*
    1E10.4,' W=',E10.4,' CL=',E10.4,' BL=',E10.4,' AL=',E10.4,'SIPREV')SP37830*
8300 FORMAT('1',' I ',' J ','  T(I,J,1)   ','  T(I,J,2)   ','  T(I,J,3)  SP37840*
    1','  T(I,J,4)   ','  T(I,J,5)   ')                                 SP37850*
8400 FORMAT(' ',2I3,5E12.4)                                             SP37860*
8500 FORMAT('1',' I ',' J ','  TK(I,J,1) ','  TK(I,J,2) ','  TK(I,J,3) SP37870*
    1','  TK(I,J,4)')                                                   SP37880*
8600 FORMAT(' ',2I3,4E12.4)                                             SP37890*
     END                                                                SP37900
```

```
      SUBROUTINE COEF(PHI,STRT,OLD,T,S,TR,TC,TK,WELL,DELX,DELY,DELZ,FACT
     1,PERM,BOTTOM,QRE)
C     ------------------------------------------------------------------
C     COMPUTE COEFFICIENTS
C     ------------------------------------------------------------------
C
C     SPECIFICATIONS:
      REAL *8PHI
C
      DIMENSION PHI(I0,J0,K0), STRT(I0,J0,K0), OLD(I0,J0,K0), T(I0,J0,K0
     1), S(I0,J0,K0), TR(I0,J0,K0), TC(I0,J0,K0), TK(IK,JK,K5), WELL(I0,
     2J0,K0), DELX(J0), DELY(I0), DELZ(K0), FACT(K0,3), PERM(IP,JP), BOT
     3TOM(IP,JP), QRE(IQ,JQ)
C
      COMMON /INTEGR/ I0,J0,K0,I1,J1,K1,I,J,K,NPER,KTH,ITMAX,LENGTH,KP,N
     1WEL,NUMT,IFINAL,IT,KT,IHEAD,IDRAW,IFLO,IERR,I2,J2,K2,IMAX,ITMX1,NC
     2H,IDK1,IDK2,IWATER,IQRE,IP,JP,IQ,JQ,IK,JK,K5,IPU1,IPU2,ITK,IEQN
     3,NKODE,KFLOW                                                         *
      COMMON /SPARAM/ TMAX,CDLT,DELT,ERR,TEST,SUM,SUMP,QR,BETA1            *
      COMMON /SARRAY/ ICHK(13),LEVEL1(9),LEVEL2(9)
      N3 = 1
      N7=-1
      RETURN
C     ..................................................................
C     ---COMPUTE TRANSMISSIVITY FOR UPPER HYDROLOGIC UNIT WHEN
C     IT IS UNCONFINED---
C     *********************
      ENTRY TRANS(N3)
C     *********************
      DO 10 I=2,I1
      DO 10 J=2,J1
      IF (PERM(I,J).EQ.0.) GO TO 10
      T(I,J,K0)=PERM(I,J)*(PHI(I,J,K0)-BOTTOM(I,J))
      IF (T(I,J,K0).GT.0.) GO TO 10
      IF (WELL(I,J,K0).LT.0.) WRITE (6,60) I,J,K0
      IF (WELL(I,J,K0).GE.0.) WRITE (6,70) I,J,K0
      PERM(I,J)=0.
      T(I,J,K0)=0.
      TR(I,J-1,K0)=0.
      TR(I,J,K0)=0.
      TC(I,J,K0)=0.
      TC(I-1,J,K0)=0.
      IF (K0.NE.1) TK(I,J,K1)=0.
      PHI(I,J,K0)=1.D30
   10 CONTINUE
      IF (N3.EQ.1) RETURN
      N1=K0
      N2=K0
      N4=K1
      GO TO 20
C     ---COMPUTE T COEFFICIENTS---
C     *********************
      ENTRY TCOF
C     *********************
      N7=N7+1
      N1=1
      N2=K0
```

```
      IF(N7.GT.0) N2=K1                                                 COF0580
      N4=1                                                              COF0590
   20 DO 40 K=N1,N2                                                     COF0600
      DO 40 I=1,I1                                                      COF0610
      DO 40 J=1,J1                                                      COF0620
      IF (T(I,J,K).EQ.0.) GO TO 32                                      COF0630
      IF (T(I,J+1,K).EQ.0.) GO TO 30                                    COF0640
      TR(I,J,K)=(2.*T(I,J+1,K)*T(I,J,K))/(T(I,J,K)*DELX(J+1)+T(I,J+1,K)*COF0650
     1DELX(J))*FACT(K,1)                                                COF0660
   30 IF (T(I+1,J,K).EQ.0.) GO TO 32                                    COF0670
      TC(I,J,K)=(2.*T(I+1,J,K)*T(I,J,K))/(T(I,J,K)*DELY(I+1)+T(I+1,J,K)*COF0680
     1DELY(I))*FACT(K,2)                                                COF0690
      IF(T(I,J+1,K).EQ.0.) GO TO 32                                     COF0700
      GO TO 40                                                          COF0710
   32 IF(T(I,J,K).EQ.0.) GO TO 33                                       COF0720
      GO TO 34                                                          COF0730
   33 TR(I,J,K)=0.                                                      COF0740
      TR(I,J-1,K)=0.                                                    COF0750
      TC(I,J,K)=0.                                                      COF0760
      TC(I-1,J,K)=0.                                                    COF0770
      TK(I,J,K)=0.                                                      COF0780
      IF(WELL(I,J,K).LT.0.)GO TO 80                                     COF0785*
   34 IF(T(I+1,J,K).EQ.0.) GO TO 35                                     COF0790
      GO TO 36                                                          COF0800
   35 TC(I,J,K)=0.                                                      COF0810
      TC(I+1,J,K)=0.                                                    COF0820
      TK(I+1,J,K)=0.                                                    COF0830
   36 IF(T(I,J+1,K).EQ.0.) GO TO 37                                     COF0840
      GO TO 40                                                          COF0850
   37 TR(I,J,K)=0.                                                      COF0860
      TR(I,J+1,K)=0.                                                    COF0870
      TK(I,J+1,K)=0.                                                    COF0880
   40 CONTINUE                                                          COF0890
      IF (K0.EQ.1.OR.ITK.EQ.ICHK(10).OR.N3.EQ.0) RETURN                 COF0900
      DO 50 K=N4,K1                                                     COF0910
      DO 50 I=2,I1                                                      COF0920
      DO 50 J=2,J1                                                      COF0930
      IF (T(I,J,K+1).EQ.0.) GO TO 50                                    COF0940
      T1=T(I,J,K)*FACT(K,3)                                             COF0950
      T2=T(I,J,K+1)*FACT(K+1,3)                                         COF0960
      TK(I,J,K)=(2.*T2*T1)/(T1*DELZ(K+1)+T2*DELZ(K))                    COF0970
   50 CONTINUE                                                          COF0980
      GO TO 100                                                         COF0981
   80 WRITE(6,73) I,J,K                                                 COF0982*
      WRITE(6,75)                                                       COF0983*
      IERR=2                                                            COF0984
      CALL OUTPJT                                                       COF0985
  100 RETURN                                                            COF0990
C                                                                       COF1000
C                                                                       COF1010
   60 FORMAT ('-',20('*'),'WELL',2I3,' IN LAYER',I3,' ,GOES DRY',20('*'))COF1020
   70 FORMAT ('-',20('*'),'NODE',2I3,' IN LAYER',I3,' GOES DRY',20('*'))COF1030
   73 FORMAT('-',20('*'),'   WELL',2I3,' IN LAYER',I3,' GOES DRY--DEWATECOF1031*
     1RED OR SALINE-WATER INTRUSION   ',20('*'))                        COF1032*
   75 FORMAT('-',5X,'*** EXECUTION TERMINATING-DISCHARGING WELL NODE (NOCOF1033*
     1T IN TOP LAYER) GOES DRY--CHECK FOR SW INTRUSION ***'/10X,'THE FOLCOF1034*
     2LOWING OUTPUT REPRESENTS STATE OF SOLUTION WHEN WELL WENT DRY')   COF1035*
      END                                                               COF 1040
```

```
      SUBROUTINE CHECKI(PHI,STRT,OLD,T,S,TR,TC,TK,WELL,DELX,DELY,DELZ,FACHK0010
     1CT,JFLO,FLOW,QRE,                                                 CHK0020
     2IDR,RH,RC,RB,IRIV)                                                CHK0030*
C     ------------------------------------------------------------------CHK0040
C     COMPUTE A VOLUMETRIC BALANCE                                      CHK0050
C     ------------------------------------------------------------------CHK0060
C                                                                       CHK0070
C     SPECIFICATIONS:                                                   CHK0080
      REAL *8PHI                                                        CHK0090
C                                                                       CHK0100
      DIMENSION PHI(I0,J0,K0), STRT(I0,J0,K0), OLD(I0,J0,K0), T(I0,J0,K0CHK0110
     1), S(I0,J0,K0), TR(I0,J0,K0), TC(I0,J0,K0), TK(IK,JK,K5), WELL(I0,CHK0120
     2J0,K0), DELX(J0), DELY(I0), DELZ(K0), FACT(K0,3), JFLO(NCH,3), FLOCHK0130
     3W(NCH), QRE(IQ,JQ)                                                CHK0140
     4,IDR(I0,J0,K0),RH(1),RC(1),RB(1)                                  CHK0150*
C                                                                       CHK0160
      COMMON /INTEGR/ I0,J0,K0,I1,J1,K1,I,J,K,NPER,KTH,ITMAX,LENGTH,KP,NCHK0170
     1WEL,NUMT,IFINAL,IT,KT,IHEAD,IDRAW,IFLO,IERR,I2,J2,K2,IMAX,ITMX1,NCCHK0180
     2H,IDK1,IDK2,IWATER,IQRE,IP,JP,IQ,JQ,IK,JK,K5,IPU1,IPU2,ITK,IEQN   CHK0190
     3,NKODE,KFLOW                                                      CHK0200*
      COMMON /SPARAM/ TMAX,CDLT,DELT,ERR,TEST,SUM,SUMP,QR,BETA1         CHK0210*
      COMMON /SARRAY/ ICHK(13),LEVEL1(9),LEVEL2(9)                      CHK0220
      COMMON /CK/ RLFLXT,STORT,QRET,CHST,CHDT,FLUXT,PUMPT,CFLUXT,FLXNT  CHK0230*
      RETURN                                                            CHK0240
C     ..................................................................CHK0250
C     ********************                                              CHK0260
      ENTRY CHECK                                                       CHK0270
C     ********************                                              CHK0280
C     ---INITIALIZE VARIABLES---                                        CHK0290
      PUMP=0.                                                           CHK0300
      STOR=0.                                                           CHK0310
      FLUXS=0.0                                                         CHK0320
      CHD1=0.0                                                          CHK0330
      CHD2=0.0                                                          CHK0340
      QREFLX=0.                                                         CHK0350
      CFLUX=0.                                                          CHK0360
      FLUX=0.                                                           CHK0370
      RLFLUX=0.                                                         CHK0380*
      FLXN=0.0                                                          CHK0390
      II=0                                                              CHK0400
C     ..................................................................CHK0410
      IF(MOD(KT,KTH).EQ.0.OR.IFINAL.EQ.1) WRITE(6,5900)                 CHK0425
C                                                                       CHK0420
C     ---COMPUTE RATES,STORAGE AND PUMPAGE FOR THIS STEP---             CHK0430
      DO 220 K=1,K0                                                     CHK0440
      DO 220 I=2,I1                                                     CHK0450
      DO 220 J=2,J1                                                     CHK0460
      INDEX=0                                                           CHK0470
      IF (T(I,J,K).EQ.0.) GO TO 220                                     CHK0480
      AREA=DELX(J)*DELY(I)                                              CHK0490
      VOLUME=AREA*DELZ(K)                                               CHK0500
      IF (S(I,J,K).GE.0.) GO TO 180                                     CHK0510
C                                                                       CHK0520
C     ---COMPUTE FLOW RATES TO AND FROM CONSTANT HEAD BOUNDARIES---     CHK0530
      II=II+1                                                           CHK0540
      FLOW(II)=0.                                                       CHK0550
      JFLO(II,1)=K                                                      CHK0560
```

```
      JFLO(II,2)=I                                                      CHK0570
      JFLO(II,3)=J                                                      CHK0580
      IF (S(I,J-1,K).LT.0..OR.T(I,J-1,K).EQ.0.) GO TO 30                CHK0590
      X=(PHI(I,J,K)-PHI(I,J-1,K))*TR(I,J-1,K)*DELY(I)                   CHK0600
      IF(IEQN.EQ.ICHK(11)) X=X*DELZ(K)                                  CHK0610
      FLOW(II)=FLOW(II)+X                                               CHK0620
      IF(KFLOW.NE.1) GO TO  5                                           CHK0630*
      IF(ABS(X).LT.0.00001) GO TO 5                                     CHK0640*
      M=J-1                                                             CHK0650*
      WRITE(6,6000) I,J,K,I,M,K,X                                       CHK0660*
      INDEX=1                                                           CHK0670*
    5 IF (X) 10,30,20                                                   CHK0690
   10 CHD1=CHD1+X                                                       CHK0700
      GO TO 30                                                          CHK0710
   20 CHD2=CHD2+X                                                       CHK0720
   30 IF (S(I,J+1,K).LT.0..OR.T(I,J+1,K).EQ.0.) GO TO 60                CHK0730
      X=(PHI(I,J,K)-PHI(I,J+1,K))*DELY(I)*TR(I,J,K)                     CHK0740
      IF(IEQN.EQ.ICHK(11)) X=X*DELZ(K)                                  CHK0750
      FLOW(II)=FLOW(II)+X                                               CHK0760
      IF(KFLOW.NE.1) GO TO 35                                           CHK0770*
      IF(ABS(X).LT.0.00001) GO TO 35                                    CHK0780*
      M=J+1                                                             CHK0790*
      IF(INDEX.EQ.0) WRITE(6,6000) I,J,K,I,M,K,X                        CHK0800*
      INDEX=1                                                           CHK0810*
      WRITE(6,6100) I,M,K,X                                             CHK0820*
   35 IF (X) 40,60,50                                                   CHK0830
   40 CHD1=CHD1+X                                                       CHK0840
      GO TO 60                                                          CHK0850
   50 CHD2=CHD2+X                                                       CHK0860
   60 IF (K.EQ.1) GO TO 90                                              CHK0870
      IF (S(I,J,K-1).LT.0..OR.T(I,J,K-1).EQ.0.) GO TO 90                CHK0880
      X=(PHI(I,J,K)-PHI(I,J,K-1))*TK(I,J,K-1)*AREA                      CHK0890
      FLOW(II)=FLOW(II)+X                                               CHK0900
      IF(KFLOW.NE.1) GO TO 65                                           CHK0910*
      IF(ABS(X).LT.0.00001) GO TO 65                                    CHK0920*
      M=K-1                                                             CHK0930*
      IF(INDEX.EQ.0) WRITE(6,6000) I,J,K,I,J,M,X                        CHK0940*
      INDEX=1                                                           CHK0950*
      WRITE(6,6100) I,J,M,X                                             CHK0960*
   65 IF (X) 70,90,80                                                   CHK0970
   70 CHD1=CHD1+X                                                       CHK0980
      GO TO 90                                                          CHK0990
   80 CHD2=CHD2+X                                                       CHK1000
   90 IF (K.EQ.KO) GO TO 120                                            CHK1010
      IF (S(I,J,K+1).LT.0..OR.T(I,J,K+1).EQ.0.) GO TO 120               CHK1020
      X=(PHI(I,J,K)-PHI(I,J,K+1))*TK(I,J,K)*AREA                        CHK1030
      FLOW(II)=FLOW(II)+X                                               CHK1040
      IF(KFLOW.NE.1) GO TO 95                                           CHK1050*
      IF(ABS(X).LT.0.00001) GO TO 95                                    CHK1060*
      M=K+1                                                             CHK1070*
      IF(INDEX.EQ.0) WRITE(6,6000) I,J,K,I,J,M,X                        CHK1080*
      INDEX=1                                                           CHK1090*
      WRITE(6,6100) I,J,M,X                                             CHK1100*
   95 IF (X) 100,120,110                                                CHK1110
  100 CHD1=CHD1+X                                                       CHK1120
      GO TO 120                                                         CHK1130
  110 CHD2=CHD2+X                                                       CHK1140
```

```
  120 IF (S(I-1,J,K).LT.0..OR.T(I-1,J,K).EQ.0.) GO TO 150                    CHK1150
      X=(PHI(I,J,K)-PHI(I-1,J,K))*TC(I-1,J,K)*DELX(J)                         CHK1160
      IF(IEQN.EQ.ICHK(11)) X=X*DELZ(K)                                        CHK1170
      FLOW(II)=FLOW(II)+X                                                     CHK1180
      IF(KFLOW.NE.1) GO TO 125                                                CHK1190
      IF(ABS(X).LT.0.00001) GO TO 125                                         CHK1200*
      M=I-1                                                                   CHK1210*
      IF(INDEX.EQ.0) WRITE(6,6000) I,J,K,M,J,K,X                              CHK1220*
      INDEX=1                                                                 CHK1230*
      WRITE(6,6100) M,J,K,X                                                   CHK1240*
  125 IF (X) 130,150,140                                                      CHK1250
  130 CHD1=CHD1+X                                                             CHK1260
      GO TO 150                                                               CHK1270
  140 CHD2=CHD2+X                                                             CHK1280
  150 IF (S(I+1,J,K).LT.0..OR.T(I+1,J,K).EQ.0.) GO TO 220                    CHK1290
      X=(PHI(I,J,K)-PHI(I+1,J,K))*TC(I,J,K)*DELX(J)                           CHK1300
      IF(IEQN.EQ.ICHK(11)) X=X*DELZ(K)                                        CHK1310
      FLOW(II)=FLOW(II)+X                                                     CHK1320
      IF(KFLOW.NE.1) GO TO 155                                                CHK1330*
      IF(ABS(X).LT.0.00001) GO TO 155                                         CHK1340*
      M=I+1                                                                   CHK1350*
      IF(INDEX.EQ.0) WRITE(6,6000) I,J,K,M,J,K,X                              CHK1360*
      INDEX=1                                                                 CHK1370*
      WRITE(6,6100) M,J,K,X                                                   CHK1380*
  155 IF (X) 160,220,170                                                      CHK1390
  160 CHD1=CHD1+X                                                             CHK1400
      GO TO 220                                                               CHK1410
  170 CHD2=CHD2+X                                                             CHK1420
      GO TO 220                                                               CHK1430
C                                                                             CHK1440
C      ---CHECK FOR EQUATION BEING SOLVED---                                  CHK1450
  180 IF(IEQN.EQ.ICHK(11)) GO TO 211                                          CHK1460
C                                                                             CHK1470
C     ---EQUATION 4---                                                        CHK1480
C     ---RECHARGE AND WELLS---                                                CHK1490
      IF (K.EQ.KO.AND.IQRE.EQ.ICHK(7)) QREFLX=QREFLX+QRE(I,J)*AREA            CHK1500
      IF (WELL(I,J,K)) 190,210,200                                            CHK1510
  190 PUMP=PUMP+WELL(I,J,K)*AREA                                              CHK1520
      GO TO 210                                                               CHK1530
  200 CFLUX=CFLUX+WELL(I,J,K)*AREA                                            CHK1540
C                                                                             CHK1550
C     ---COMPUTE VOLUME FROM STORAGE---                                       CHK1560
  210 STOR=STOR+S(I,J,K)*(OLD(I,J,K)-PHI(I,J,K))*AREA                         CHK1570
      IF(IRIV.LE.0) GO TO 220                                                 CHK1580*
C                                                                             CHK1585*
C     ---COMPUTE 'RIVER' LEAKAGE---                                           CHK1590*
C                                                                             CHK1595*
      ND=IDR(I,J,K)                                                           CHK1600*
      IF(ND.EQ.0) GO TO 220                                                   CHK1610*
      IF(PHI(I,J,K).GT.RB(ND)) GO TO 217                                      CHK1620*
      FLXRAT=RC(ND)*(RH(ND)-RB(ND))*AREA                                      CHK1630*
      IF(KFLOW.EQ.1) WRITE(6,6200) I,J,K,FLXRAT                               CHK1640*
      RLFLUX=RLFLUX+FLXRAT                                                    CHK1650*
      GO TO 220                                                               CHK1660*
  217 FLXRAT=RC(ND)*(RH(ND)-PHI(I,J,K))*AREA                                  CHK1670*
      IF(KFLOW.EQ.1) WRITE(6,6200) I,J,K,FLXRAT                               CHK1680*
      RLFLUX=RLFLUX+FLXRAT                                                    CHK1690*
```

```
      GO TO 220
C
C     ---EQUATION 3---
C     ---RECHARGE AND WELLS---
  211 IF (K.EQ.K0.AND.IQRE.EQ.ICHK(7)) QREFLX=QREFLX+QRE(I,J)*VOLUME
      IF(WELL(I,J,K)) 212,214,213
  212 PUMP=PUMP+WELL(I,J,K)*VOLUME
      GO TO 214
  213 CFLUX=CFLUX+WELL(I,J,K)*VOLUME
C
C     ---COMPUTE VOLUME FROM STORAGE---
  214 STOR=STOR+S(I,J,K)*(OLD(I,J,K)-PHI(I,J,K))*VOLUME
  220 CONTINUE
C     ..................................................................
C
C     ---COMPUTE CUMULATIVE VOLUMES, TOTALS, AND DIFFERENCES---
      FLXPT=0.0
      RLFLXT=RLFLXT-RLFLUX*DELT
      STORT=STORT+STOR
      STOR=STOR/DELT
      QRET=QRET+QREFLX*DELT
      CHDT=CHDT-CHD1*DELT
      CHST=CHST+CHD2*DELT
      PUMPT=PUMPT-PUMP*DELT
      CFLUXT=CFLUXT+CFLUX*DELT
      TOTL1=STORT+QRET+CFLUXT+CHST+FLXPT
      TOTL2=CHDT+PUMPT+RLFLXT+FLXNT
      SUMR=QREFLX+CFLUX+CHD2+CHD1+PUMP+RLFLUX+FLUXS+STOR
      DIFF=TOTL2-TOTL1
      PERCNT=0.0
      IF (TOTL2.EQ.0.) GO TO 230
      PERCNT=DIFF/TOTL2*100.
  230 RETURN
C     ..................................................................
C
C     ---PRINT RESULTS---
C     ************************
      ENTRY CWRITE
C     ************************
C
      WRITE (6,260) STOR,QREFLX,STORT,CFLUX,QRET,PUMP,CFLUXT,RLFLUX,CHST
     1,FLXPT,CHD2,TOTL1,CHD1,FLUX,FLUXS,RLFLXT,CHDT,SUMR,PUMPT,FLXNT,TOT
     2L2,DIFF,PERCNT
      IF (NCH.EQ.0) GO TO 240
      WRITE (6,270)
      WRITE (6,280) ((JFLO(I,J),J=1,3),FLOW(I),I=1,NCH)
C
C     ---COMPUTE VERTICAL FLOW---
  240 X=0.
      Y=0.
      IF (K0.EQ.1) RETURN
      DO 250 I=2,I1
      DO 250 J=2,J1
      X=X+(PHI(I,J,1)-PHI(I,J,2))*TK(I,J,1)*DELX(J)*DELY(I)
  250 Y=Y+(PHI(I,J,K1)-PHI(I,J,K0))*TK(I,J,K1)*DELX(J)*DELY(I)
      WRITE (6,290) Y,X
      RETURN
```

```
C                                                                       CHK2270
C     ---FORMATS---                                                     CHK2280
C                                                                       CHK2290
C     ------------------------------------------------------------------CHK2300
C                                                                       CHK2310
C                                                                       CHK2320
C                                                                       CHK2330
  260 FORMAT ('0',10X,'CUMULATIVE MASS BALANCE:',16X,'L**3',23X,'RATES FCHK2340
     1OR THIS TIME STEP:',16X,'L**3/T'/11X,24('-'),43X,25('-')//20X,'SOUCHK2350
     2RCES:',69X,'STORAGE =',F20.4/20X,8('-'),68X,'RECHARGE =',F20.4/27XCHK2360
     3,'STORAGE =',F20.2,35X,'CONSTANT FLUX =',F20.4/26X,'RECHARGE =',F2CHK2370
     40.2,41X,'PUMPING =',F20.4/21X,'CONSTANT FLUX =',F20.2,30X,'     RICHK2380*
     5VER LEAKAGE =',F20.4/21X,'CONSTANT HEAD =',F20.2,34X,'CONSTANT HEACHK2390*
     6D:'/27X,'LEAKAGE =',F20.2,46X,'IN =',F20.4/21X,'TOTAL SOURCES =',FCHK2400
     720.2,45X,'OUT =',F20.4/96X,'LEAKAGE:'/20X,'DISCHARGES:',45X,'FROM CHK2410
     8PREVIOUS PUMPING PERIOD =',F20.4/20X,11('-'),68X,'TOTAL =',F20.4/1CHK2420
     96X,'     RIVER LEAKAGE =',F20.2/21X,'CONSTANT HEAD =',F20.2,36X,'SCHK2430*
     $UM OF RATES =',F20.4/19X'QUANTITY PUMPED =',F20.2/27X,'LEAKAGE =',CHK2440
     $F20.2/19X,'TOTAL DISCHARGE =',F20.2//17X,'DISCHARGE-SOURCES =',F20CHK2450
     $.2/15X,'PER CENT DIFFERENCE =',F20.2//)                           CHK2460
  270 FORMAT ('0FLOW RATES TO CONSTANT HEAD NODES:'/' ',34('-')//' ',3(9CHK2470
     1X,'K',4X,'I',4X,'J',5X,'RATE (L**3/T)')/' ',3(9X,'-',4X,'-',4X,'-'CHK2480
     2,5X,13('-'))/)                                                    CHK2490
  280 FORMAT (/'1X,3(I10,2I5,G18.7)))                                   CHK2500
  290 FORMAT ('0FLOW TO TOP LAYER =',G15.7,'   FLOW TO BOTTOM LAYER =',GCHK2510
     115.7,'       POSITIVE UPWARD')                                    CHK2520
 5900 FORMAT('1','   I    J    K   FROM    I    J    K     CONSTANT HEAD RATE CHK2530*
     1        RIVER LEAKAGE'//)                                         CHK2540*
 6000 FORMAT(' ',3I4,7X,3I4,8X,G12.4)                                   CHK2550*
 6100 FORMAT(' ',19X,3I4,8X,G12.4)                                      CHK2560*
 6200 FORMAT(' ',3I4,48X,G12.4)                                         CHK2565*
      END                                                               CHK2570
```

```
      SUBROUTINE PRNTAI(PHI,STRT,T,S,WELL,DELX,DELY)
C     -----------------------------------------------------------------
C     PRINT MAPS OF DRAWDOWN AND HYDRAULIC HEAD
C     -----------------------------------------------------------------
C
C     SPECIFICATIONS:
      REAL *8PHI,Z,XLABEL,YLABEL,TITLE,XN1,MESUR
      REAL *4K
C
      DIMENSION PHI(I0,J0,K0), STRT(I0,J0,K0), S(I0,J0,K0), WELL(I0,J0,K
     1O), DELX(J0), DELY(I0), T(I0,J0,K0)
C
      COMMON /INTEGR/ I0,J0,K0,I1,J1,K1,I,J,K,NPER,KTH,ITMAX,LENGTH,KP,N
     1WEL,NUMT,IFINAL,IT,KT,IHEAD,IDRAW,IFLO,IERR,I2,J2,K2,IMAX,ITMX1,NC
     2H,IDK1,IDK2,IWATER,IQRE,IP,JP,IQ,JQ,IK,JK,K5,IPU1,IPU2,ITK,IEQN
     3,NKODE,KFLOW
      COMMON /PR/ XLABEL(3),YLABEL(6),TITLE(6),XN1,MESUR,PRNT(122),BLANK
     1(60),DIGIT(122),VF1(6),VF2(6),VF3(7),XSCALE,DINCH,SYM(17),XN(100),
     2YN(13),NA(4),N1,N2,N3,YSCALE,FACT1,FACT2
      RETURN
C     .................................................................
C
C     ---INITIALIZE VARIABLES FOR PLOT---
C     ********************
      ENTRY MAP
C     ********************
      YDIM=0.
      WIDTH=0.
      DO 10 J=2,J1
   10 WIDTH=WIDTH+DELX(J)
      DO 20 I=2,I1
   20 YDIM=YDIM+DELY(I)
   30 XSF=DINCH*XSCALE
      YSF=DINCH*YSCALE
      NYD=YDIM/YSF
      IF (NYD*YSF.LE.YDIM-DELY(I1)/2.) NYD=NYD+1
      IF (NYD.LE.12) GO TO 40
      DINCH=YDIM/(12.*YSCALE)
      WRITE (6,330) DINCH
      IF (YSCALE.LT.1.0) WRITE (6,340)
      GO TO 30
   40 NXD=WIDTH/XSF
      IF (NXD*XSF.LE.WIDTH-DELX(J1)/2.) NXD=NXD+1
      N4=NXD*N1+1
      N5=NXD+1
      N6=NYD+1
      N8=N2*NYD+1
      NA(1)=N4/2-1
      NA(2)=N4/2
      NA(3)=N4/2+3
      NC=(N3-N8-10)/2
      ND=NC+N8
      NE=MAX0(N5,N6)
      VF1(3)=DIGIT(ND)
      VF2(3)=DIGIT(ND)
      VF3(3)=DIGIT(NC)
      XLABEL(3)=MESUR
```

```
      YLABEL(6)=MESUR
      DO 60 I=1,NE
      NNX=N5-I
      NNY=I-1
      IF (NNY.GE.N6) GO TO 50
      YN(I)=YSF*NNY/YSCALE
   50 IF (NNX.LT.0) GO TO 60
      XN(I)=XSF*NNX/YSCALE
   60 CONTINUE
      RETURN
C     ...................................................................
C
C     ********************
      ENTRY PRNTA(NG,LA)
C     ********************
C     ---VARIABLES INITIALIZED EACH TIME A PLOT IS REQUESTED---
      DIST=WIDTH-DELX(J1)/2.
      JJ=J1
      LL=1
      Z=NXD*XSF
      IF (NG.EQ.1) WRITE (6,300) (TITLE(I),I=1,3),LA
      IF (NG.EQ.2) WRITE (6,300) (TITLE(I),I=4,6),LA
      DO 290 I=1,N4
C
C     ---LOCATE X AXES---
      IF (I.EQ.1.OR.I.EQ.N4) GO TO 70
      PRNT(1)=SYM(12)
      PRNT(N8)=SYM(12)
      IF ((I-1)/N1*N1.NE.I-1) GO TO 90
      PRNT(1)=SYM(14)
      PRNT(N8)=SYM(14)
      GO TO 90
C
C     ---LOCATE Y AXES---
   70 DO 80 J=1,N8
      IF ((J-1)/N2*N2.EQ.J-1) PRNT(J)=SYM(14)
   80 IF ((J-1)/N2*N2.NE.J-1) PRNT(J)=SYM(13)
C
C     ---COMPUTE LOCATION OF NODES AND DETERMINE APPROPRIATE SYMBOL---
   90 IF (DIST.LT.0..OR.DIST.LT.Z-XN1*XSF) GO TO 240
      YLEN=DELY(2)/2.
      DO 220 L=2,I1
      J=YLEN*N2/YSF+1.5
      IF (T(L,JJ,LA).EQ.0.) GO TO 160
      IF (S(L,JJ,LA).LT.0.) GO TO 210
      INDX3=0
      GO TO (100,110), NG
  100 K=(STRT(L,JJ,LA)-PHI(L,JJ,LA))*FACT1
C     -TO CYCLE SYMBOLS FOR DRAWDOWN, REMOVE C FROM COL. 1 OF NEXT CARD-
C     K=AMOD(K,10.)
      GO TO 120
  110 K=PHI(L,JJ,LA)*FACT2
  120 IF (K) 130,160,140
  130 IF (J-2.GT.0) PRNT(J-2)=SYM(13)
      N=-K+.5
      IF (N.LT.100) GO TO 150
      GO TO 190
```

```
  140 N=K+.5
      IF (N.LT.100) GO TO 150
      IF (N.GT:999) GO TO 190
      INDX3=N/100
      IF (J-2.GT.0) PRNT(J-2)=SYM(INDX3)
      N=N-INDX3*100
  150 INDX1=MOD(N,10)
      IF (INDX1.EQ.0) INDX1=10
C     -TO CYCLE SYMBOLS FOR DRAWDOWN, REMOVE C FROM COL. 1 OF NEXT CARD-
C     IF (NG.EQ.1) GO TO 170
      INDX2=N/10
      IF (INDX2.GT.0) GO TO 180
      INDX2=10
      IF (INDX3.EQ.0) INDX2=15
      GO TO 180
  160 INDX1=15
  170 INDX2=15
  180 IF (J-1.GT.0) PRNT(J-1)=SYM(INDX2)
      PRNT(J)=SYM(INDX1)
      GO TO 220
  190 DO 200 II=1,3
      JI=J-3+II
  200 IF (JI.GT.0) PRNT(JI)=SYM(11)
  210 IF (S(L,JJ,LA).LT.0.) PRNT(J)=SYM(16)
  220 YLEN=YLEN+(DELY(L)+DELY(L+1))/2.
  230 DIST=DIST-(DELX(JJ)+DELX(JJ-1))/2.
      JJ=JJ-1
      IF (JJ.EQ.0) GO TO 240
      IF (DIST.GT.Z-XN1*XSF) GO TO 230
  240 CONTINUE
C
C     ---PRINT AXES,LABELS, AND SYMBOLS---
      IF (I-NA(LL).EQ.0) GO TO 260
      IF ((I-1)/N1*N1-(I-1)) 270,250,270
  250 WRITE (6,VF1) (BLANK(J),J=1,NC),(PRNT(J),J=1,N8),XN(1+(I-1)/6)
      GO TO 280
  260 WRITE (6,VF2) (BLANK(J),J=1,NC),(PRNT(J),J=1,N8),XLABEL(LL)
      LL=LL+1
      GO TO 280
  270 WRITE (6,VF2) (BLANK(J),J=1,NC),(PRNT(J),J=1,N8)
C
C     ---COMPUTE NEW VALUE FOR Z AND INITIALIZE PRNT---
  280 Z=Z-2.*XN1*XSF
      DO 290 J=1,N8
  290 PRNT(J)=SYM(15)
C
C     ---NUMBER AND LABEL Y AXIS AND PRINT LEGEND---
      WRITE (6,VF3) (BLANK(J),J=1,NC),(YN(I),I=1,N6)
      WRITE (6,320) (YLABEL(I),I=1,6)
      IF (NG.EQ.1) WRITE (6,310) FACT1
      IF (NG.EQ.2) WRITE (6,310) FACT2
      RETURN
C
C     ---FORMATS---
C
C     -----------------------------------------------------------------
C
```

```
C
  300 FORMAT ('1',49X,3A8,'LAYER',I4//)
  310 FORMAT ('0EXPLANATION'/' ',11('-')//' R = CONSTANT HEAD BOUNDARY'/
     1' *** = VALUE EXCEEDED 3 FIGURES'/' MULTIPLICATION FACTOR =',F8.3)
  320 FORMAT ('0',39X,6A8)
  330 FORMAT ('0',25X,10('*'),' TO FIT MAP WITHIN 12 INCHES, DINCH REVIS
     1ED TO',G15.7,1X,10('*'))
  340 FORMAT ('0',45X,'NOTE: GENERALLY SCALE SHOULD BE > OR = 1.0')
      END
```

```
 BLOCK DATA
 ----------
 SPECIFICATIONS:
 REAL *8XLABEL,YLABEL,TITLE,XN1,MESUR
 COMMON /SARRAY/ ICHK(13),LEVEL1(9),LEVEL2(9)
 COMMON /PR/ XLABEL(3),YLABEL(6),TITLE(6),XN1,MESUR,PRNT(122),BLANK
1(60),DIGIT(122),VF1(6),VF2(6),VF3(7),XSCALE,DINCH,SYM(17),XN(100),
2YN(13),NA(4),N1,N2,N3,YSCALE,FACT1,FACT2
 ***********************************************************************
 DATA ICHK/'DRAW','HEAD','MASS','DK1','DK2','WATE','RECH','PUN1','P
1UN2','ITKR','EQN3',2*0/
 DATA SYM/'1','2','3','4','5','6','7','8','9','0','*','|','-','+','
1 ','R','W'/
 DATA PRNT/122*' '/,N1,N2,N3,XN1/6,10,133,.833333333D-1/,BLANK/60*'
1 '/,NA(4)/1000/
 DATA XLABEL/' X DIS- ','TANCE IN','  MILES '/,YLABEL/'DISTANCE',' 
1FROM OR','IGIN IN ','Y DIRECT','ION, IN ','MILES   '/,TITLE/'PLOT 
2OF ','DRAWDOWN',' ','PLOT OF ','HYDRAULI','C HEAD'/
 DATA DIGIT/'1','2','3','4','5','6','7','8','9','10','11','12','13'
1,'14','15','16','17','18','19','20','21','22','23','24','25','26',
2'27','28','29','30','31','32','33','34','35','36','37','38','39','
340','41','42','43','44','45','46','47','48','49','50','51','52','5
43','54','55','56','57','58','59','60','61','62','63','64','65','66
5','67','68','69','70','71','72','73','74','75','76','77','78',' 79
6','80','81','82','83','84','85','86','87','88','89','90','91','92'
7,'93','94','95','96','97','98','99','100','101','102','103',' 104'
8,'105','106','107','108','109','110','111','112','113','114','115'
9,'116','117','118','119','120','121','122'/
 DATA VF1/'(1H ',',','    ','A1,F','10.2',')'/
 DATA VF2/'(1H ',',','    ','A1,1','X,A8',')'/
 DATA VF3/'(1H0',',','    ','A1,F','3.1,','12F1','0.2)'/
 ***********************************************************************
 END
```

FACTORS FOR CONVERTING INCH-POUND UNITS

TO INTERNATIONAL SYSTEM OF UNITS (SI)

The following factors can be used to convert inch-pound units to International System of Units (SI).

Multiply inch-pound units	By	To obtain SI Units
foot (ft)	0.3048	meter (m)
foot per day (ft/d)	0.3048	meter per day (m/d)
foot per day per foot [(ft/d)/ft]	0.3048	meter per day per meter [(m/d)/m]
square foot per day (ft^2/d)	9.290×10^{-2}	square meter per day (m^2/d)
cubic foot per second (ft^3/s)	2.832×10^{-2}	cubic meter per second (m^3/s)
inch (in)	2.54	centimeter (cm)
inch per year (in/year)	2.54	centimeter per year (cm/year)
mile (mi)	1.609	kilometer (km)
gallon per minute (gal/min)	6.310×10^{-2}	liter per second (L/s)
ohm-foot	0.3048	ohm-meter

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 "Mean Sea Level." NGVD of 1929 is referred to as sea level in this report.