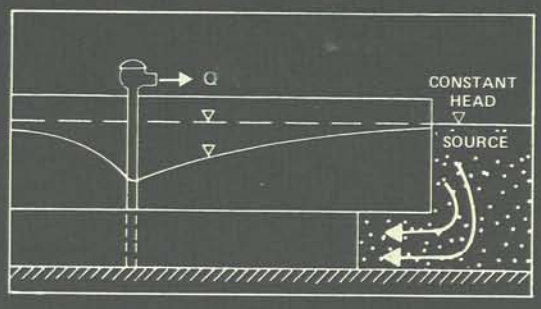
ISWS/BUL-65/81
BULLETIN 65
STATE OF ILLINOIS
ILLINOIS DEPARTMENT OF ENERGY AND NATURAL RESOURCES

A "Random-Walk" Solute Transport Model for Selected Groundwater Quality Evaluations

by THOMAS A. PRICKETT, THOMAS G. NAYMIK, and CARL G. LONNQUIST



ILLINOIS STATE WATER SURVEY
CHAMPAIGN
1981

CORRECTION

For figures 4 and 5, pages 10-11, Bulletin 65

On pages 10 and 11, the figure captions are on the correct pages, but the illustrations have been reversed. The correct illustration for figure 4 appears on page 11, and the correct illustration for figure 5 appears on page 10.

BULLETIN 65

A "Random-Walk" Solute Transport Model for Selected Groundwater Quality Evaluations

by THOMAS A. PRICKETT, THOMAS G. NAYMIK, and CARL G. LONNQUIST

Title: A "Random-Walk" Solute Transport Model for Selected Groundwater Quality Evaluations.

Abstract: A generalized computer code is given that can simulate a large class of solute transport problems in groundwater. The effects of convection, dispersion, and chemical reactions are included. The solutions for groundwater flow include a finite difference formulation. The solute transport portion of the code is based on a particle-in-a-cell technique for the convective mechanisms, and a random-walk technique for the dispersion effects. The code can simulate one- or two-dimensional nonsteady/steady flow problems in heterogeneous aquifers under water table and/or artesian or leaky artesian conditions. The program covers time-varying pumpage or injection by wells, natural or artificial recharge, the flow relationships of water exchange between surface waters and groundwater, the process of groundwater evapotranspiration, the mechanism of possible conversion of storage coefficients from artesian to water table conditions, and the flow from springs. It also allows specification of chemical constituent concentrations of any segment of the model. Further features of the program include variable finite difference grid sizes and printouts of input data, time series of heads, sequential plots of solute concentration distributions, concentrations of water flowing into sinks, and the effects of dispersion and dilution or mixing of waters having various solute concentrations.

Reference: Prickett, Thomas A., Thomas G. Naymik, and Carl G. Lonnquist. A "Random-Walk" Solute Transport Model for Selected Groundwater Quality Evaluations. Illinois State Water Survey, Champaign, Bulletin 65, 1981.

Indexing Terms: Aquifers, case histories, computer codes, computer programs, contaminant plumes, contamination, convection, dispersion, distribution coefficient, finite difference theory, groundwater, groundwater flow, groundwater quality, groundwater tracers, models, pollution, program listings, random walk, solute transport.

STATE OF ILLINOIS
HON. JAMES R. THOMPSON, Governor

DEPARTMENT OF ENERGY AND NATURAL RESOURCES
MICHAEL B. WITTE, B.A., Acting Director

BOARD OF NATURAL RESOURCES AND CONSERVATION

Michael B. Witte, B.A., Chairman

Walter E. Hanson, M.S., Engineering

Laurence L. Sloss, Ph.D., Geology

H. S. Gutowsky, Ph.D., Chemistry

Lorin I. Nevling, Ph.D., Forestry

William L. Everitt, E.E., Ph.D.,
University of Illinois

John C. Guyon, Ph.D.,
Southern Illinois University

STATE WATER SURVEY DIVISION
STANLEY A. CHANGNON, JR., M.S., Chief

CHAMPAIGN
1981

Printed by authority of the State of Illinois

(11-81-1500)

CONTENTS

(Concluded on next page)

CONTENTS *(Concluded)*

A "Random-Walk" Solute Transport Model for Selected Groundwater Quality Evaluations

by Thomas A. Prickett, Thomas G. Naymik, and Carl G. Lonnquist

ABSTRACT

A generalized computer code is given that can simulate a large class of solute transport problems in groundwater. The effects of convection, dispersion, and chemical reactions are included. The solutions for groundwater flow include a finite difference formulation. The solute transport portion of the code is based on a particle-in-a-cell technique for the convective mechanisms, and a random-walk technique for the dispersion effects.

The code can simulate one- or two-dimensional nonsteady/steady flow problems in heterogeneous aquifers under water table and/or artesian or leaky artesian conditions. Furthermore this program covers time-varying pumpage or injection by wells, natural or artificial recharge, the flow relationships of water exchange between surface waters and the groundwater reservoir, the process of groundwater evapotranspiration, the mechanism of possible conversion of storage coefficients from artesian to water table conditions, and the flow from springs.

In addition, the program allows specification of chemical constituent concentrations of any segment of the model including, but not limited to, injection of contaminated water by wells, vertically averaged salt-water fronts, leachate from landfills, leakage from overlying source beds of differing quality than the aquifer, and surface water sources such as contaminated lakes and streams.

Further features of the program include variable finite difference grid sizes and printouts of input data, time series of heads, sequential plots of solute concentration distributions, concentrations of water flowing into sinks, and the effects of dispersion and dilution or mixing of waters having various solute concentrations.

The discussion of the digital technique includes the necessary mathematical background, the documented program listing, explanations of job setup procedures, sample input data, theoretical versus computer comparisons, and one field application.

INTRODUCTION

Presently, there are four classes of problems of concern in studies of solute transport in groundwater: 1) chemical problems such as predicting TDS, Cl, nitrates, etc., as when dealing with sea water intrusion, excessive fertilizer applications, hazardous waste leachate, and injection of chemical wastes into the subsurface using disposal wells; 2) bacterial problems associated with cesspools, artificial recharge, sanitary landfills, waste injection wells, etc.; 3) thermal problems involving injection of hot water into the groundwater reservoir, development of geothermal energy, and induced infiltration of surface water having varying temperatures; and 4) multi-phase problems arising from such situations as development of steam-water systems, secondary recovery of water by air injection, and possible air-water interfaces due to overpumping. Although this transport program may be extended, the emphasis of this report is placed on the chemical problems.

One form of the governing equation for solute transport in one dimension is

$$\frac{\partial}{\partial x}\left(\frac{D}{R_d}\cdot\frac{\partial C}{\partial x}\right) - \frac{V}{R_d}\frac{\partial C}{\partial x} \pm C_s Q = \frac{\partial C}{\partial \tau} \quad (1)$$

DISPERSION - CONVECTION ± PRODUCTION = QUALITY ACCUMULATION

where

V = interstitial velocity
D = coefficient of hydrodynamic dispersion
D = $d_x V$ + D*, where d_x = longitudinal dispersivity and D* = coefficient of molecular diffusion (neglected in the following development)
x = space dimension
R_d = retardation factor
$C_S Q$ = source or sink function having a concentration C_s
C = concentration

Problems including solute transport in groundwater involve solving equation 1 in one, two, or (rarely) three dimensions. For the derivation of the equation and further explanation, refer to Freeze and Cherry (1979), Bear (1972), and Ogata (1970).

A popular numerical technique used for solving equation 1 is the method of characteristics (MOC), or the particle-in-a-cell method. MOC treats the equation in two parts. First, the convective term containing the velocity (V) is solved with an adaptation of the usual finite difference flow type of model. Then the dispersive term is solved by using another finite difference grid associated with the concentration distribution. A large number of computer—generated particles move about by the velocity vectors which are solved in the flow model and which carry the concentration information between the convection and dispersion terms during the solution of the equation. The description of the MOC is straightforward, but the computer code is highly involved and very expensive to operate, and it requires a large computer to effect a solution. Therefore, researchers have been looking for a more efficient and more direct way to solve problems concerning solute transport in groundwater. The conceived "random-walk" method follows.

The random-walk technique is based on the concept that dispersion in porous media is a random process. On a microscopic basis, dispersion may occur as shown in figure 1. As indicated in figure 1C, dispersion can take place in two directions even though the mean flow is in one direction to the right. A particle, representing the mass of a specific chemical constituent contained in a defined volume of water, moves through an aquifer with two types of motion. One motion is with the mean flow (along streamlines determined by finite differences), and the other is random motion, governed by scaled prob-

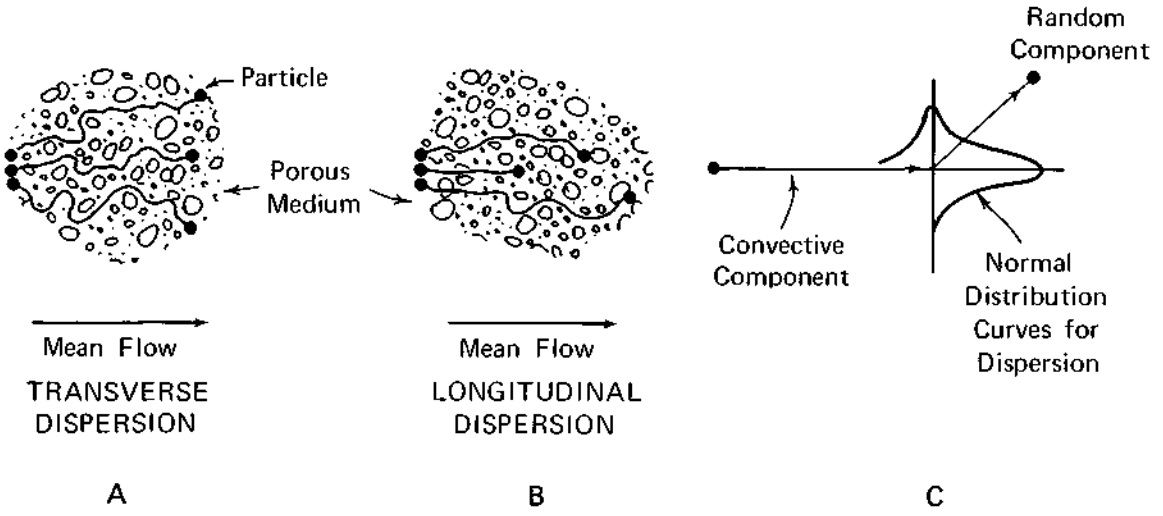


Figure 1. Basic concepts of "random-walk" computer program

ability curves related to flow length and the longitudinal and transverse dispersion coefficients. Finally, in the computer code, enough particles are included so that their locations and density, as they move through a flow model, are adequate to describe the distribution of the dissolved constituent of interest.

The advantages of this random-walk technique over the MOC or, for that matter, over many other numerical schemes are many:

1) There is no dispersion equation to solve. The dispersion part of equation 1 is solved in the computer code by the addition of only 11 Fortran statements attached to the solution of the convection part of equation 1.
2) There is only one finite difference grid involved in solving the convective portion of equation 1. The particle movement takes place in continuous space.
3) Concentration distribution needs to be calculated only when it is of interest. In the MOC, after each particle is moved, new concentrations are assigned on the basis of the solution of the dispersion term in the above equation for every time step of the sim ulation.
4) Computer CPU time is drastically reduced. The simulations in Part 4 of this bulletin took no more than a few seconds, including compiling and loading on a CDC CYBER 175.
5) Particles are needed only where water quality is of interest. These particles are not needed everywhere in the model, as with the MOC.
6) Solutions are additive. If not enough particles are included for adequate definition in one run, a second run can be done and the results accumulated. This is not true of the MOC, where possible sparse areas of particles may occur, causing loss of accuracy.
7) This method is particularly suited to time-sharing systems where velocity fields can be stored and manipulated in conjunction with an on-line particle mover code.
8) In the traditional sense of the words, finite differencing phenomena associated with "overshoot" and "numerical dispersion" are eliminated.

Although there are numerous advantages to this technique, there are also some disadvantages:

1) As with the MOC, concentrations greater than initial conditions are possible, especially when coarse discretizing is used.
2) A printout of concentrations may not be pleasing to the eye when the number of particles is small. (A Calcomp plotting and smoothing routine could be added to this model to eliminate this problem; however, this is beyond the scope and objectives of this report.)
3) The method may take an unusually large number of particles to produce an acceptable solution for some problems. No more than 5,000 particles are used in this report, however.
4) Engineering judgment is an absolute requirement in arriving at an acceptable solution. This is because of the "lumpy" character of the output. Therefore, experience with this technique is needed before one can apply the code successfully to a field situation.

The main objective of this report is to present a generalized computer code that will simulate a large class of solute transport problems involving convection and dispersion. In the present version, the effect of density-induced convection is not included. This complication is necessary only when a vertically averaged concentration distribution is inadequate. The class of problems that require a vertical averaged concentration distribution is sufficiently large, and the effects of density differences will be addressed in a later publication.

Acknowledgments

This report was prepared under the general supervision of Dr. William C. Ackermann, former Chief, and Stanley A. Changnon, Jr., present Chief of the Illinois State Water Survey. Richard J. Schicht, Assistant Chief, and James P. Gibb, Head of the Groundwater Section, made particularly useful suggestions with regard to this material.

The helpful comments of Adrian P. Visocky, Hydrologist, and Robert A. Sinclair, Head of the Data and Information Management Unit, are appreciated, as is the support from the Analytical Chemistry Laboratory Unit.

Thanks are also due Pamela Lovett for typing the manuscript and the camera copy; Gail Taylor for final editing; and John W. Brother, Jr., Linda Riggin, and William Motherway, Jr., for preparing the illustrations.

PART 1. MATHEMATICAL BACKGROUND

The computer code contained in this report can be broken into two parts. The first part relates to the "flow model" and the second part to the particle moving section which is termed the "solute transport model."

Some sort of model is needed to provide the velocity vectors of an aquifer flow system in order to calculate the convective movement of particles. In most of the situations to follow, a finite difference scheme is used to generate the head distribution. From this head distribution a velocity field is derived. The velocity field then provides the means of moving the particles advectively in the aquifer.

Calculations of Flow

Three methods for head calculations are written into the computer code in order to generate a head distribution which can then be used to calculate a velocity field. Two of these head calculations are analytical and the third is a finite difference method. A brief background of the finite difference scheme for producing a head distribution follows.

The partial differential equation (Jacob, 1950) governing the nonsteady-state two-dimensional flow of groundwater may be expressed as

$$\frac{\partial}{\partial x}\left(K_x b \frac{\partial h}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y b \frac{\partial h}{\partial y}\right) = S \frac{\partial h}{\partial t} + Q \qquad (2)$$

where

$K_x b = T_x$ = aquifer transmissivity in the x direction
$K_y b = T_y$ = aquifer transmissivity in the y direction
K_x = aquifer hydraulic conductivity in the x direction
K_y = aquifer hydraulic conductivity in the y direction
S = aquifer storage coefficient
h = head above bottom of aquifer
b = saturated thickness of aquifer
t = time
Q = source or sink functions expressed as net flow rates per unit area

A numerical solution of equation 2 can be obtained through a finite difference approach. The finite difference approach involves first replacing the continuous aquifer system parameters with an equivalent set of discrete elements. Second, the equations governing the flow of groundwater in the discretized model are written in finite difference form. Finally, the resulting set of finite difference equations is solved numerically with the aid of a digital computer.

A finite difference grid is superposed over a map of an aquifer, as illustrated in figure 2. The aquifer is thus subdivided into volumes having dimensions $b\Delta x\Delta y$, where b is the saturated thickness of the aquifer. The differentials ∂x and ∂y are approximated by the finite lengths Δx and Δy,

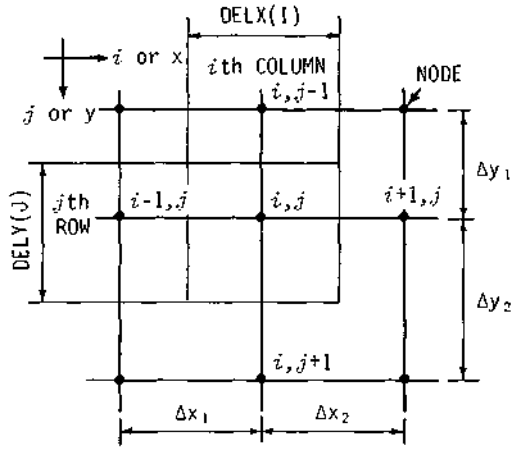


Figure 2. Finite difference grid

respectively. The area $\Delta x \Delta y$ should be small compared with the total area of the aquifer to the extent that the discrete model is a reasonable representation of the continuous system. The intersections of grid lines are called nodes and are referenced with a column (i) and row (j) coordinate system colinear with the x and y directions, respectively.

The general form of the finite difference equation governing the flow of groundwater in the discretized model is then given by

$$T_{i-1,j,2}(h_{i-1,j}-h_{i,j})/\Delta x^2 + T_{i,j,2}(h_{i+1,j}-h_{i,j})/\Delta x^2 + T_{i,j,1}(h_{i,j+1}-h_{i,j})/\Delta y^2 + T_{i,j-1,1}(h_{i,j-1}-h_{i,j})/\Delta y^2 = S_{i,j}(h_{i,j}-h\Phi_{i,j})/\Delta t + Q_{i,j}/\Delta x \Delta y \qquad (3)$$

where

$T_{i,j,1}$= aquifer transmissivity between nodes *i,j* and *i,j* + 1 calculated as $PERM_{i,j,1}$ times h where PERM is nydraulic conductivity

$T_{i,j,2}$= aquifer transmissivity between nodes *i,j* and *i*+1,*j* calculated as $PERM_{i,j,2}$ times h where PERM is hydraulic conductivity

$h_{i,j}$= calculated heads at nodes *i, j* at the end of a time increment measured from an arbitrary reference level

$h\Phi_{i,j}$ = calculated heads at nodes *i,j* at the end of the previous time increment measured from the same reference level defining $h_{i,j}$

Δt = time increment elapsed since last calculation of heads

$S_{i,j}$= aquifer storage coefficient at node *i,j*

$Q_{i,j}$= net withdrawal rate if positive, or net accretion rate if negative, at node *i,j*

Since there is an equation of the same form as equation 3 for every node of the digital model, a large set of simultaneous equations must be solved for the principal unknowns $h_{i,j}$. The modified iterative alternating direction implicit method (MIADI) given by Prickett and Lonnquist (1971) is used to solve the set of simultaneous equations.

Briefly, the MIADI method involves first, for a given time increment, reducing the large set of simultaneous equations down to a number of small sets. This is done by solving the node equations, by Gauss elimination, of an individual row of the model while all terms related to the nodes in the two adjacent rows are held constant. After all row equations have been processed row by row, the node equations are solved again by Gauss elimination for an individual column while all terms related to the two adjacent columns are held constant. After all equations have been solved column by column, an "iteration" has been completed. The above process is repeated until convergence is achieved, completing the calculations for the given time increment. The calculated heads are then used as initial conditions for the next time increment. This total process is repeated for successive time increments until the desired simulation is completed.

Equation 3 may be rewritten to illustrate the general form for calculations by rows. As a first simplification it is assumed that the finite difference grid is made up of squares such that $\Delta y = \Delta x$. (The case where Δy does not equal Δx is treated by Prickett and Lonnquist, 1971.) Equation 3 is then expanded, the signs are reversed, and terms of $h_{i,j}$ are grouped together to yield

$$-T_{i-1,j,2}h_{i-1,j} + h_{i,j}(T_{i-1,j,2} + T_{i,j,2} + T_{i,j,1} + T_{i,j-1,1} + S_{i,j}\Delta x^2/\Delta t) - T_{i,j,2}h_{i+1,j} = (S_{i,j}\Delta x^2/\Delta t)h\Phi_{i,j} + T_{i,j-1,1}h_{i,j-1} + T_{i,j,1}h_{i,j+1} - Q_{i,j} \qquad (4)$$

Equation 3 is of the form

$$AAh_{i-1,j} + BBh_{i,j} + CCh_{i+1,j} = DD$$

where

$$AA = -T_{i-1,j,2}$$

$$BB = T_{i-1,j,2} + T_{i,j,2} + T_{i,j,1} + T_{i,j-1,1} + S_{i,j}\Delta x^2/\Delta t$$

$$CC = -T_{i,j,2}$$

$$DD = (S_{i,j}\Delta x^2/\Delta t)h\phi_{i,j} + T_{i,j-1,1}h_{i,j-1} + T_{i,j,1}h_{i,j+1} - Q_{i,j}$$

A similar set of equations can be written for calculations by columns.

One of the flow calculations used in this report, which pertains to the above mathematical description, is the "composite aquifer simulation program" given by Prickett and Lonnquist (1971). The report by Prickett and Lonnquist is a companion to this report, as many details of flow modeling are taken directly from it for use in this report.

Let us emphasize again that several methods can be used to produce head distributions from which velocities can be calculated. As will be explained later, heads can be entered from theoretical distributions, field data, or other techniques involving totally different methods for numerically generating heads.

Once a head distribution is defined for all the nodes of the finite difference grid, then the velocity distribution is calculated according to

$$V = \frac{KI}{7.48n} \qquad (5)$$

where

V = interstitial velocity
K = hydraulic conductivity
I = hydraulic gradient
n = effective porosity

Since the model in this report includes nonhomogeneous aquifer properties and variable grid dimensions, the calculation of the velocity distribution by equation 5 accordingly takes this into account. Velocities are thus stored on the basis of either

V(I,J,1) = V defined midway between nodes I,J and I,J+1, or
V(I,J,2) = V defined midway between nodes I,J and I+1,J

These velocities are used as the input to a rather elaborate interpolation scheme to provide velocity vectors for the movement of each particle in continuous space.

Solute Transport Calculations

The basis for the transport calculations of dissolved constituents in this computer code is that the distribution of the concentration of chemical constituents of the water in an aquifer can be represented by the distribution of a finite number of discrete particles. Each of these particles is moved by groundwater flow and is assigned a mass which represents a fraction of the total mass of chemical constituent involved. In the limit, as the number of particles gets extremely large and approaches the molecular level, an exact solution to the actual situation is obtained. However, it is our experience that relatively few particles are needed to arrive at a solution that will suffice for many engineering applications .

There are two prime mechanisms which can change contaminant concentration in groundwater: dispersion, and dilution and mixing. The effects of mechanical dispersion as the fluid spreads through the pore space of the porous medium are described by the first and second terms on the left side of equation 1. The effects of dilution and mixing are expressed in the second and third terms on the left side of equation 1.

Dispersion

To illustrate the details of the random-walk technique as it relates to dispersion, consider the progress of a unit slug of tracer-marked fluid, placed initially at x = 0, in an infinite column of porous medium with steady flow in the x direction. With C_sQ equal to zero, equation 1 describes the concentration of the slug as it moves downstream. Bear (1972) describes the solution as

$$C(x,t) = \frac{1}{(4\pi d_L Vt)^{1/2}} \exp\left[-\frac{(x-Vt)^2}{4d_L Vt}\right] \quad (6)$$

where

C = concentration
d_L = longitudinal dispersivity
V = interstitial velocity
t = time
x = distance along the x axis

The shapes of the curves C(x',t) are shown in figure 3 where x' = x-Vt.

Based upon concepts found in comprehensive statistics books (for example, see Mood and Graybill, 1963) a random variable x is said to be normally distributed if its density function, n(x), is given by

$$n(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{(x-u)^2}{2\sigma^2}\right] \quad (7)$$

where

σ = standard deviation of the distribution
u = mean of the distribution

Now, let us equate the following terms of equations 6 and 7 as

$$\sigma = \sqrt{2d_L Vt} \quad (8)$$

$$u = Vt \quad (9)$$

$$n(x) = C(x,t) \quad (10)$$

With the identities of equations 8 through 10 taken into account, it will be seen that equations 6 and 7 are equivalent.

The key to solute transport as described in this report is the realization that dispersion in a porous medium can be considered a random process, tending to the normal distribution.

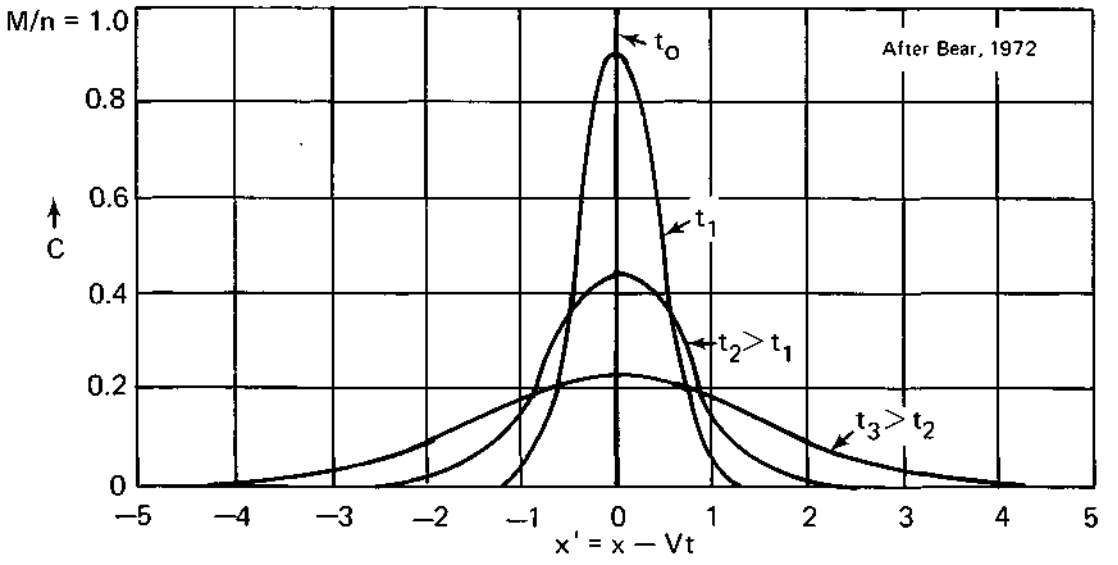


Figure 3. Progress of a slug around the mean flow

There are no new concepts here, as very complete discussions of statistical models of dispersion and the way dispersion affects water quality have been given by others, including Bear (1972) and Fried (1975). The method by which this statistical approach gets into the computer code and is applied to transport problems is new.

Figure 4A represents the way particles are moved in the computer code when the flow is in the x direction and one considers only longitudinal dispersion. During a time increment, DELP, a particle with coordinates xx,yy is first moved from an old to a new position in the aquifer by convection according to its velocity at the old position V_x. Then, a random movement in the +x or -x direction is added to represent the effects of dispersion. This random movement is given the magnitude

$$\sqrt{2d_L DX} \times \text{ANORM(0)} \qquad (11)$$

where

$$\sqrt{2d_L DX} = \sigma = \sqrt{2d_L V_x \text{ DELP}}$$

ANORM(0) = a number between -6 and +6, drawn from a normal distribution of numbers having a standard deviation of 1 and a mean of zero

The new position of the particle in figure 4A is the old position plus a convective term (V_x DELP) plus the effect of the dispersion term,

$$\sqrt{2d_L DX} \text{ ANORM(0)}$$

If the above process is repeated for numerous particles, all having the same initial position and convective term, a map of the new positions of the particles can be created having the discrete density distribution

$$C(x,t) \rightarrow n(x) \rightarrow \frac{N}{d_x}$$

$$= \frac{N_o}{\sqrt{2\pi}\sqrt{2d_x DX}} \exp\left[\frac{(x-V_x \text{ DELP})^2}{4d_x V_x T}\right] \qquad (12)$$

where

d_x = incremental distances over which N particles are found

N_o = total number of particles in the experiment

Equations 6, 7, and 12 are equivalent, with the exception that equations 6 and 7 are continuous distributions and equation 12 is discrete. As illustrated in figure 4A, the distribution of particles around the mean position, V_XDELP, is made to be normally distributed via the function ANORM(0). The function ANORM(0) is generated in the computer code as a simple function involving a summation of random numbers. Probable locations of particles, however, are considered only out to 6 standard deviations either side of the mean. On a practical basis, the probability is low of a particle moving beyond that distance.

One further emphasis is appropriate concerning the so-called "density function" of equations 6, 7, and 12. The equivalent density functions C(x,t) and N/d_x provide the means for relating the concentration of a contaminant in a field problem to the concentration of particles found in portions of a finite difference model. Various density functions will be defined later, by example, as they are needed for application purposes.

Figure 4B illustrates the extension of the random-walk method to account for dispersion in a direction transverse to the mean flow. Figures 5A and 5B illustrate the algebra involved when the flow is not aligned with the x-y coordinate system. Finally, figure 6 shows both longitudinal and transverse dispersion taking place simultaneously, and the appropriate vector algebra.

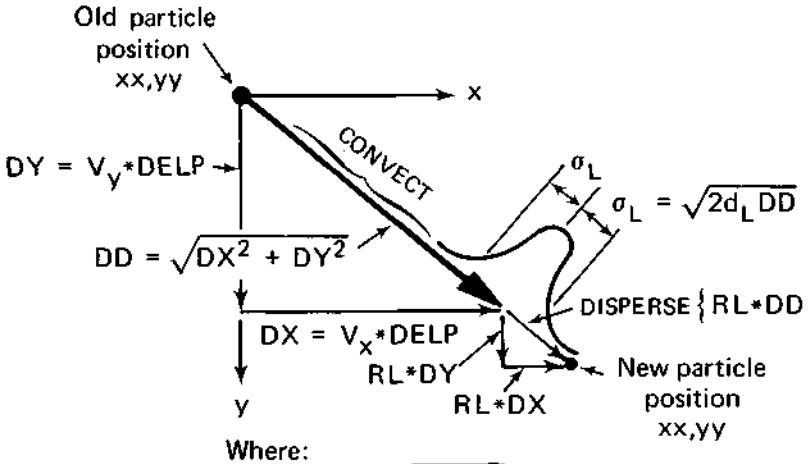
Dilution, Mixing, Retardation, and Radioactive Decay

Consider the one-dimensional flow problem in figure 7A in which the flow and concentrations of the sources are given. With dispersion set to zero and

(A) LONGITUDINAL DISPERSION

$d_L > 0$

$d_T = 0$



Where:

$RL*DD = \sqrt{2d_L DD}$ ANORM (0)

New position = Old position + Convection + Dispersion

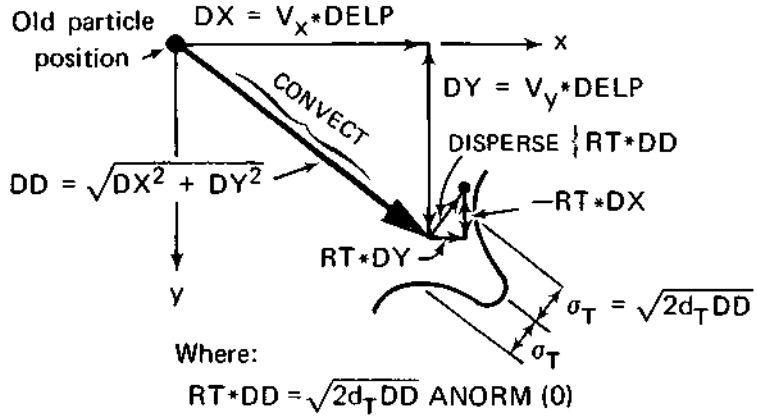
xx = xx + DX + RL*DX

yy = yy + DY + RL*DY

(B) TRANSVERSE DISPERSION

d_L = 0 (to avoid a zero divide in code — DISPL = 10^{-30})

$d_T > 0$



Where:

$RT*DD = \sqrt{2d_T DD}$ ANORM (0)

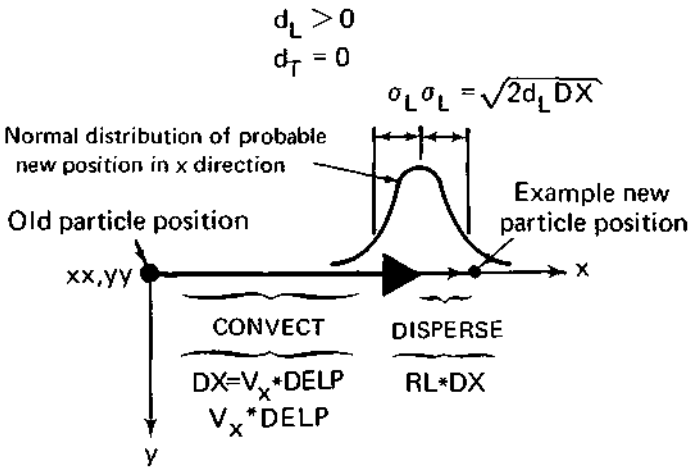
New position = Old position + Convection + Dispersion

xx = xx + DX + RT*DY

yy = yy + DY — RT*DX

Figure 4. Computer code scheme for convection and longitudinal (A) and transverse (B) dispersion along x axis

(A) LONGITUDINAL DISPERSION



Where:

$RL*DX = \sqrt{2d_L DX} * ANORM(0)$

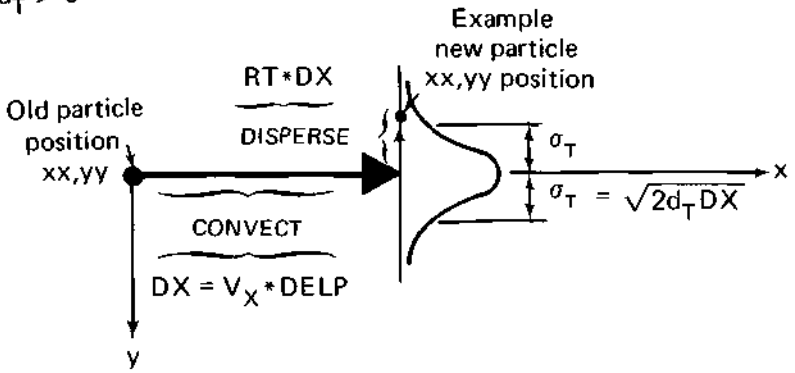
New position = Old position + Convection + Dispersion

xx	=	xx	+	DX	+	RL*DX
yy	=	yy				

(B) TRANSVERSE DISPERSION

d_L = 0 (to avoid a zero divide check SET to 10^{-30})

$d_T > 0$



Where:

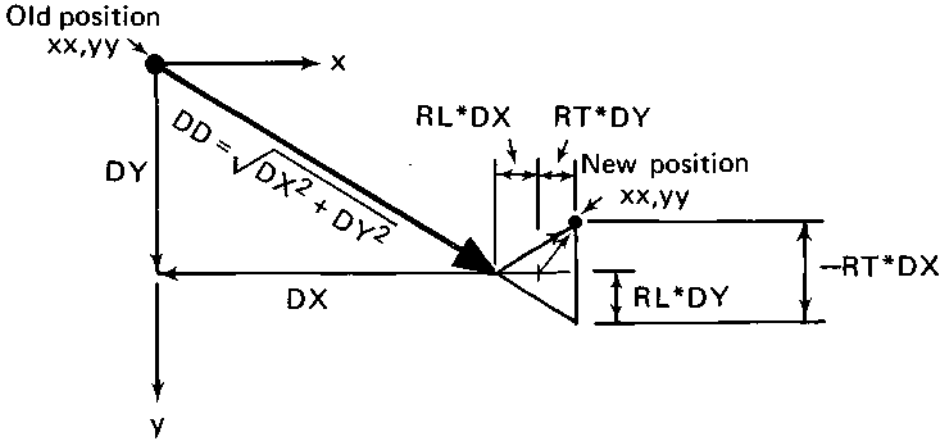
$RT*DX = \sqrt{2d_T DX}$ ANORM (0)

New position = Old position + Convection + Dispersion

xx	=	xx	+	DX	+	0
yy	=	yy	+	0	—	RT*DX

Figure 5. General scheme for convection and longitudinal (A) and transverse (B) dispersion

LONGITUDINAL AND
TRANSVERSE DISPERSION



$$RL*DD = \sqrt{2d_L DD}\ \text{ANORM (0)}$$

$$RT*DD = \sqrt{2d_T DD}\ \text{ANORM (0)}$$

New position	=	Old position	+	Convection	+	Longitudinal Dispersion	±	Transverse Dispersion
xx	=	xx	+	DX	+	RL*DX	+	RT*DY
yy	=	yy	+	DY	+	RL*DY	−	RT*DX

Figure 6. General scheme for convection and dispersion

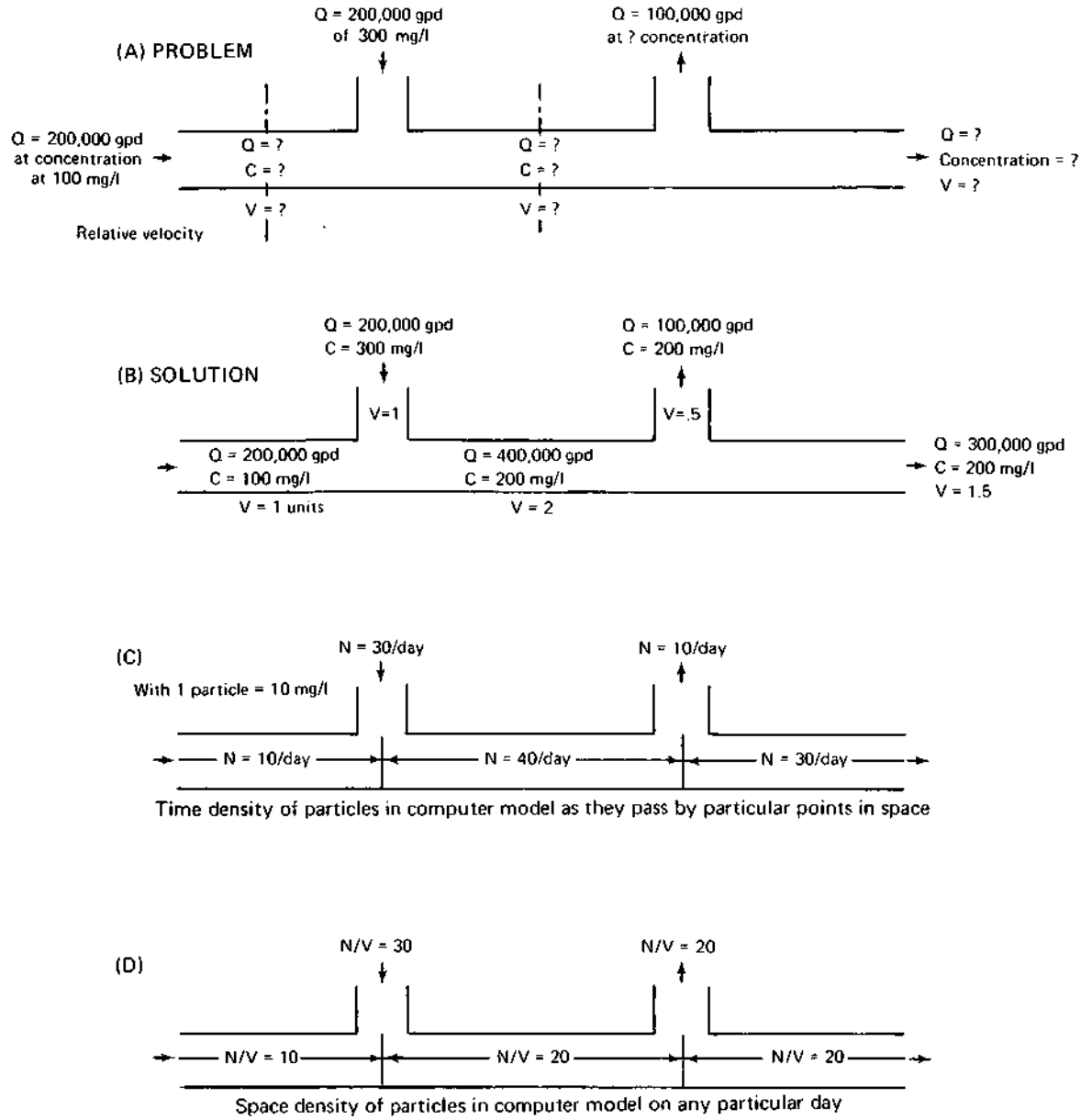


Figure 7. Mixing and dilution effects in water quality problems

retardation set to one, the distribution of concentrations in the system is simply a result of pure mixing as illustrated in figure 7B.

Now, let us assume that in the computer model one particle represents 10 mg/l. Figure 7C shows the time density of particles that would be used for input data in the computer model. Figure 7D shows the space density of particles in the computer model that would be simulated. Once the space density of particles is known, a multiplication by the particle mass yields the concentration of the flowing water.

In equation 1 the retardation factor (Rd) is used to represent the change in the solute concentration in the fluid caused by chemical reactions with the medium. These reactions include adsorption, organic fixation, etc. Chemical reactions between the dissolved constituent and the medium tend to retard the movement of the constituent relative to the groundwater velocity. The retardation of a concentration front in groundwater relative to the bulk mass of water is described by the relation

$$\frac{V}{V_c} = 1 + \frac{\rho_b}{n} \cdot K_d = R_d \qquad (13)$$

where

V = interstitial velocity of the groundwater

V_c = velocity of the C/C_o = 0.5 in the concentration front

ρ_b = bulk mass density

n = effective porosity

K_d = distribution coefficient

R_d = retardation factor

The transport code herein can take either K_d or R_d as input in order to incorporate the effects of chemical reactions. Later, in figure 42, example output will be shown for t = 1000 days using R_d = 1.0 and R_d = 2.0. Determining K_d or R_d for a constituent in groundwater requires a great deal of information regarding the composition of the groundwater and the interaction of the constituent with the medium. To pursue this, one should consult other references, such as Helfferich (1962), Higgins (1959), Baetsle (1967,1969), Freeze and Cherry (1979), and Borg et al. (1976).

Radioactive decay may be added to the model if desired. The statement "PM = PM*(0.5)**(TIME/HL)" should be added to subroutine ADVAN(DELP) after the statement "TMAP = TMAP+DELP." In the statement, HL is the half-life of the isotope in days. The number of days can either substitute for HL or be made an input parameter.

```
      PROGRAM TRANS(OUTPUT,TRIN,TROUT,TAPE5=TRIN,TAPE1=TROUT,
     X TAPE6=OUTPUT,DEBUG=OUTPUT)
C
C***  ILLINOIS STATE WATER SURVEY
C***  MASS TRANSPORT CODE WITH
C***  OR WITHOUT DISPERSION AND RETARDATION.
C
C***  THE TRANSPORT CODE IS THE FOLLOWING:
C***     DISPERSION--RANDOM WALK
C***     CONVECTION--PARTICLE IN A CELL
C
C***  THE MAJOR FLOW SUBROUTINE IS THE FOLLOWING:
C***     COMPOSITE AQUIFER SIMULATION PROGRAM
C***     LISTING FROM PAGES 51-52 OF "SELECTED
C***     DIGITAL COMPUTER TECHNIQUES FOR GROUNDWATER
C***     EVALUATION", ILLINOIS STATE WATER SURVEY
C***     BULLETIN 55, 1971, BY T.A. PRICKETT AND
C***     C.G. LONNQUIST.  THE FLOW CODE HANDLES
C***     HETEROGENEOUS AQUIFERS WITH LEAKY AND/OR
C***     NONLEAKY, INDUCED INFILTRATION, EVAPOTRANS-
C***     PIRATION, ARTESIAN TO WATER-TABLE STORAGE
C***     COEFFICIENT CONVERSION, ARTESIAN AND/OR
C***     WATER TABLE CONDITIONS, AND VARIABLE NET
C***     WITHDRAWAL RATES.  IN ADDITION, PRINTOUT
C***     OF INPUT DATA, VARIABLE GRID INTERVALS,
C***     AND VELOCITY CALCULATIONS ARE PROVIDED.
C
C
C***  DEFINITION OF VARIABLES
C***  VARIABLE
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  APOR          ACTUAL POROSITY
C***  ARRAY         ARRAY USED IN SYSTEM SUBROUTINE CALL
C***  B             PEACEMAN-RACHFORD B ARRAY
C***  BOT           ELEVATION OF BOTTOM OF AQUIFER (FT)
C***  BOTT          DEFAULT VALUE FOR BOT
C***  CCH           DEFAULT VALUE FOR CH
C***  CH            ELEVATION OF TOP OF AQUIFER (FT)
C***  CONSOR        CONCENTRATION, IN MG/L AT THE SOURCE
C***  DELP          TIME INCREMENT PARTICLES ARE ALLOWED TO MOVE(DAYS).
C***  DELTA         TIME INCREMENT (DAYS)
C***  DELX          MODEL GRID INTERVAL, IN FEET, IN THE X DIRECTION
C***  DELY          MODEL GRID INTERVAL, IN FEET, IN THE Y DIRECTION.
C***  DISPL         LONGITUDINAL DISPERSITY IN FEET
C***  DISPT         TRANSVERSE DISPERSIVITY IN FEET
C***  DX            SPACIAL INCREMENT ADDED TO X COORDINATE
C***                OVER WHICH PARTICLES WILL BE PLACED OR MOVED
C***  DY            SPACIAL INCREMENT ADDED TO Y COORDINATE OVER WHICH
C***                OVER WHICH PARTICLES WILL BE PLACED OR MOVED
C***  EPOR          EFFECTIVE POROSITY
C***  G             PEACEMAN-RACHFORD G ARRAY
C***  H             HEADS AT END OF TIME INCREMENT (FT)
C***  HH            DEFAULT VALUE FOR HEADS
C***  HO            HEADS AT START OF TIME INCREMENT
C***  I             INDEX VARIABLE
C***  IN            LOGICAL UNIT NUMBER FOR INPUT
C***  IP            X COORDINATE OF PUMP
C***  ISTEP         INDEX VARIABLE
C***  J             INDEX VARIABLE
C***  JP            Y COORDINATE OF PUMP
C***  K             INDEX VARIABLE
C***  KD            DISTRIBUTION COEFFICIENT
C***  M             INDEX VARIABLE
C***  MARK          TABLE OF SINK COORDINATES.  THE VALUE STORED
C***                IN MARK IS THE SINK NUMBER.  IF MARK IS 0, THERE
C***                IS NO SINK.
```

Figure 8. Listing of the transport program
(See Part 2)

```
C***  MAXP          MAXIMUM NUMBER OF PARTICLES IN SYSTEM.
C***  NC            NUMBER OF COLUMNS IN MODEL
C***  NP            TOTAL NUMBER OF PARTICLES IN SYSTEM
C***  NPITS         NUMBER OF RANDOM WALK TIME STEPS PER FLOW
C***                TIME STEP
C***  NPUMP         NUMBER OF PUMPS IN SYSTEM.
C***  NR            NUMBER OF ROWS IN MODEL
C***  NRT           NUMBER OF RATES IN PUMPING SCHEDULE
C***  NSP           NUMBER OF TIME INCREMENTS PER PUMPAGE CHANGE
C***  NSTEPS        NUMBER OF TIME INCREMENTS
C***  OUT           LOGICAL UNIT NUMBER FOR OUTPUT
C***  PERM          HYDRAULIC CONDUCTIVITY OF AQUIFER (GAL/DAY/SQ. FT.)
C***  PM            PARTICLE MASS
C***  Q             CONSTANT WITHDRAWAL RATES (GPD)
C***  QQ            DEFAULT WITHDRAWAL RATES (GPD)
C***  R             RECHARGE FACTOR (GAL/DAY/FT)
C***  RD            ELEVATION OF BOTTOM OF STREAMBED OR
C***                ELEVATION BELOW WHICH EVAPOTRANSPIRATION CEASES (FT)
C***  RD1           VECTOR RETARDATION FACTOR
C***  RH            ELEVATION OF LAND OR STREAM SURFACE (FT)
C***  RHO           BULK MASS DENSITY
C***  RR            DEFAULT RECHARGE FACTOR
C***  RRD           DEFAULT VALUE FOR RD
C***  RRH           DEFAULT VALUE FOR RH
C***  SF1           STORAGE CONDITIONS FOR ARTESIAN CONDITIONS (GAL)
C***  SF2           STORAGE CONDITIONS FOR WATER TABLE CONDITIONS (GAL/FT)
C***  SOR           VOLUME OF WATER TO BE ACCOUNTED FOR BEFORE NEXT
C***                PARTICLE IS EMITTED
C***  S1            DEFAULT VALUE FOR SF1
C***  S2            DEFAULT VALUE FOR SF2
C***  T             AQUIFER TRANSMISSIVITY (GAL/DAY*FT)
C***  TIME          TIME ELAPSED DURING SIMULATION
C***  TMAP          TIME SINCE PARTICLES WERE FIRST ALLOWED TO MOVE (DAYS)
C***  TT            DEFAULT VALUE FOR TRANMISSIVITY
C***  V             ARRAY CONTAINING COMPONENTS OF VECTOR VELOCITIES
C***  X1            TEMP VARIABLE
C***  Y1            TEMP VARIABLE
C***  Z             TEMP VARIABLE
C
C
C***  DECLARATIONS
C
      DIMENSION IP(100),JP(100),P(100,12)
      COMMON /RETARD/ RD1,KD,RHO
      REAL KD
      COMMON/EXTRA/PERM(30,29,2),T(30,29,2),
     1  SF1(30,29),SF2(30,29)
      COMMON/EXTRA3/ITER,NSTEPS,DL(30,29),ISTEP
      COMMON/POR/APOR,EPOR,CONSOR(30,29)
      COMMON/VEL/NC,NR,ANC,ANR,V(30,29,2)
      COMMON/TRACE/NP,MAXP,PM,DISPL,DISPT,X(5001),
     1  Y(5001),MARK(30,29),TMAP
      COMMON/EXTRA2/CH(30,29),HO(30,29),
     1  G(30),B(30),R(30,29),TIME,ERROR,E,ARRAY(6)
      COMMON/AQUI/H(30,29),RH(30,29),
     1  DELTA,Q(30,29),SOR(30,29)
      COMMON/VAR/DELX(50),DELY(50)
      COMMON/AQUI2/RD(30,29),BOT(30,29)
      COMMON/POL/X1,DX,Y1,DY,DELP
C
C***  DEFINE INPUT AND OUTPUT DEVICE NUMBERS
C
      INTEGER OUT
      DATA IN,OUT/5,6/
C
      DATA B/30*0/
      DATA ARRAY/6*0.0/
      CALL SYSTEMC(115,ARRAY)
```

Figure 8. Continued

```
C
C*** THE RANDOM NUMBER GENERATOR FOR THE CDC 6400/CYBER SYSTEM
C*** FORTRAN IS CALLED RANF.  WHEREVER RANF OCCURS IN THIS
C*** PROGRAM, IT CAN BE REPLACED WITH THE APPROPRIATE FUNCTION
C*** CALL FOR ANY OTHER SYSTEM.  THE RANDOM NUMBER GENERATOR
C*** MUST BE INITIALIZED, AND SUBROUTINE RANSET DOES THIS.
C*** THE CALL OF RANSET WITH ARGUMENT T CAUSES T TO BE USED AS
C*** THE SEED FOR THE RANDOM NUMBER GENERATOR.  SUBROUTINE
C*** SECOND WHEN CALLED WITH A PARAMETER OF T RETURNS THE
C*** NUMBER OF SECONDS PAST MIDNIGHT IN T.  THE FOLLOWING
C*** 2 STATEMENTS INITIALIZES THE SEED OF RANF TO BE
C*** THE NUMBER OF SECONDS PAST MIDNIGTH AT TIME OF EXECUTION.
C*** THIS PROVIDES FOR BETTER RANDOMNESS.
C
      CALL SECOND(T)
      CALL RANSET(T)
C
C***  READ PARAMETER CARD AND
C***  DEFAULT VALUE CARD
C
      READ(IN,10)NSTEPS,DELTA,ERROR,NPITS,
     1  NC,NR,TT,S1,HH,QQ,RR,RRH,RRD,S2,CCH,PP,BOTT
   10 FORMAT(I6,2F6.0,I6/2I6,11F6.0)
C
C***   WRITE OUT PARAMETER AND DEFAULT VALUES
C
      WRITE(OUT,20)
   20 FORMAT(/" VALUES ON PARAMETER CARD")
      WRITE(OUT,30)
   30 FORMAT(/" NSTEPS    DELTA    ERROR    NPITS")
      WRITE(OUT,40)NSTEPS,DELTA,ERROR,NPITS
   40 FORMAT(1X,I7,2X,E7.2,2X,E7.2,1X,I7)
      WRITE(OUT,50)
   50 FORMAT(/" VALUES ON DEFAULT VALUE CARD")
      WRITE(OUT,60)
   60 FORMAT(/" NC NR          TT          S1       ",
     1  "    S2          HH          QQ          RR    ",
     2  " RRH      PP    BOTT     RRD     CCH")
      WRITE(OUT,70)NC,NR,TT,S1,S2,HH,QQ,
     1  RR,RRH,PP,BOTT,RRD,CCH
   70 FORMAT(2I3,6F11.3,2F7.0,3F7.2)
C
C***  READ PUMP PARAMETER CARD
C
      READ(IN,80) NPUMP,NSP,NRT
   80 FORMAT(3I6)
C
C***  WRITE PUMP PARAMETER CARD
C
        WRITE(OUT,90)
   90   FORMAT(/" PUMP PARAMETER VALUES")
        WRITE(OUT,100)
  100   FORMAT(/"  NPUMP    NSP    NRT")
        WRITE(OUT,110)NPUMP,NSP,NRT
  110   FORMAT(3I6)
C
C***  READ PUMPING SCHEDULES
C
        DO 130 I=1,NPUMP
          READ(IN,120) IP(I),JP(I),
     1       (P(I,K),K=1,NRT)
  120     FORMAT(2I3,12F6.0)
  130     CONTINUE
C
C***  WRITE PUMPING SCHEDULES
C
```

Figure 8. Continued

```
      WRITE(OUT,140)
  140 FORMAT(/" PUMPING SCHEDULES")
      WRITE(OUT,150)
  150 FORMAT(" IP JP         P1         P2         P3",
     1    "         P4         P5         P6         P7",
     2    "         P8         P9        P10        P11",
     3    "        P12")
      DO 170 I=1,NPUMP
        WRITE(OUT,160)IP(I),JP(I),(P(I,M),M=1,NRT)
  160   FORMAT(2I3,2X,12F10.3/)
  170   CONTINUE
  175 CONTINUE
C
C***  READ VARIABLE GRID SIZES
C
      READ(IN,180)(DELX(I),I=1,NC)
  180 FORMAT(10F8.0)
      READ(IN,180)(DELY(J),J=1,NR)
      WRITE(OUT,190)
  190 FORMAT(/"    VALUES OF ROW J AND DELY")
      DO 210 J=1,NR
        WRITE(OUT,200)J,DELY(J)
  200   FORMAT(14X,I5,F8.0)
  210   CONTINUE
      WRITE(OUT,220)
  220 FORMAT(/" VALUES OF COLUMN I AND DELX")
      DO 240 I=1,NC
        WRITE(OUT,230)I,DELX(I)
  230   FORMAT(14X,I5,F8.0)
  240   CONTINUE
C
C*** FILL ARRAYS WITH DEFAULT VALUES
C
      DO 260 I=1,NC

        DO 250 J=1,NR
          T(I,J,1)=TT
          T(I,J,2)=TT
          PERM(I,J,1)=PP
          PERM(I,J,2)=PP
          SF1(I,J)=S1
          SF2(I,J)=S2
          H(I,J)=HH
          HO(I,J)=HH
          R(I,J)=RR
          RH(I,J)=RRH
          RD(I,J)=RRD
          CH(I,J)=CCH
          BOT(I,J)=BOTT
          DL(I,J)=0.0
          Q(I,J)=QQ
  250     CONTINUE
  260   CONTINUE
C
      DELX(NC+1)=0.0
      DELY(NR+1)=0.0
C
C***  READ AND WRITE POLLUTION INITIAL CONDITIONS
C
      READ(IN,270)X1,DX,Y1,DY,DELP
  270 FORMAT(5F10.0)
      WRITE(OUT,280)
  280 FORMAT(/" INITIAL LOCATION OF PARTICLES")
      WRITE(OUT,290)
  290 FORMAT(" AND TIME INCREMENT DELP")
      WRITE(OUT,300)
  300 FORMAT("         X1         DX         Y1",
     1  "         DY       DELP")
      WRITE(OUT,270)X1,DX,Y1,DY,DELP
C
```

Figure 8. Continued

```
C***  READ AND WRITE POLLUTION PARAMETERS
C
      READ(IN,310)PL,MAXP,PM,DISPL,DISPT,EPOR,APOR,RD1,KD,RHO
  310 FORMAT(F7.1,I7,8F7.1)
      CALL RDSOLV(EPOR,RHO,KD,RD1)
      WRITE(OUT,320)
  320 FORMAT(/" VALUES OF POLLUTION PARAMETERS")
      WRITE(OUT,330)
  330 FORMAT(/"         PL       MAXP         PM",
     1  "       DISPL    DISPT        EPOR       APOR       RD        KD",
     2  "       RHO")
      WRITE(OUT,340)PL,MAXP,PM,DISPL,DISPT,EPOR,APOR,RD1,KD,RHO
  340 FORMAT(1F10.0,I10,3F10.3,5F10.3)
      CALL INIT
C
C***  READ AND WRITE SINK LOCATION AND
C***  IDENTIFYING NUMBER.
C
        WRITE(OUT,343)
  343   FORMAT(/" LOCATION OF SINKS AND IDENTIFIER")
        WRITE(OUT,345)
  345   FORMAT(/"  I  J MARK")
  350 READ(IN,360)I,J,MRK
  360 FORMAT(3I3)
      IF (I.EQ.0.AND.J.EQ.0)GO TO 390
        MARK(I,J)=MRK
        WRITE(OUT,360)I,J,MARK(I,J)
        GO TO 350
  390   CONTINUE
C
C***  READ AND WRITE SOURCE CONCENTRATION DATA
C
      WRITE(OUT,400)
  400 FORMAT(/" LOCATION OF SOURCES AND THEIR",
     1  " CONCENTRATION")
  410 READ(IN,420)I,J,CONSOR(I,J)
  420 FORMAT(2I3,1F14.4)
      IF (I.EQ.0.AND.J.EQ.0)GO TO 430
        WRITE(OUT,420)I,J,CONSOR(I,J)
        GO TO 410
  430   CONTINUE
C
C***  READ AND WRITE NODE CARDS
C
      WRITE(OUT,440)
  440 FORMAT(/" NODE CARD VALUES")
      WRITE(OUT,450)
  450 FORMAT(/"  I  J         T1         T2        SF1",
     1  "        SF2          H          Q          R",
     2  "      RH   PERM1   PERM2     BOT      RD      CH")
  460 READ(5,470)I,J,T(I,J,1),T(I,J,2),
     1  SF1(I,J),H(I,J),Q(I,J),R(I,J),RH(I,J),
     2  RD(I,J),SF2(I,J),CH(I,J),PERM(I,J,1),
     3  PERM(I,J,2),BOT(I,J)
  470 FORMAT(2I3,2F6.0,2F4.0,9F6.0)
      IF ( EOF ( 5 ) .NE. 0 ) GO TO 490
        WRITE(OUT,480)I,J,T(I,J,1),T(I,J,2),SF1(I,J),
     1    SF2(I,J),H(I,J),Q(I,J),R(I,J),RH(I,J),PERM(I,J,1),
     2    PERM(I,J,2),BOT(I,J),RD(I,J),CH(I,J)
  480  FORMAT(2I3,2F10.0,2F10.5,2F10.0,1F10.3,6F7.0)
      GO TO 460
  490 CONTINUE
C
C***  ADJUST AQUIFER PARAMETERS FOR VARIABLE GRID
C
```

Figure 8. Continued

```
      DO 510 I=1,NC
        DO 500 J=1,NR
          V(I,J,1)=0.
          V(I,J,2)=0.
          T(I,J,1)=T(I,J,1)*2*DELX(I)/(DELY(J)+DELY(J+1))
          T(I,J,2)=T(I,J,2)*2*DELY(J)/(DELX(I)+DELX(I+1))
          SF1(I,J)=SF1(I,J)*7.48*DELX(I)*DELY(J)
          SF2(I,J)=SF2(I,J)*7.48*DELY(J)*DELX(I)
          R(I,J)=R(I,J)*DELY(J)*DELX(I)
          PERM(I,J,1)=2*PERM(I,J,1)*DELX(I)/(DELY(J)+
     1      DELY(J+1))
          PERM(I,J,2)=2*PERM(I,J,2)*DELY(J)/(DELX(I)+
     1      DELX(I+1))
  500     CONTINUE
  510   CONTINUE
C
C***  START OF SIMULATION
C
      TIME=0
      DELI=DELTA
      KC=1
      DO 630 ISTEP=1,NSTEPS
C
C***  ENTER PUMPAGE SCHEDULES
C*** IF THERE ARE NO PUMPS, SKIP THE NEXT SECTION OF CODE.
C
        IF ( NPUMP .EQ. 0) GO TO 535
          Z=(ISTEP-1.0)/NSP+1.0
          IF ( Z - KC ) 540 , 520 , 540
  520     CONTINUE
          DO 530 K=1,NPUMP
            I=IP(K)
            J=JP(K)
            Q(I,J)=P(K,KC)
  530       CONTINUE
  535     CONTINUE
        DELTA=DELI
        KC=KC+1
  540   CONTINUE
C
        CALL CLEAR
        CALL HSOLV4
C
C***  PRINT VALUES OF HEAD
C
        WRITE(OUT,550)TIME,ITER,E
  550   FORMAT(////" TIME ELAPSED SINCE PUMPING ",
     1    "STARTED=",F6.2,1X,"DAYS"/1X,"NUMBER OF ",
     2    "ITERATIONS FOR THIS TIME INCREMENT=",
     3    I5/1X,"SUM OF CHANGES IN HEAD FOR ALL NODE",
     4    " POINTS=",E20.7,"FT.")
        WRITE(OUT,560)
  560   FORMAT(/40X,"WATER LEVEL ELEVATIONS"/
     1    40X,22("*"))
        II=NC/10
        IF (MOD(NC,10).GT.0)  II=II+1
        DO 582 III=1,II
          IJ=(III-1)*10
          IIJ=MIN0(10,NC-IJ)
          DO 580 J=1,NR
            WRITE(OUT,570)(H(I+IJ,J),I=1,IIJ)
  570       FORMAT(1X,10F10.2)
  580       CONTINUE
      WRITE(OUT,581)
581    FORMAT(//)
  582    CONTINUE
C
C
```

Figure 8. Continued

```
C***  CALCULATE INTERSTITIAL VELOCITIES
C***  FOR NODES OF MODEL.
C
        DO 610 I=1,NC
          DO 600 J=1,NR
            IF (J.NE.NR)V(I,J,1)=-PERM(I,J,1)*(H(I,J+1)-
     1        H(I,J))/(DELX(I)*EPOR*7.48052)
            IF (I.NE.NC)V(I,J,2)=-PERM(I,J,2)*(H(I+1,J)-
     1        H(I,J))/(DELY(J)*EPOR*7.48052)
  600       CONTINUE
  610     CONTINUE
C
C***  CALL APPROPRIATE SUBROUTINES
C***  FOR AQUIFER SITUATION
C
        DO 620 I=1,NPITS
          CALL SUMMRY
          CALL SNKCON
          CALL CLEAR
          CALL ADVAN(DELP)
         CALL GENP4
      CALL MAP
      CALL CONMAP
  620     CONTINUE
C
        DELTA=DELTA
  630   CONTINUE
C
      STOP
      END
C*************************************************************************
C*************************************************************************
C***                                                                    ***
C***              SUBROUTINES AND FUNCTIONS                             ***
C***                                                                    ***
C*************************************************************************
C*************************************************************************
C
C***  ******************************************
C***  THIS SUBROUTINE IS CALLED WHEN PARTICLES
C***  ARE TO BE ADDED TO THE SYSTEM.  IF
C***  NUMBER OF PARTICLES TO BE ADDED IS
C***  GREATER THAN MAXP THEN PM CHANGES WHILE
C***  NP STAYS THE SAME.
C***  ******************************************
C
      SUBROUTINE ADD(XX,YY)
C
C*** DEPINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  DISPL         LONGITUDINAL DISPERSIVITY IN FEET
C***  DISPT         TRANSVERSE DISPERSIVITY IN FEET
C***  K             INDEX VARIABLE
C***  MARK          TABLE OF SINK COORDINATES.  THE VALUE STORED
C***                IN MARK IS THE SINK NUMBER.  IF MARK IS 0,
C***                THERE IS NO SINK.
C***  MAXP          MAXIMUM NUMBER OF PARTICLES IN SYSTEM
C***  NP            NUMBER OF PARTICLES IN SYSTEM
C***  PM            PARTICLE MASS
C***  TMAP          TIME SINCE PARTICLES WERE FIRST ALLOWED TO MOVE (DAYS).
C
C*** DECLARATIONS:
C
      COMMON/TRACE/NP,MAXP,PM,DISPL,DISPT,X(5001),
     1  Y(5001),MARK(30,29),TMAP
```

Figure 8. Continued

```
C
C*** BEGIN SUBROUTINE ADD:
C
      NP=NP+1
      X(NP)=XX
      Y(NP)=YY
      IF (NP.LT.MAXP)RETURN
      K=RANF(0)*MAXP+1.0
      X(K)=X(NP)
      Y(K)=Y(NP)
      PM=PM*MAXP/(MAXP-1)
      NP=NP-1
      RETURN
      END
C
C***  ******************************************
C***  THIS SUBROUTINE ADVANCES ALL PARTICLES
C***  TO THEIR NEW POSITIONS DURING TIME
C***  INCREMENT DELP
C***  ******************************************
C
      SUBROUTINE ADVAN(DELP)
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  DELP          TIME INCREMENT PARTICLES ARE ALLOWED TO MOVE (DAYS).
C***  DISPL         LONGITUDINAL DISPERSIVITY IN FEET
C***  DISPT         TRANSVERSE DISPERSIVITY IN FEET
C***  K             INDEX VARIABLE
C***  MARK          TABLE OF SINK COORDINATES.  THE VALUE STORED
C***                IN MARK IS THE SINK NUMBER.  IF MARK IS 0,
C***                THER IS NO SINK.
C***  MAXP          MAXIMUM NUMBER OF PARTICLES IN SYSTEM
C***  NP            NUMBER OF PARTICLES IN SYSTEM
C***  PM            PARTICLE MASS
C***  TMAP          TIME SINCE PARTICLES WERE FIRST ALLOWED TO MOVE (DAYS).
C
C*** DECLARATIONS:
C
      COMMON/TRACE/NP,MAXP,PM,DISPL,DISPT,X(5001),
     1  Y(5001),MARK(30,29),TMAP
C
C*** BEGIN SUBROUTINE ADVAN:
C
      TMAP=TMAP+DELP
      K=0
    1 K=K+1
    2 IF (K.GT.NP) RETURN
C
C***  IF A PARTICLE GOES INTO A SINK, THE LAST
C***  PARTICLE (NP) COORDINATES ARE THEN STORED
C***  IN THE THUS VACATED X(K) AND Y(K) POSITION.
C
      IF (MOVE(X(K),Y(K),DELP).EQ.0) GO TO 1
      X(K)=X(NP)
      Y(K)=Y(NP)
      NP=NP-1
      GO TO 2
      END
C
C***  ******************************************
C***  THIS FUNCTION PRODUCES A RANDOM
C***  NUMBER BETWEEN -6 AND +6,
C***  FOLLOWS A NORMAL DISTRIBUTION, AND
C***  HAS A STANDARD DEVIATION OF 1 AND A
C***  MEAN OF ZERO.
C***  ******************************************
```


Figure 8. Continued

```
C
        FUNCTION ANORM(K)
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  ANORM         RANDOM NUMBER (-6,6) WITH NORMAL DISTRIBUTION, ETC.
C***  I             INDEX VARIABLE
C***  K             INDEX VARIABLE
C
C*** BEGIN FUNCTION ANORM:
C
        ANORM=-6.0
        DO 10 I=1,12
          ANORM=ANORM+RANF(0)
   10   CONTINUE
        RETURN
        END
C
C***  ******************************************
C***  THIS SUBROUTINE CLEARS OUT NPART
C***  AND TABLE
C***  ******************************************
C
      SUBROUTINE CLEAR
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  I             INDEX VARIABLE
C***  J             INDEX VARIABLE
C***  NC            NUMBER OF COLUMNS IN THE MODEL
C***  NPART         NUMBER OF PARTICLES AT EACH NODE
C***  NR            NUMBER OF ROWS IN MODEL
C***  TABLE         MASS OF PARTICLES IN SINKS
C***  V             ARRAY CONTAINING MICROSCOPIC VELOCITIES OF PARTICLES
C
C*** DECLARATIONS:
C
      COMMON /VEL/ NC,NR,ANC,ANR,V(30,29,2)
      COMMON /FORMS/TABLE(50),NPART(30,29)
C
C*** BEGIN SUBROUTINE CLEAR:
C
      DO 20 I=1,NC
        DO 10 J=1,NR
          NPART(I,J)=0
   10     CONTINUE
   20   CONTINUE
      DO 30 I=1,50
        TABLE(I)=0.0
   30   CONTINUE
      RETURN
      END
C
C***  ******************************************
C***  THIS SUBROUTINE CONVERTS NPART
C***  TO A CONCENTRATION MAP
C***  ******************************************
C
      SUBROUTINE CONMAP
C
```

Figure 8. Continued

```
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  APOR          ACTUAL POROSITY
C***  BOT           ELEVATION OF BOTTOM OF AQUIFER (FT)
C***  CONC          CONCENTRATION OF PARTICLES (MG/L) AT EACH NODE
C***  CONSOR        CONCENTRATION AT THE SOURCE
C***  DELTA         TIME INCREMENT (DAYS).
C***  DELX          MODEL GRID INTERVAL, IN FEET, IN THE X DIRECTION
C***  DELY          MODEL GRID INTERVAL, IN FEET, IN THE Y DIRECTION
C***  EPOR          EFFECTIVE POROSITY
C***  H             HEADS AT END OF TIME INCREMENT (FT)
C***  I             INDEX VARIABLE
C***  J             INDEX VARIABLE
C***  MAXP          MAXIMUM NUMBER OF PARTICLES IN SYSTEM
C***  NC            NUMBER OF COLUMNS IN MODEL
C***  NP            NUMBER OF PARTICLES IN SYSTEM
C***  NPART         NUMBER OF PARTICLES AT EACH NODE
C***  NR            NUMBER OF ROWS IN MODEL
C***  PM            PARTICLE MASS
C***  Q             CONSTANT WITHDRAWAL RATES (GPD)
C***  RD            ELEVATION OF BOTTOM OF STREAMBED OR
C***                ELEVATION BELOW WHICH EVAPOTRANSPIRATION CEASES (FT).
C***  RH            ELEVATION OF LAND OR STREAM SURFACE (FT)
C***  TABLE         MASS OF PARTICLES IN SINKS
C***  V             ARRAY CONTAINING MICROSCOPIC VELOCITIES OF PARTICLES
C
C*** DECLARATIONS:
C
      COMMON/AQUI2/RD(30,29),BOT(30,29)
      COMMON/AQUI/H(30,29),RH(30,29),DELTA,Q(30,29)
      COMMON/VEL/NC,NR,ANC,ANR,V(30,29,2)
      COMMON/FORMS/TABLE(50),NPART(30,29)
      COMMON/POR/APOR,EPOR,CONSOR(30,29)
      COMMON/VAR/DELX(50),DELY(50)
      COMMON/TRACE/NP,MAXP,PM
      DIMENSION CONC(30,29)
      INTEGER OUT
      DATA OUT/6/
C
C*** BEGIN SUBROUTINE CONMAP:
C
      WRITE(6,1)
1     FORMAT(" MAP SHOWING CONCENTRATION IN PPM"/)
      DO 20 I=1,NC
        DO 10 J=1,NR
          CONC(I,J)=0.0
   10     CONTINUE
   20   CONTINUE
      DO 40 I=1,NC
        DO 30 J=1,NR
          CONC(I,J)=NPART(I,J)*PM*1E6/(APOR*(AMIN1(H(I,J),
     1      RD(I,J))-BOT(I,J))*(DELX(I)*DELY(J))*62.4)
   30     CONTINUE
   40   CONTINUE
      II=NC/10
      IF (MOD(NC,10).GT.0) II=II+1
      DO 70 III=1,II
         IJ=(III-1)*10
         IIJ=MIN0(10,NC-IJ)
         DO 60 J=1,NR
           WRITE(OUT,50)(CONC(I+IJ,J),I=1,IIJ)
   50      FORMAT(1X,10E10.3)
   60      CONTINUE
               WRITE(6,65)
65             FORMAT(//)
   70    CONTINUE
      RETURN
      END
```


Figure 8. Continued

```
C
C***  ********************************************
C***  THIS SUBROUTINE ENABLES PARTICLES
C***  TO BE INITIATED IN X, Y SPACE AND OVER
C***  TIME DELP RANDOMLY
C***  ********************************************
C
      SUBROUTINE GENP(PL)
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  DE            RANDOM PORTION OF DELP
C***  DELP          TIME INCREMENT PARTICLES ARE ALLOWED TO MOVE (DAYS).
C***  DISPL         LONGITUDINAL DISPERSIVITY IN FEET
C***  DISPT         TRANSVERSE DISPERSIVITY IN FEET
C***  DX            SPACIAL INCREMENT ADDED TO X COORDINATE
C***                OVER WHICH PARTICLES WILL BE PLACED OR MOVED..
C***  DY            SPACIAL INCREMENT ADDED TO Y COORDINATE
C***                OVER WHICH PARTICLES WILL BE  PLACED OR MOVED.
C***  MAXP          MAXIMUM NUMBER OF PARTICLES IN SYSTEM
C***  NP            NUMBER OF PARTICLES IN SYSTEM
C***  PL            TOTAL AMOUNT OF POLLUTANT
C***  PM            PARTICLE MASS
C
C*** DECLARATIONS:
C
      COMMON/TRACE/NP,MAXP,PM,DISPL,DISPT
      COMMON/POL/X1,DX,Y1,DY,DELP
C
C*** BEGIN SUBROUTINE GENP:
C
      P=PL
    1 IF (RANF(0).GT.P/PM)RETURN
        P=P-PM
        XX=X1+DX*RANF(0)
        YY=Y1+DY*RANF(0)
        DE=DELP*RANF(0)
        IF (MOVE(XX,YY,DE).EQ.0) CALL ADD(XX,YY)
        GO TO 1
      END
C
C***  ********************************************
C***  THIS SUBROUTINE GENERATES PARTICLES
C***  ALONG COLUMN 30.  IT IS USED IN
C***  THE EXAMPLE PROBLEMS IN THIS REPORT.
C***  ********************************************
C
      SUBROUTINE GENP2
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  I             INDEX VARIABLE
C
C*** BEGIN SUBROUTINE GENP2:
C
      DO 10 I=1,29
        XX=30.
        YY=30.-I
        CALL ADD(XX,YY)
   10   CONTINUE
      RETURN
      END
```

Figure 8. Continued

```
C
C***  *****************************************
C***  THIS SUBROUTINE EMITS 51 PARTICLES
C***  AT COORDINATES (1,2).  IT IS USED IN
C***  THE EXAMPLE PROBLEMS OF THIS BULLETIN.
C***  *****************************************
C
      SUBROUTINE GENP3
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  I             INDEX VARIABLE
C
C*** BEGIN SUBROUTINE GENP3:
C
      DO 10 I=1,51
        YY=2.
        XX=1.
        CALL ADD(XX,YY)
   10   CONTINUE
      RETURN
      END
C
C***  *****************************************
C***  THIS SUBROUTINE GENERATES PARTICLES AROUND
C***  A CIRCLE CONCENTRIC WITH THE WELL AT CENTER.
C***  THIS PARTICULAR SUBROUTINE GENERATES 360
C***  PARTICLES IN A CIRCLE AT A DISTANCE
C***  OF 0.7 GRIDS FROM THE CENTER OF THE INJECTION
C***  WELL AT MODEL COORDINATES (15,15) AND
C***  RANDOMLY DURING DELP.
C***  *****************************************
C
      SUBROUTINE GENP4
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  ANG           CONVERSTION FACTOR FROM ANGLES TO RADIANS
C***  ANGS          ANGLE IN DEGREES
C***  DE            RANDOM PORTION OF DELP
C***  DELP          TIME INCREMENT PARTICLES ARE ALLOWED TO MOVE (DAYS).
C***  DISPL         LONGITUDINAL DISPERSIVITY IN FEET
C***  DISPT         TRANSVERSE DISPERSIVITY IN FEET
C***  DX            SPACIAL INCREMENT ADDED TO X COORDINATE
C***                OVER WHICH PARTICLES WILL BE PLACED OR MOVED..
C***  DY            SPACIAL INCREMENT ADDED TO Y COORDINATE
C***                OVER WHICH PARTICLES WILL BE  PLACED OR MOVED.
C***  I             INDEX VARIABLE
C***  MARK          TABLE OF SINK COORDINATES.  THE VALUE STORED
C***                IN MARK IS THE SINK NUMBER.  IF MARK IS 0,
C***                THERE IS NO SINK.
C***  MAXP          MAXIMUM NUMBER OF PARTICLES IN SYSTEM
C***  N             NUMBER OF DEGREES IN A CIRCLE
C***  NP            NUMBER OF PARTICLES IN SYSTEM
C***  PM            PARTICLE MASS
C***  TH            ANGLE IN RADIANS
C***  TMAP          TIME SINCE PARTICLES WERE FIRST ALLOWED TO MOVE (DAYS).
C
```

Figure 8. Continued

```
C*** DECLARATIONS:
C
      COMMON/TRACE/NP,MAXP,PM,DISPL,DISPT,X(5001),
     1  Y(5001),MARK(30,29),TMAP
      COMMON/POL/X1,DX,Y1,DY,DELP
C
C*** BEGIN SUBROUTINE GENP4:
C
      XX=15.
      YY=15.
      ANG=2*3.141593
      RR=0.7
      ANGS=0
      N=360
      ANG=ANG/N
      DO 10 I=1,N
        TH=ANG*I+ANGS
        XX=15.+RR*SIN(TH)
        YY=15.+RR*COS(TH)
        DE=DELP*RANF(0)
        IF (MOVE(XX,YY,DE).EQ.0)CALL ADD(XX,YY)
   10   CONTINUE
      RETURN
      END
C
C***  ******************************************
C***  THIS SUBROUTINE GENERATES PARTICLES AROUND
C***  A CIRCLE CONCENTRIC WITH THE WELL AT CENTER.
C***  THIS PARTICULAR SUBROUTINE GENERATES 101
C***  PARTICLES IN A CIRCLE AT A DISTANCE
C***  OF 0.7 GRIDS FROM THE CENTER OF THE INJECTION
C***  WELL AT MODEL COORDINATES (15,15) AT THE
C***  ONSET OF THE CALL STATEMENT.
C***  ******************************************
C
      SUBROUTINE GENP5
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  ANG           CONVERSION FACTOR FROM DEGREES TO RADIANS
C***  ANGS          ANGLE IN DEGREES
C***  DISPL         LONGITUDINAL DISPERSIVITY IN FEET
C***  DISPT         TRANSVERSE DISPERSIVITY IN FEET
C***  I             INDEX VARIABLE
C***  MARK          TABLE OF SINK COORDINATES.  THE VALUE STORED
C***                IN MARK IS THE SINK NUMBER.  IF MARK IS 0,
C***                THERE IS NO SINK.
C***  MAXP          MAXIMUM NUMBER OF PARTICLES IN SYSTEM
C***  N             NUMBER OF DEGREES IN A CIRCLE
C***  NP            NUMBER OF PARTICLES IN SYSTEM
C***  PM            PARTICLE MASS
C***  TH            ANGLE IN RADIANS
C***  TMAP          TIME SINCE PARTICLES WERE FIRST ALLOWED TO MOVE (DAYS).
C
C*** DECLARATIONS:
C
      COMMON/TRACE/NP,MAXP,PM,DISPL,DISPT,X(5001),
     1  Y(5001),MARK(30,29),TMAP
C
```

Figure 8. Continued

```
C*** BEGIN SUBROUTINE GENP5:
C
      XX=15.
      YY=15.
      ANG=2*3.141593
      RR=0.7
      ANGS=0.
      N=101
      ANG=ANG/N
      DO 10 I=1,N
        TH=ANG*I+ANGS
        XX=15.+RR*SIN(TH)
        YY=15.+RR*COS(TH)
        CALL ADD(XX,YY)
   10   CONTINUE
      RETURN
      END
C
C***  ********************************************
C***  THIS SUBROUTINE SOLVES FOR THE HEADS OF THE
C***  AQUIFER SITUATION.  THIS CODE COMES DIRECTLY
C***  FROM BULLETIN 55.
C***  ********************************************
C
      SUBROUTINE HSOLVE
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  AA            USED TO CALCULATE PEACEMAN-RACHFORD ARRAYS
C***  B             PEACEMAN-RACHFORD B ARRAY
C***  BB            USED TO CALCULATE PEACEMAN-RACHFORD ARRAYS
C***  BOT           ELEVATION OF BOTTOM OF AQUIFER (FT)
C***  CC            USED TO CALCULATE PEACEMAN-RACHFORD ARRAYS
C***  CH            ELEVATION OF TOP OF AQUIFER
C***  D             USED TO CALCULATE PEACEMAN-RACHFORD ARRAYS
C***  DD            USED TO CALCULATE PEACEMAN-RACHFORD ARRAYS
C***  DELTA         TIME INCREMENT (DAYS).
C***  E             USED TO CALCULATE PEACEMAN-RACHFORD ARRAYS
C***  F             USED TO CALCULATE PEACEMAN-RACHFORD ARRAYS
C***  G             PEACEMAN-RACHFORD G ARRAY
C***  H             HEADS AT END OF TIME INCREMENT (FT)
C***  I             INDEX VARIABLE
C***  ISTEP         INDEX VARIABLE (FROM 1 TO NSTEPS).
C***  ITER          NUMBER OF ITERATIONS
C***  J             INDEX VARIABLE
C***  N             INDEX VARIABLE
C***  NC            NUMBER OF COLUMNS IN MODEL
C***  NR            NUMBER OF ROWS IN MODEL
C***  NSTEPS        NUMBER OF TIME INCREMENTS FOR MODEL
C***  PERM          HYDRAULIC CONDUCTIVITY OF AQUIFER (GAL/DAY/SQ. FT.)
C***  Q             CONSTANT WITHDRAWAL RATES (GPD)
C***  R             RECHARGE FACTOR (GAL/DAY/FT)
C***  RD            ELEVATION OF BOTTOM OF STREAMBED OR
C***                ELEVATION BELOW WHICH EVAPOTRANSPIRATION CEASES (FT).
C***  RH            ELEVATION OF LAND OR STREAM SURFACE (FT)
C***  SF1           STORAGE CONDITIONS FOR ARTESIAN CONDITIONS (GAL)
C***  SF2           STORAGE CONDITIONS FOR WATER TABLE CONDITIONS (GAL/FT)
C***  T             AQUIFER TRANSMISSIVITY (GAL/DAY/FT)
C***  TIME          TIME ELAPSED DURING SIMULATION
C***  V             ARRAY CONTAINING MICROSCOPIC VELOCITIES OF PARTICLES
C
```

Figure 8. Continued

```
C*** DECLARATIONS:
C
      COMMON/EXTRA3/ITER,NSTEPS,DL(30,29),ISTEP
      COMMON/EXTRA2/CH(30,29),HO(30,29),G(30),B(30),R(30,29),
     1  TIME,ERROR,E
      COMMON/EXTRA/PERM(30,29,2),T(30,29,2),SF1(30,29),SF2(30,29)
      COMMON/AQUI/H(30,29),RH(30,29),DELTA,Q(30,29)
      COMMON/AQUI2/RD(30,29),BOT(30,29)
      COMMON/VEL/NC,NR,ANC,ANR,V(30,29,2)
C
C*** BEGIN SUBROUTINE HSOVLE:
C
C
C***  PREDICT HEADS FOR NEXT
C***  TIME INCREMENT
C
      TIME=0.0
   10 DO 40 I=1,NC
        DO 30 J=1,NR
          D=H(I,J)-HO(I,J)
          HO(I,J)=H(I,J)
          F=1.0
          IF (DL(I,J).EQ.0.0)GO TO 20
            IF (ISTEP.GT.2)F=D/DL(I,J)
            IF (F.GT.5)F=5.0
            IF (F.LT.0.0)F=0.0
   20     DL(I,J)=D
          H(I,J)=H(I,J)+D*F
          IF (H(I,J).LE.BOT(I,J))H(I,J)=BOT(I,J)+0.01
   30     CONTINUE
   40   CONTINUE
C
C***  REFINE HEADS
C
C
C***  UPDATE TIME AND INITIALIZE ERROR
C***  AND ITERATION COUNTER
C
      TIME=TIME+DELTA
      ITER=0
   50 E=0.0
        ITER=ITER+1
C
C***  COLUMN CALCULATIONS
C
        DO 230 II=1,NC
          I=II
          IF (MOD(ISTEP+ITER,2).EQ.1)I=NC-I+1
          DO 200 J=1,NR
C
```

Figure 8. Continued

```
C***  CALCULATE PEACEMAN-RACHFORD
C***  B AND G ARRAYS
C
            AA=0.0
            IF (H(I,J).LT.BOT(I,J))H(I,J)=BOT(I,J)
            IF (H(I,J).LT.RD(I,J))GO TO 60
              RE=0.0
              RB=1.0
              GO TO 70
   60       RE=(RH(I,J)-RD(I,J))*R(I,J)
            RB=0.0
   70       IF (H(I,J).LT.CH(I,J))GO TO 80
              S=SF1(I,J)
              GO TO 90
   80       S=SF2(I,J)
   90       BB=S/DELTA+R(I,J)*RB
            DD=HO(I,J)*S/DELTA-Q(I,J)+RE
            IF ((HO(I,J)-CH(I,J))*(H(I,J)-CH(I,J))
     1        .LT.0.0)DD=DD+(HO(I,J)-CH(I,J))*(SF1(I,J)-
     2        SF2(I,J))/DELTA
            CC=0.0
            IF (J-1)100,110,100
  100         AA=-T(I,J-1,1)
              BB=BB+T(I,J-1,1)
  110       IF (J-NR)120,140,120
  120         IF (PERM(I,J,1).EQ.0.0)GO TO 130
                T(I,J,1)=PERM(I,J,1)*SQRT((AMIN1(H(I,J),
     1            CH(I,J))-BOT(I,J))*(AMIN1(H(I,J+1),CH(I,J+1))-
     2            BOT(I,J+1)))
  130         CC=-T(I,J,1)
              BB=BB+T(I,J,1)
  140       IF (I-1)150,160,150
  150         BB=BB+T(I-1,J,2)
              DD=DD+H(I-1,J)*T(I-1,J,2)
  160       IF (I-NC)170,190,170
  170         IF (PERM(I,J,2).EQ.0.0)GO TO 180
                T(I,J,2)=PERM(I,J,2)*SQRT((AMIN1(H(I,J),
     1            CH(I,J))-BOT(I,J))*(AMIN1(H(I+1,J),CH(I+1,J))-
     2            BOT(I+1,J)))
  180         BB=BB+T(I,J,2)
              DD=DD+H(I+1,J)*T(I,J,2)
  190       IF (J.GT.1) W=BB-AA*B(J-1)
            IF (J.EQ.1) W=BB
            B(J)=CC/W
            IF (J.EQ.1) G(J)=DD/W
            IF (J.GT.1) G(J)=(DD-AA*G(J-1))/W
  200       CONTINUE
C
C***  RE-ESTIMATE HEADS
C
          E=E+ABS(H(I,NR)-G(NR))
          H(I,NR)=G(NR)
          N=NR-1
  210     HA=G(N)-B(N)*H(I,N+1)
            E=E+ABS(HA-H(I,N))
            H(I,N)=HA
            N=N-1
            IF (N.GT.0)GO TO 210
          DO 220 N=1,NR
            IF (H(I,N).GT.BOT(I,N))GO TO 220
              E=E+BOT(I,N)+0.01-H(I,N)
              H(I,N)=BOT(I,N)+0.01
  220       CONTINUE
  230     CONTINUE
C
```

Figure 8. Continued

```
C***  ROW CALCULATIONS
C
          DO 410 JJ=1,NR
            J=JJ
            IF (MOD(ISTEP+ITER,2).EQ.1)J=NR-J+1
            DO 380 I=1,NC
              AA=0.0
              IF (H(I,J).LT.BOT(I,J))H(I,J)=BOT(I,J)
              IF (H(I,J).LT.RD(I,J))GO TO 240
                RE=0.0
                RB=1.0
                GO TO 250
  240         RE=(RH(I,J)-RD(I,J))*R(I,J)
              RB=0.0
  250         IF (H(I,J).LT.CH(I,J))GO TO 260
                S=SF1(I,J)
                GO TO 270
  260         S=SF2(I,J)
  270         BB=S/DELTA+R(I,J)*RB
              DD=HO(I,J)*S/DELTA-Q(I,J)+RE
              IF ((HO(I,J)-CH(I,J))*(H(I,J)-CH(I,J))
     1          .LT.0.0)DD=DD+(HO(I,J)-CH(I,J))*(SF1(I,J)-
     2          SF2(I,J))/DELTA
              CC=0.0
              IF (J-1)280,290,280
  280           BB=BB+T(I,J-1,1)
                DD=DD+H(I,J-1)*T(I,J-1,1)
  290         IF (J-NR)300,320,300
  300           IF (PERM(I,J,1).EQ.0.0)GO TO 310
                  T(I,J,1)=PERM(I,J,1)*SQRT((AMIN1(H(I,J),
     1              CH(I,J))-BOT(I,J))*(AMIN1(H(I,J+1),CH(I,J+1))-
     2              BOT(I,J+1)))
  310           DD=DD+H(I,J+1)*T(I,J,1)
                BB=BB+T(I,J,1)
  320         IF (I-1)330,340,330
  330           BB=BB+T(I-1,J,2)
                AA=-T(I-1,J,2)
  340         IF (I-NC)350,370,350
  350           IF (PERM(I,J,2).EQ.0.0)GO TO 360
                  T(I,J,2)=PERM(I,J,2)*SQRT((AMIN1(H(I,J),
     1              CH(I,J))-BOT(I,J))*(AMIN1(H(I+1,J),CH(I+1,J))-
     2              BOT(I+1,J)))
  360           BB=BB+T(I,J,2)
                CC=-T(I,J,2)
  370         IF (I.GT.1) W=BB-AA*B(I-1)
              IF (I.EQ.1) W=BB
              B(I)=CC/W
              IF (I.EQ.1) G(I)=DD/W
              IF (I.GT.1) G(I)=(DD-AA*G(I-1))/W
  380         CONTINUE
C
C***  RE-ESTIMATE HEADS
C
            E=E+ABS(H(NC,J)-G(NC))
            H(NC,J)=G(NC)
            N=NC-1
  390       HA=G(N)-B(N)*H(N+1,J)
              E=E+ABS(H(N,J)-HA)
              H(N,J)=HA
              N=N-1
              IF (N.GT.0)GO TO 390
            DO 400 N=1,NC
              IF (H(N,J).GT.BOT(N,J))GO TO 400
                E=E+BOT(N,J)+0.01-H(N,J)
                H(N,J)=BOT(N,J)+0.01
  400         CONTINUE
  410       CONTINUE
        IF (E.GT.ERROR) GO TO 50
  420 CONTINUE
      RETURN
      END
```

Figure 8. Continued

```
C
C***  ********************************************
C***  THIS SUBROUTINE DEFINES THE HEADS FOR A
C***  30 BY 29 MODEL HAVING UNIFORM
C***  FLOW WITH A VELOCITY OF 1 FT/DAY.THE
C***  GRID INTERVAL IS 10 FEET.  THE HYDRAULIC
C***  CONDUCTIVITY IS 10000 GPD/FT**2, AND THE
C***  EFFECTIVE POROSITY IS 0.2.
C***  ********************************************
C
      SUBROUTINE HSOLV2
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  H             HEADS AT END OF TIME INCREMENT (FT)
C***  I             INDEX VARIABLE
C***  J             INDEX VARIABLE
C
C*** DECLARATIONS:
C
      COMMON/AQUI/H(30,29)
C
C*** BEGIN SUBROUTINE HSOLV2:
C
      DO 20 I=1,30
        DO 10 J=1,29
          H(I,J)=.044883-I*.001496
   10     CONTINUE
   20   CONTINUE
      RETURN
      END
C
C***  ********************************************
C***  THIS SUBROUTINE PROVIDES THE PROPER
C***  HEADS FOR THE GRID OF THE MODEL DUE TO A SINGLE
C***  INJECTION WELL AT THE NODE COORDINATES 15,15
C***  AT A RATE OF 1,622,387 GPD.  THIS IS THE
C***  THEIM FORMULA AT WORK WITH A HEAD OF 0.0 AT
C***  10,000 FEET FROM A WELL OF RADIUS 208 FEET.
C***  THE INJECTION WELL IS PRESSURIZED TO 10 FEET
C***  OF HEAD.
C***  ********************************************
C
      SUBROUTINE HSOLV4
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  H             HEADS AT END OF TIME INCREMENT (FT)
C***  I             INDEX VARIABLE
C***  J             INDEX VARIABLE
C***  NC            NUMBER OF COLUMNS IN MODEL
C***  NR            NUMBER OF ROWS IN MODEL
C***  R             DISTANCE TO (15,15)
C***  V             ARRAY CONTAINING MICROSCOPIC VELOCITIES OF PARTICLES
C
C*** DECLARATIONS:
C
      COMMON/AQUI/H(30,29)
      COMMON/VEL/NC,NR,ANC,ANR,V(30,29,2)
C
```

Figure 8. Continued

```
C*** BEGIN SUBROUTINE HSOLVE4:
C
      DO 30 I=1,NC
        DO 20 J=1,NR
          X=I-15.
          Y=J-15.
          R=SQRT(X**2+Y**2)*1000
          IF (R.EQ.0.)GO TO 10
            H(I,J)=(1622387*2.302585*ALOG10(10000/R))/
     1         (2*3.141593*1E5)
   10     H(15,15)=10.
   20     CONTINUE
   30   CONTINUE
      RETURN
      END
C
C***  ******************************************
C***  IN THIS SUBROUTINE THE MARK ARRAY,
C***  TMAP AND NP ARE ZEROED OUT AND ANC
C***  AND ANR ARE INITIALIZED.
C***  ******************************************
C
      SUBROUTINE INIT
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  APOR          ACTUAL POROSITY
C***  B             PEACEMAN-RACHFORD B ARRAY
C***  CH            ELEVATION OF TOP OF AQUIFER (FT)
C***  CONSOR        CONCENTRATION AT THE SOURCE
C***  DELTA         TIME INCREMENT (DAYS).
C***  EPOR          EFFECTIVE POROSITY
C***  H             HEADS AT END OF TIME INCREMENT (FT)
C***  I             INDEX VARIABLE
C***  J             INDEX VARIABLE
C***  MARK          TABLE OF SINK COORDINATES.  THE VALUE STORED
C***                IN MARK IS THE SINK NUMBER.  IF MARK IS 0,
C***                THERE IS NO SINK.
C***  MAXP          MAXIMUM NUMBER OF PARTICLES IN SYSTEM
C***  NC            NUMBER OF COLUMNS IN MODEL
C***  NP            NUMBER OF PARTICLES IN SYSTEM
C***  NR            NUMBER OF ROWS IN MODEL
C***  PM            PARTICLE MASS
C***  Q             CONSTANT WITHDRAWAL RATES (GPD)
C***  SOR           VOLUME OF WATER TO BE ACCOUNTED FOR BEFORE
C***                NEXT PARTICLE IS EMITTED
C***  TMAP          TIME SINCE PARTICLES WERE FIRST ALLOWED TO MOVE (DAYS).
C***  V             ARRAY CONTAINING MICROSCOPIC VELOCITIES OF PARTICLES
C
C*** DECLARATIONS:
C
      COMMON/TRACE/NP,MAXP,PM,DISPL,DISPT,X(5001),
     1  Y(5001),MARK(30,29),TMAP
      COMMON/AQUI/H(30,29),RH(30,29),DELTA,Q(30,29)
     1  ,SOR(30,29)
      COMMON/EXTRA2/CH(30,29),HO(30,29),G(30),
     1  B(30),R(30,29),TIME,E
      COMMON /VEL/ NC,NR,ANC,ANR,V(30,29,2)
      COMMON/POR/APOR,EPOR,CONSOR(30,29)
C
```

Figure 8. Continued

```
C*** BEGIN SUBROUTINE INIT:
C
      TMAP=0.
      NP=0
      ANC=NC+0.5
      ANR=NR+0.5
      DO 20 I=1,NC
        DO 10 J=1,NR
          SOR(I,J)=0.0
          CONSOR(I,J)=0.0
          MARK(I,J)=0
   10     CONTINUE
   20   CONTINUE
      RETURN
      END
C
C***  ******************************************
C***  THIS SUBROUTINE PRINTS THE NPART MATRIX
C***  WHERE NUMBER OF PARTICLES ARE STORED.
C***  ******************************************
C
      SUBROUTINE MAP
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  I             INDEX VARIABLE
C***  J             INDEX VARIABLE
C***  MARK          TABLE OF SINK COORDINATES.  THE VALUE STORED
C***                IN MARK IS THE SINK NUMBER.  IF MARK IS 0,
C***                THERE IS NO SINK.
C***  MAXP          MAXIMUM NUMBER OF PARTICLES IN SYSTEM
C***  NC            NUMBER OF COLUMNS IN MODEL
C***  NP            NUMBER OF PARTICLES IN SYSTEM
C***  NPART         NUMBER OF PARTICLES AT EACH NODE
C***  NR            NUMBER OF ROWS IN MODEL
C***  PM            PARTICLE MASS
C***  TABLE         MASS OF PARTICLES IN SINKS
C***  TMAP          TIME SINCE PARTICLES WERE FIRST ALLOWED TO MOVE (DAYS).
C***  V             ARRAY CONTAINING MICROSCOPIC VELOCITIES OF PARTICLES
C
C*** DECLARATIONS:
C
      COMMON/TRACE/NP,MAXP,PM,DISPL,DISPT,X(5001),
     1  Y(5001),MARK(30,29),TMAP
      COMMON /FORMS/TABLE(50),NPART(30,29)
      COMMON/VEL/NC,NR,ANC,ANR,V(30,29,2)
C
C*** BEGIN SUBROUTINE MAP:
C
      WRITE (6,10)
   10 FORMAT(" MAP SHOWING NUMBER OF PARTICLES"
     1  ," RESIDING IN THE MODEL GRIDS")
      WRITE (6,20)PM,NP,TMAP
   20 FORMAT(" PARTICLE MASS= ",E15.4/
     1  " NUMBER OF PARTICLES IN SYSTEM = ",I5/
     2  " TIME = ",F8.2," DAYS"/)
      DO 40 J=1,NR
        WRITE(6,30)(NPART(I,J),I=1,NC)
   30   FORMAT(9X,30I3)
   40   CONTINUE
      RETURN
      END
C
```

Figure 8. Continued

```
C***  ******************************************
C***  THIS FUNCTION MOVES THE PARTICLES TO NEW
C***  LOCATIONS AS CALLED ON BY SUBROUTINES
C***  ADVAN OR GENP.  IT ALSO KEEPS TABS ON THE
C***  SINKS AND TABULATES WHEN PARTICLES
C***  ENTER SAME.  ALSO LIMITS MOVEMENT OF
C***  PARTICLES TO 0.2 GRIDS BEFORE USING NEW
C***  VELOCITIES TO CONTINUE MOVEMENT
C***  ALONG FLOW PATH.  FINALLY, THIS FUNC-
C***  TION COLLECTS INFORMATION ON
C***  NUMBER OF PARTICLES IN VOLUMES OF
C***  AQUIFER.
C***  ******************************************
C
      FUNCTION MOVE(XX,YY,DEL)
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  D             DAYS MOVED FOR THIS PARTICLE
C***  DD            DISTANCE PARTICLE HAS TRAVELLED
C***  DEL           TIME INCREMENTS THIS PARTICLE IS ALLOWED TO MOVE
C***  DELX          MODEL GRID INTERVAL, IN FEET, IN THE X DIRECTION
C***  DELY          MODEL GRID INTERVAL, IN FEET, IN THE Y DIRECTION
C***  DISPL         LONGITUDINAL DISPERSIVITY IN FEET
C***  DISPT         TRANSVERSE DISPERSIVITY IN FEET
C***  DX            SPACIAL INCREMENT ADDED TO X COORDINATE
C***                OVER WHICH PARTICLES WILL BE PLACED OR MOVED..
C***  DY            SPACIAL INCREMENT ADDED TO Y COORDINATE
C***                OVER WHICH PARTICLES WILL BE  PLACED OR MOVED.
C***  KD            DISTRIBUTION COEFFICIENT
C***  MARK          TABLE OF SINK COORDINATES.  THE VALUE STORED
C***                IN MARK IS THE SINK NUMBER.  IF MARK IS 0, THERE IS NO SINK.
C***  MAXP          MAXIMUM NUMBER OF PARTICLES IN SYSTEM
C***  MOVE          NUMBER OF SINK THAT GRABS THIS PARTICLE
C***  NC            NUMBER OF COLUMNS IN MODEL
C***  NP            NUMBER OF PARTICLES IN SYSTEM
C***  NR            NUMBER OF ROWS IN MODEL
C***  PM            PARTICLE MASS
C***  RD1           RETARDATION FACTOR
C***  RHO           BULK MASS DENSITY
C***  TABLE         MASS OF PARTICLES IN SINKS
C***  TMAP          TIME SINCE PARTICLES WERE FIRST ALLOWED TO MOVE (DAYS).
C***  V             ARRAY CONTAINING MICROSCOPIC VELOCITIES OF PARTICLES
C***  VX            VELOCITY IN THE X DIRECTION
C***  VY            VELOCITY IN THE Y DIRECTION
C
C*** DECLARATIONS:
C
      COMMON/VAR/DELX(50),DELY(50)
      COMMON/TRACE/NP,MAXP,PM,DISPL,DISPT,X(5001),
     1  Y(5001),MARK(30,29),TMAP
      COMMON /FORMS/TABLE(50),NPT(30,29)
      COMMON /VEL/ NC,NR,ANC,ANR,V(30,29,2)
      COMMON/RETARD/ RD1,KD,RHO
      REAL KD
C
C*** BEGIN FUNCTION MOVE:
C
      D=DEL
      DMIX=1/(2*DISPL)
    1 IF (D.LE.0.0) GO TO 2
        CALL VELO(XX,YY,VX,VY)
C
C***  LIMIT DISTANCE PARTICLE CAN MOVE BEFORE
C***  USING NEW VALUES OF VELOCITY
```

Figure 8. Continued

```
C
        IXX=XX
        IYY=YY
        T=AMIN1(D,.2*ABS(DELX(IXX)/VX),
     1     .2*ABS(DELY(IYY)/VY))
        IF (VX.EQ.0.)T=AMIN1(D,.2*ABS(DELY(IYY)/VY))
        IF (VY.EQ.0.)T=AMIN1(D,.2*ABS(DELX(IXX)/VX))
        IF (VX.EQ.0..AND.VY.EQ.0.)T=D
        D=D-T
        DX=VX*T
        DY=VY*T
      SIGNX=1.0
      SIGNY=1.0
      IF(DX.LT.0.0)SIGNX=-1.0
      IF(DY.LT.0.0)SIGNY=-1.0
      IF(DX.EQ.0.0)DX=1E-30
      IF(DY.EQ.0.0)DY=1E-30
      PHI=ATAN(DY/DX)
        DD=SQRT(DX**2+DY**2)/RD1
        IF (DD.EQ.0.0)DD=1E-30
        RL=(SQRT(2*DISPL*DD)/DD)*ANORM(0)
        RT=(SQRT(2*DISPT*DD)/DD)*ANORM(0)
        IXX=MIN0(NC,INT(XX+.5))
        IYY=MIN0(NR,INT(YY+.5))
      DX=DD*ABS(COS(PHI))*SIGNX
      DY=DD*ABS(SIN(PHI))*SIGNY
        XX=XX+(DX+RL*DX+RT*DY)/(DELX(IXX))
        YY=YY+(DY+RL*DY-RT*DX)/(DELY(IYY))
C
C***  THESE NEXT TWO STATEMENTS LIMIT THE
C***  MOVEMENT OF THE PARTICLES TO THE
C***  PLAYING FIELD DEFINED BY GRID COORDINANTS
C***  1,1; 1,NR; NC,1; AND NC,NR
C
        XX=AMIN1(ANC,AMAX1(XX,1.0))
        YY=AMIN1(ANR,AMAX1(YY,1.0))
C
C***  CHECK TO SEE IF PARTICLE GOES INTO A SINK.
C***  IF SO, SUM PARTICLE MASS INTO TABLE.
C
        IX=MIN0(NC,INT(XX+0.5))
        IY=MIN0(NR,INT(YY+0.5))
        MOVE=MARK(IX,IY)
        IF (MOVE.EQ.0)GO TO 1
         TABLE(MOVE)=TABLE(MOVE)+PM
        RETURN
C
C***  AFTER PARTICLE IS MOVED DELP DAYS,
C***  THIS SECTION TABULATES IN WHICH GRID THE
C***  PARTICLE RESIDES
C
    2 IX=XX+0.5
      IY=YY+0.5
      NPT(IX,IY)=NPT(IX,IY)+1
      MOVE=0
      RETURN
      END
C
C***   ********************************************************
C***   THIS SUBROUTINE CALCULATES THE RETARDATION
C***   FACTOR (RD1) FOR USE IN MOVE.  IF AN RD VALUE HAS BEEN
C***   READ IN, THEN THAT IS USED.  OTHERWISE, RD IS CALCULATED
C***   FROM KD AND RHO VALUES.
C***   ********************************************************
C
      SUBROUTINE RDSOLV(EPOR,RHO,KD,RD1)
C
C*** DEFINITION OF VARIABLES:
C
```

Figure 8. Continued

```
C***  NAM           DESCRIPTION
C***  ----          -----------
C***
C***  EPOR          EFFECTIVE POROSITY
C***  KD            DISTRIBUTION COEFFICIENT
C***  RDl           RETARDATION FACTOR
C***  RHO           BULK MASS DENSITY
C
C*** BEGIN SUBROUTINE RDSOLVE:
C
      REAL KD
      IF (RDl.NE.0) RETURN
        RDl=l+(RHO/EPOR)*KD
        RETURN
      END
C
C***  ********************************************
C***  THIS SUBROUTINE PRINTS OUT THE CONCENTRATION
C***  IN THE WATER FLOWING INTO SINKS SO DESIGNATED
C***  BY SPECIFYING A VALUE IN MARK(I,J).  TWO
C***  TYPES OF SINKS ARE ALLOWED.  THE FIRST TYPE
C***  IS A NODE THAT IS PUMPING (A VALUE IN THE
C***  Q(I,J) ARRAY FOR THIS NODE EXISTS).  THE
C***  SECOND TYPE IS A NODE THAT REPRESENTS
C***  NON-PUMPED LINES OF DISCHARGE SUCH AS A
C***  RIVER.
C***  ********************************************
C
      SUBROUTINE SNKCON
C
C*** DEFINITION OF VARIABLES:
C
C*** NAME          DESCRIPTION
C***  ----          -----------
C***
C***  CH            ELEVATION OF TOP OF AQUIFER (FT).
C***  CONSNK        CONCENTRATIONS OF WATER FLOWING INTO THE SINKS
C***  DELP          TIME INCREMENT PARTICLES ARE ALLOWED TO MOVE (DAYS).
C***  DELTA         TIME INCREMENT (DAYS).
C***  DISPL         LONGITUDINAL DISPERSIVITY IN FEET
C***  DISPT         TRANSVERSE DISPERSIVITY IN FEET
C***  DX            SPACIAL INCREMENT ADDED TO X COORDINATE
C***                OVER WHICH PARTICLES WILL BE PLACED OR MOVED..
C***  DY            SPACIAL INCREMENT ADDED TO Y COORDINATE
C***                OVER WHICH PARTICLES WILL BE  PLACED OR MOVED.
C***  MARK          TABLE OF SINK COORDINATES.  THE VALUE STORED
C***                IN MARK IS THE SINK NUMBER.  IF MARK IS 0,
C***                THERE IS NO SINK.
C***  MAXP          MAXIMUM NUMBER OF PARTICLES IN SYSTEM
C***  MOVE
C***  NC            NUMBER OF COLUMNS IN MODEL
C***  NP            NUMBER OF PARTICLES IN SYSTEM
C***  NR            NUMBER OF ROWS IN MODEL
C***  PM            PARTICLE MASS
C***  Q             CONSTANT WITHDRAWAL RATES (GPD)
C***  RH            ELEVATION OF LAND OR STREAM SURFACE (FT)
C***  TABLE         MASS OF PARTICLES IN SINKS
C***  TIME          TIME SINCE START OF SIMULATION
C***  TMAP          TIME SINCE PARTICLES WERE FIRST ALLOWED TO MOVE (DAYS).
C***  V             ARRAY CONTAINING MICROSCOPIC VELOCITIES OF PARTICLES
C
```

Figure 8. Continued

```
C*** DECLARATIONS:
C
      COMMON/EXTRA2/CH(30,29),HO(30,29),G(30),
     1  B(30),R(30,29),TIME,E
      COMMON/VEL/NC,NR,ANC,ANR,V(30,29,2)
      COMMON/TRACE/NP,MAXP,PM,DISPL,DISPT,X(5001),
     1  Y(5001),MARK(30,29),TMAP
      COMMON/AQUI/H(30,29),RH(30,29),DELTA,Q(30,29)
      COMMON/FORMS/TABLE(50)
      COMMON/POL/X1,DX,Y1,DY,DELP
      DIMENSION CONSNK(30,29)
C
C*** BEGIN SUBROUTINE SNKCON:
C
      DO 20 I=1,NC
        DO 10 J=1,NR
          CONSNK(I,J)=0.0
   10     CONTINUE
   20   CONTINUE
      DO 70 I=1,NC
        DO 60 J=1,NR
          IF (MARK(I,J).LT.1)GO TO 60
            MOVE=MARK(I,J)
            IF (TABLE(MOVE).EQ.0)GO TO 60
              IF (Q(I,J).GT.0.0)GO TO 30
                IF (R(I,J).EQ.0.)GO TO 60
                  IF ((H(I,J)-RH(I,J)).EQ.0.0)GO TO 60
                    CONSNK(I,J)=TABLE(MOVE)*1E6/((H(I,J)-
     1                RH(I,J))*DELP*R(I,J)*8.3453)
                  GO TO 40
   30           CONSNK(I,J)=TABLE(MOVE)*1E6/
     1            ((Q(I,J)*DELP*8.3453))
   40       CONTINUE
            IF (TABLE(MOVE).GT.0.0)WRITE(6,50)I,J,CONSNK(I,J)
   50       FORMAT(/" SINK COORDINATES",2I3,2X,F10.4)
   60       CONTINUE
   70   CONTINUE
      RETURN
      END
C
C***  ********************************************
C***  THIS SUBROUTINE EMITS PARTICLES WHENEVER
C***  FLOW FROM A SOURCE IS DETECTED.
C***  ********************************************
C
      SUBROUTINE SORGEN
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  APOR          ACTUAL POROSITY
C***  CH            ELEVATION OF TOP OF AQUIFER (FT).
C***  CONSOR        CONCENTRATION AT THE SOURCE
C***  DELP          TIME INCREMENT PARTICLES ARE ALLOWED TO MOVE (DAYS).
C***  DELTA         TIME INCREMENT (DAYS).
C***  DX            SPACIAL INCREMENT ADDED TO X COORDINATE
C***                OVER WHICH PARTICLES WILL BE PLACED OR MOVED..
C***  DY            SPACIAL INCREMENT ADDED TO Y COORDINATE
C***                OVER WHICH PARTICLES WILL BE  PLACED OR MOVED.
C***  EPOR          EFFECTIVE POROSITY
C***  NC            NUMBER OF COLUMNS IN MODEL
C***  NR            NUMBER OF ROWS IN MODEL
C***  PL            TOTAL AMOUNT OF POLLUTANT
C***  Q             CONSTANT WITHDRAWAL RATES (GPD)
C***  R             RECHARGE FACTOR (GAL/DAY/FT)
```

Figure 8. Continued

```
C***  RATE          RATE OF POLLUTANT ENTRY AT A SOURCE
C***  RH            ELEVATION OF LAND OR STREAM SURFACE (FT)
C***  V             ARRAY CONTAINING MICROSCOPIC VELOCITIES OF PARTICLES
C
C*** DECLARATIONS:
C
      COMMON/POR/APOR,EPOR,CONSOR(30,29)
      COMMON/AQUI/H(30,29),RH(30,29),
     1  DELTA,Q(30,29)
      COMMON/EXTRA2/CH(30,29),HO(30,29),
     1  G(30),B(30),R(30,29)
      COMMON/POL/X1,DX,Y1,DY,DELP
      COMMON/VEL/NC,NR,ANC,ANR,V(30,29,2)
C
C*** BEGIN SUBROUTINE SORGEN:
C
      DO 20 I=1,NC
        DO 10 J=1,NR
          RATE=(RH(I,J)-H(I,J))*R(I,J)
          IF (Q(I,J).LT.0.0)RATE=-Q(I,J)
          IF (RATE.LE.0.0)GO TO 10
            IF (CONSOR(I,J).EQ.0.0)GO TO 10
              PL=RATE*DELP*CONSOR(I,J)*8.3453/1E6
              CALL GENP(PL)
   10     CONTINUE
   20   CONTINUE
      RETURN
      END
C
C***  ********************************************
C***  THIS SUBROUTINE ALLOWS PARTICLES TO BE
C***  GENERATED FROM SPECIFIED SOURCES OF
C***  POLLUTION AT A RATE GOVERNED BY FLOW FROM
C***  THE SOURCE INTO THE AQUIFER.
C***  ********************************************
C
      SUBROUTINE SOURCE
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  A             NUMBER OF PARTICLES ADDED
C***  CH            ELEVATION OF TOP OF AQUIFER (FT).
C***  CONSOR        CONCENTRATION AT THE SOURCE
C***  DE            RANDOM PORTION OF DELP
C***  DELP          TIME INCREMENT PARTICLES ARE ALLOWED TO MOVE (DAYS).
C***  DELTA         TIME INCREMENT (DAYS).
C***  DX            SPACIAL INCREMENT ADDED TO X COORDINATE
C***                OVER WHICH PARTICLES WILL BE PLACED OR MOVED..
C***  DY            SPACIAL INCREMENT ADDED TO Y COORDINATE
C***                OVER WHICH PARTICLES WILL BE  PLACED OR MOVED.
C***  EPOR          EFFECTIVE POROSITY
C***  MAXP          MAXIMUM NUMBER OF PARTICLES IN SYSTEM
C***  NC            NUMBER OF COLUMNS IN MODEL
C***  NP            NUMBER OF PARTICLES IN SYSTEM
C***  NR            NUMBER OF ROWS IN MODEL
C***  PM            PARTICLE MASS
C***  Q             CONSTANT WITHDRAWAL RATES (GPD)
C***  R             RECHARGE FACTOR (GAL/DAY/FT)
C***  RH            ELEVATION OF LAND OR STREAM SURFACE (FT)
C***  SOR           VOLUME OF WATER TO BE ACCOUNTED FOR BEFORE NEXT
C***                PARTICLE IS EMITTED
C***  V             ARRAY CONTAINING MICROSCOPIC VELOCITIES OF PARTICLES
```

Figure 8. Continued

```
C
C*** DECLARATIONS:
C
      COMMON/EXTRA2/CH(30,29),HO(30,29),
     1  G(30),B(30),R(30,29)
      COMMON/AQUI/H(30,29),RH(30,29),
     1  DELTA,Q(30,29),SOR(30,29)
      COMMON/VEL/NC,NR,ANC,ANR,V(30,29,2)
      COMMON/POR/APOR,EPOR,CONSOR(30,29)
      COMMON/POL/X1,DX,Y1,DY,DELP
      COMMON/TRACE/NP,MAXP,PM
C
C*** BEGIN SUBROUTINE SOURCE:
C
      DO 60 I=1,NC
        DO 50 J=1,NR
          A=0.
          DELSOR=DELP
          RATE=(RH(I,J)-H(I,J))*R(I,J)
          IF (Q(I,J).LT.0.0)RATE=-Q(I,J)
          IF (RATE.LE.0.0)GO TO 50
            IF (CONSOR(I,J).EQ.0.0)GO TO 50
              DELR=SOR(I,J)/RATE
              IF (DELR.GT.DELP)GO TO 40
   10           DELSOR=DELSOR-DELR
                XX=I*1.
                YY=J*1.
                DE=DELSOR
                IF (MOVE(XX,YY,DE).EQ.0.0)CALL ADD(XX,YY)
                A=A+1
                IF (A.GT.1)GO TO 20
                  P=RATE*(DELP-DELR)*CONSOR(I,J)*8.3453/1E6
   20             K=P/PM
                  IF (K.LT.1)GO TO 30
                    P=P-PM
                    DELR=PM*1E6/(CONSOR(I,J)*8.3452*RATE)
                    GO TO 10
   30             SOR(I,J)=(PM-P)*1E6/(CONSOR(I,J)*8.3453)
                  GO TO 50
   40           SOR(I,J)=SOR(I,J)-DELP*RATE
   50       CONTINUE
   60     CONTINUE
        RETURN
        END
C
C***   ********************************************
C***   THIS SUBROUTINE PRINTS THE NUMBER OF
C***   PARTICLES CAPTURED BY SINK DURING
C***   TIME INCREMENT DELP.
C***   ********************************************
C
      SUBROUTINE SUMMRY
C
C*** DEFINITION OF VARIABLES:
C
C***   NAME          DESCRIPTION
C***   ----          -----------
C***
C***   I             INDEX VARIABLE
C***   TABLE         MASS OF PARTICLES IN SINKS
C
C*** DECLARATIONS:
C
      COMMON /FORMS/TABLE(50)
C
```

Figure 8. Continued

```
C*** BEGIN SUBROUTINE SUMMRY:
C
      DO 20 I=1,50
        IF (TABLE(I).GT.0.0)WRITE(6,10)I,TABLE(I)
   10   FORMAT(" SINK#",I2,F10.4)
   20   CONTINUE
      RETURN
      END
C
C***  ******************************************
C***  THIS FUNCTION PROVIDES PROPER
C***  VELOCITIES TO SUBROUTINE VELO.
C***  ******************************************
C
      FUNCTION V(I,J,K)
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  I             INDEX VARIABLE
C***  J             INDEX VARIABLE
C***  K             INDEX VARIABLE
C***  NC            NUMBER OF COLUMNS IN MODEL
C***  NR            NUMBER OF ROWS IN MODEL
C***  V             ARRAY CONTAINING MICROSCOPIC VELOCITIES OF PARTICLES
C***  VV            ARRAY CONTAINING MICROSCOPIC VELOCITIES OF PARTICLES
C
C*** DECLARATIONS:
C
      COMMON /VEL/NC,NR,ANC,ANR,VV(30,29,2)
C
C*** BEGIN FUNCTION V:
C
      V=VV(I,J,K)
      RETURN
      END
C
C***  ******************************************
C***  THIS SUBROUTINE CALCULATES VELOCITIES OF
C***  PARTICLES BY DOUBLE INTERPOLATION
C***  OF NODAL VELOCITIES, CHECKS BOUNDARIES
C***  TO KEEP PARTICLES INSIDE MODEL, AND
C***  INSURES PROPER VELOCITY DETERMINATIONS.
C***  ******************************************
C
      SUBROUTINE VELO(XX,YY,VX,VY)
C
C*** DEFINITION OF VARIABLES:
C
C***  NAME          DESCRIPTION
C***  ----          -----------
C***
C***  I             INDEX VARIABLE
C***  J             INDEX VARIABLE
C***  NC            NUMBER OF COLUMNS IN MODEL
C***  NR            NUMBER OF ROWS IN MODEL
C***  V             ARRAY CONTAINING MICROSCOPIC VELOCITIES OF PARTICLES
C***  VX            VELOCITY IN THE X DIRECTION
C***  VY            VELOCITY IN THE Y DIRECTION
C
C*** DECLARATIONS:
C
      COMMON /VEL/NC,NR,ANC,ANR,V(30,29,2)
C
```

Figure 8. Continued

```
C*** BEGIN SUBROUTINE VELO:
C
      LIJ(NR,J,Y)=MIN0(NR-1,MAX0(1,J+
     1  IFIX(SIGN(1.0,Y-0.5))))
      I=XX
      IF (I.LE.0)I=1
      IF (I.GE.NC)I=NC-1
      J=YY
      IF (J.LE.0)J=1
      IF (J.GE.NR)J=NR-1
      X=XX-I
      Y=YY-J
      VX=Y*V(I,J+1,2)+(1.-Y)*V(I,J,2)
      VY=X*V(I+1,J,1)+(1.-X)*V(I,J,1)
      JY=LIJ(NR,J,Y)
      IX=LIJ(NC,I,X)
      IF (ABS(VY).LE.1E-30) GO TO 10
         IF (JY.EQ.J) GO TO 10
          YZ=ABS(Y-0.5)
          VY=VY*(1.0-YZ)+YZ*
     1       (X*V(I+1,JY,1)+(1.0-X)*V(I,JY,1))
   10 IF (IX.EQ.I) GO TO 20
        IF (ABS(VX).LE.1E-20) GO TO 20
          XZ=ABS(X-0.5)
          VX=VX*(1.0-XZ)+XZ*
     1       (Y*V(IX,J+1,2)+(1.0-Y)*V(IX,J,2))
   20 CONTINUE
      IF (ABS(VX).LE.1E-30)VX=1E-30
      IF (ABS(VY).LE.1E-30)VY=1E-30
      RETURN
      END
```

Figure 8. Concluded

PART 2. RANDOM-WALK SOLUTE TRANSPORT PROGRAM

The solute transport code listing given in figure 8 was coded in FORTRAN IV to solve the sets of equations involved. This section illustrates what parameters are included and how to set up a simulation model. The operational sequences of the computer code are given later in Part 3.

Figure 9 illustrates some important parameters included in the program. Briefly, the program can simulate one- or two-dimensional nonsteady/steady flow problems in heterogeneous aquifers under water table and/or artesian or leaky artesian conditions. Furthermore, this program covers time-varying pumpage or injection from or into wells, natural or artificial recharge rates, the relationships of water exchange between surface waters and the groundwater reservoir, the process of groundwater evapotranspiration, the mechanism of possible conversion of storage coefficients from artesian to water table conditions, and the mechanism of flow from springs. In addition, the program allows specification of the water quality concentrations of any part of the model including, but not limited to, injection well water, salt-water fronts, leachate from landfills, leakage from overlying source beds of different quality than the aquifer, and surface water sources such as lakes and streams. Further features of the code allow variable finite difference grid sizes and printouts of input data, time series of aquifer heads, and sequences of maps of aquifer water concentration, concentration of water flowing from sinks, and the effects of dispersion and mixing of water of various concentrations.

The program listing of figure 8 is written in such a way as to be in the gallon-day-foot system of units. However, this code can be converted to a consistent set of units by simply replacing the constants, 7.48 gallons/cubic foot and 8.3453 pounds/gallon, by the equivalent measure of the units desired.

Job Setup

The computer job setup will be explained in general first, and then in more detail as necessary.

The aquifer system properties are discretized by superposing a finite difference grid over a map of the aquifer system as shown in figure 10. The total dimensions of the grid are defined by NC, the number of columns of the model, and by NR, the number of rows of the model. Next, a parameter card (line) and default value card (line) are prepared according to the formats illustrated in figures 11A and 11B. The default value card (line) provides data for simulating an NC by NR aquifer system model having homogeneous properties with identical initial heads and net withdrawal rates. Then, pumping and parameter schedule cards (lines) are prepared according to the card (line) format illustrated in figures 11C and 11D. Next, a group of cards (lines) is prepared which define the sizes of the variable grid in the x and y directions according to the format shown in figures HE and 11F. Following this are the cards (lines) pertaining to the water quality aspects of the problem. Included are the card for initial conditions of pollution (see format in figure 11G) and the card for the particle information, aquifer dispersivities, and porosity of the aquifer (see format in figure 11H). The sink location cards are then prepared according to the format illustrated in figure HI, and the source concentration cards are prepared according to figure 11J. Finally, a node card deck is prepared according to the format illustrated in figure UK. The node card deck contains one card for each node that has any aquifer system properties differing from those defined on the default value card.

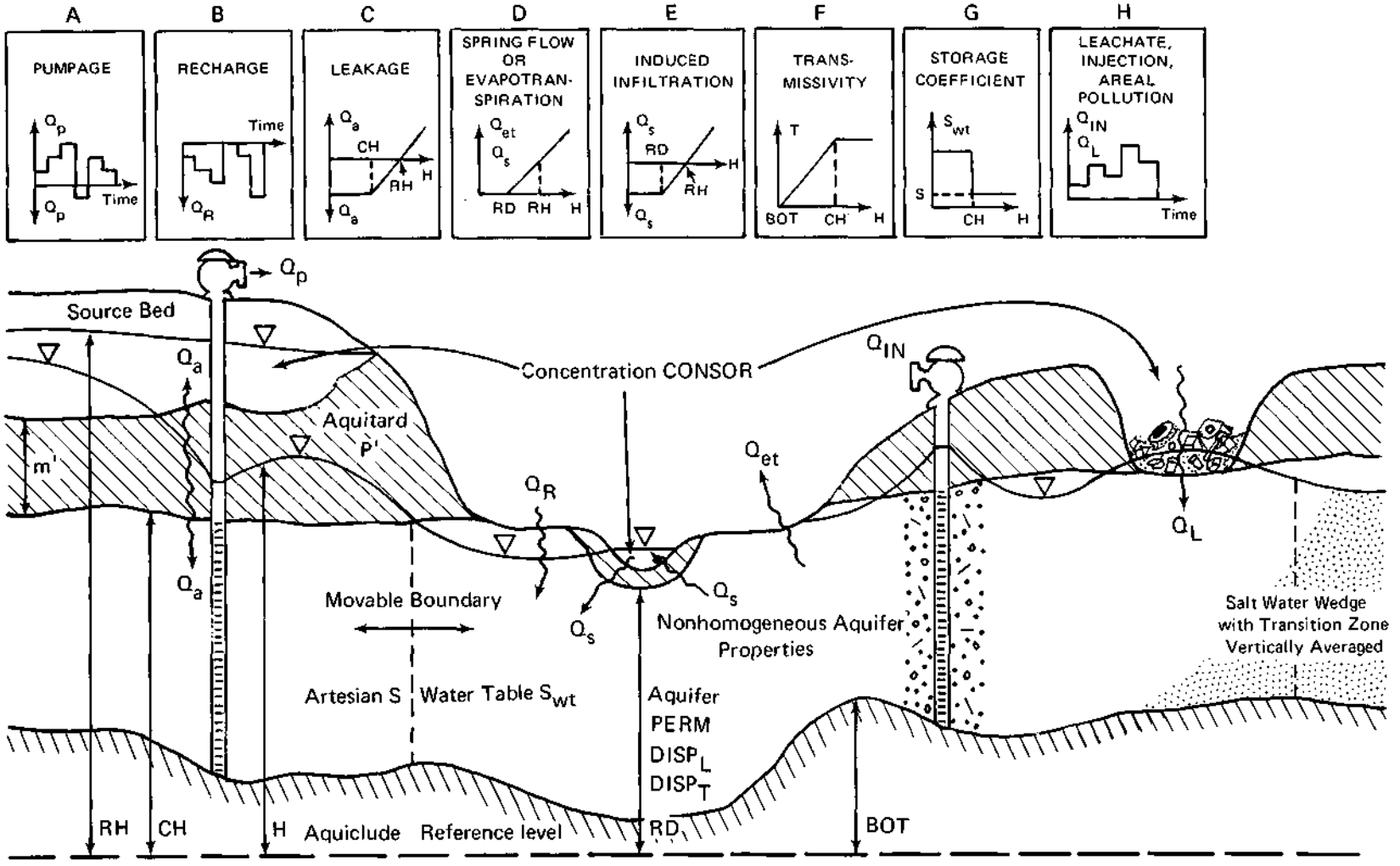


Figure 9. Generalized aquifer cross section showing simulation program parameters

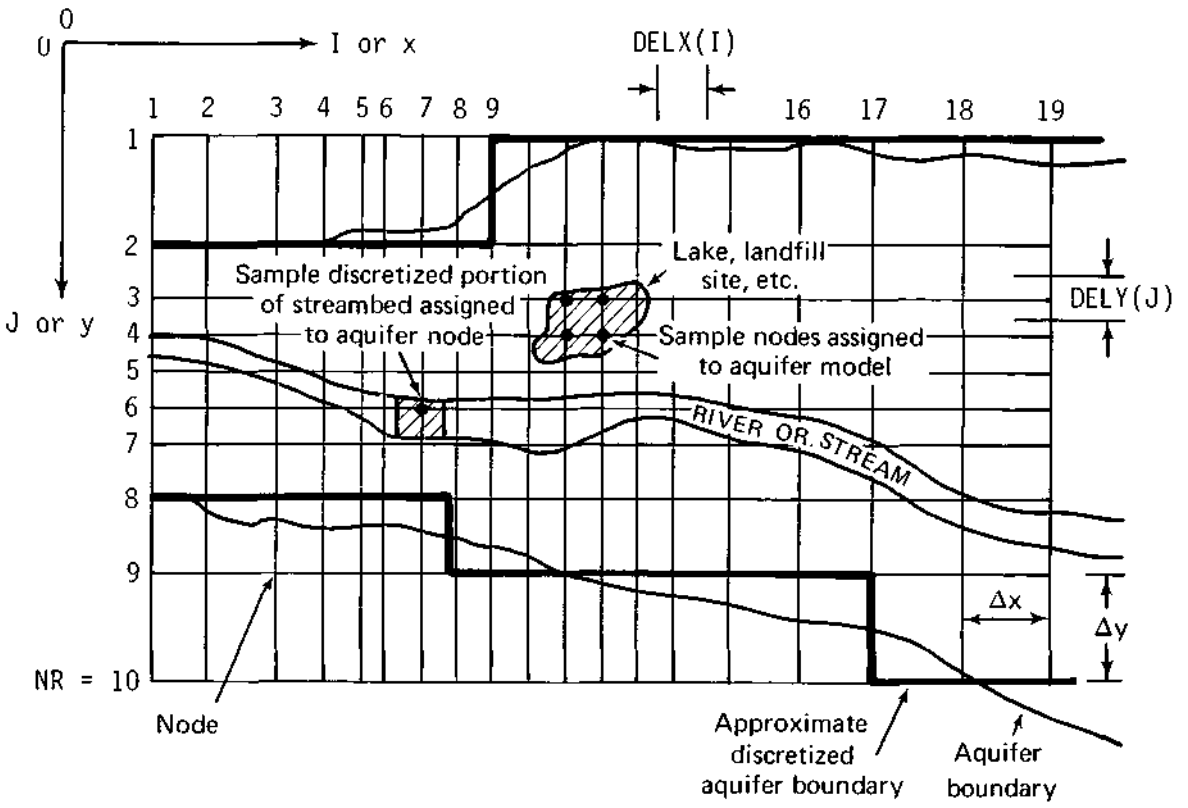


Figure 10. Plan view of finite difference grid over majority of sample aquifer system

The program deck and data decks are assembled in the order illustrated in figure 12. Appropriate control cards are included for the particular computer installation, and the program is ready to run.

Preparation of Data

Parameter Card

Enter numerical values for NSTEP, DELTA, ERROR, and NPITS according to the format given in figure 11A. A general rule for choosing an initial DELTA is to decide at what minimum time drawdowns or heads of interest occur and then precede this time by at least six time increments. Choose an initial value of ERROR from the following formula:

$$ERROR = Q \times DELTA/(10 \times DELX \times DELY \times 7.48 \times s) \quad (14)$$

where

Q = total net withdrawal rate of model, in gpd
DELTA = initial time increment, in days
S = average storage coefficient of model, in gpd/ft
DELX and DELY = typical grid spacing in center of model

The computer program is then "tested" by making a few preliminary runs with different values of ERROR, with a final value being chosen at the point where reduction in that term does not significantly change the solution.

To produce a steady state head distribution, set NSTEPS equal to 1, DELTA equal to 10^{10}, and ERROR equal to 0.1. A water balance subroutine may be added here to further check on the accuracy

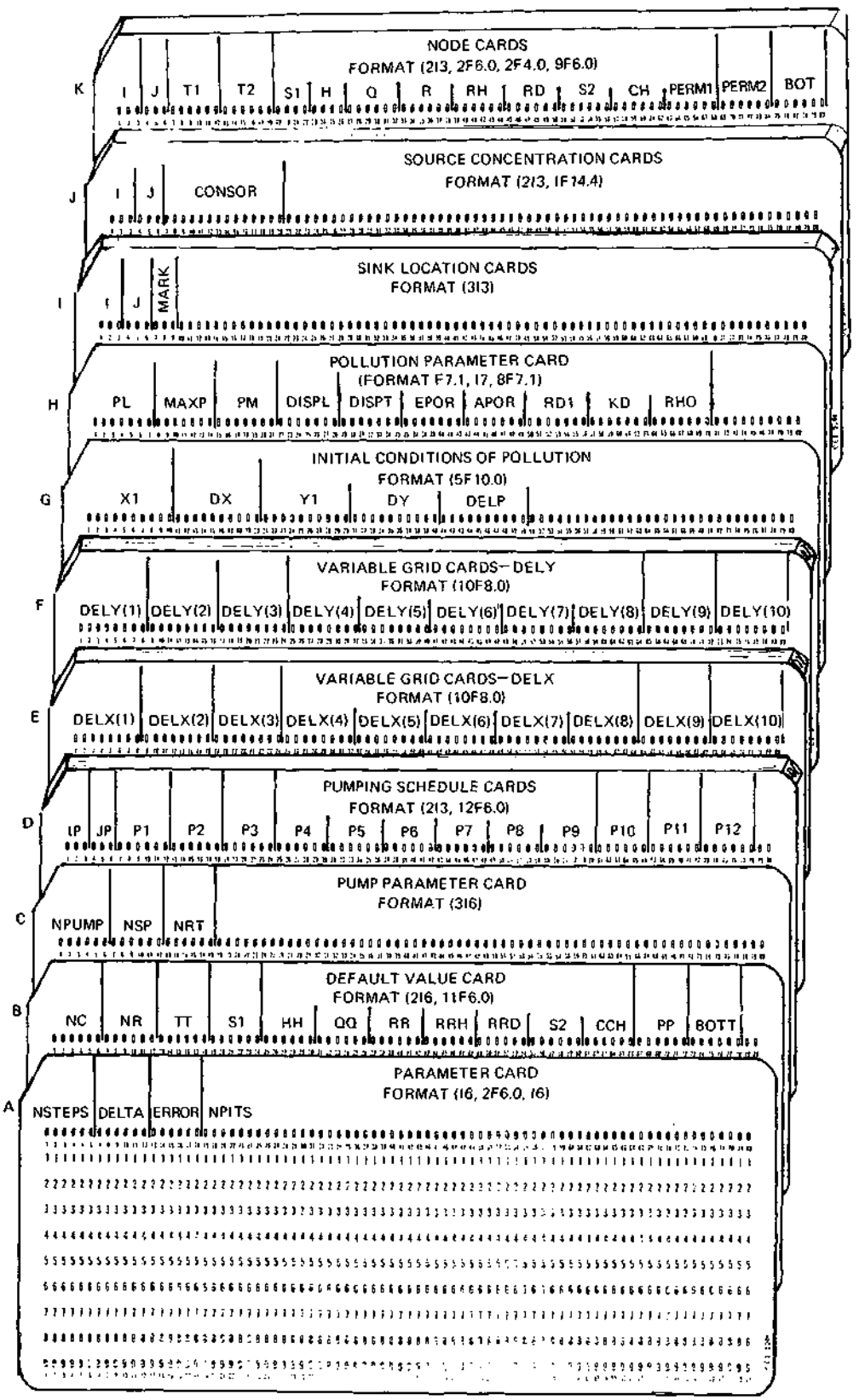


Figure 11. Data deck setup for a mass transport problem

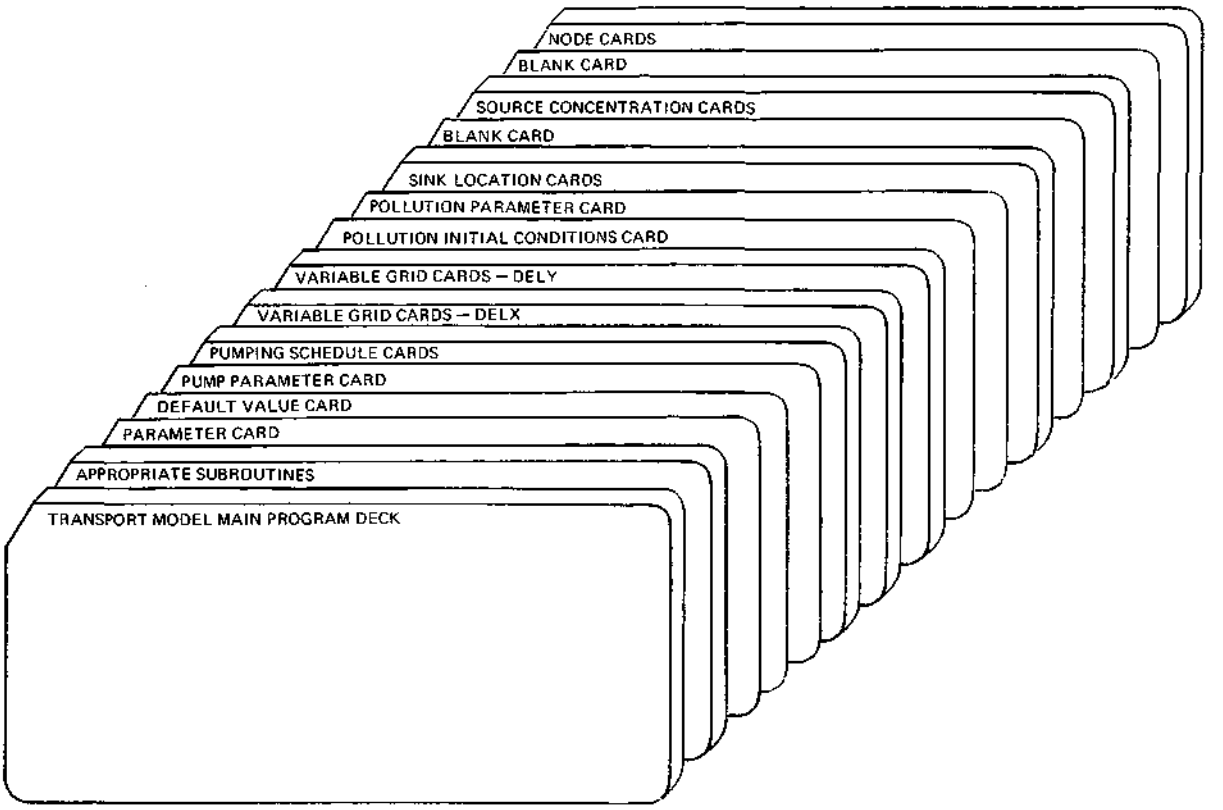


Figure 12. Order of input cards for job setup

of a steady state solution. Otherwise, the solution should be tested by printing out the head distribution every 10 iterations or so to check to see if the heads have reached a steady value.

NPITS is the number of times particles will be advanced in the transport simulation during each DELTA. Therefore NPITS x DELP = total simulation time over which transport occurs in a steady-state flow situation.

Default Value Card

After entering values of NC and NR on the card (line), enter the most commonly occurring values of the aquifer system parameters.

Pump Parameter Card and Pumping Schedule Cards

A stepwise pumping schedule, as illustrated in figure 13, is set up for each well of the model. The computer program can manipulate positive (pumping), negative (recharge), or zero withdrawal rates for the total number of wells (NPUMP). Between changes in pumping rate, the computer program operates with nonuniform time steps within each period of pumping. However, the initial DELTA is reset at each change in pumping rate for the reasons outlined in the nonuniform time increment section of this report.

As an example, figure 13 shows four pumping rate changes (NRT = 4). Each pumping rate is in effect for 11 time increments (NSP = 11), and the total number of time increments is 44 (NSTEPS = 44). The length of time that each pumping rate is effective is the same for all wells.

Once the pumping schedules have been set up, the pump parameter card is included to define NP, NSP, and NRT according to the format shown in figure

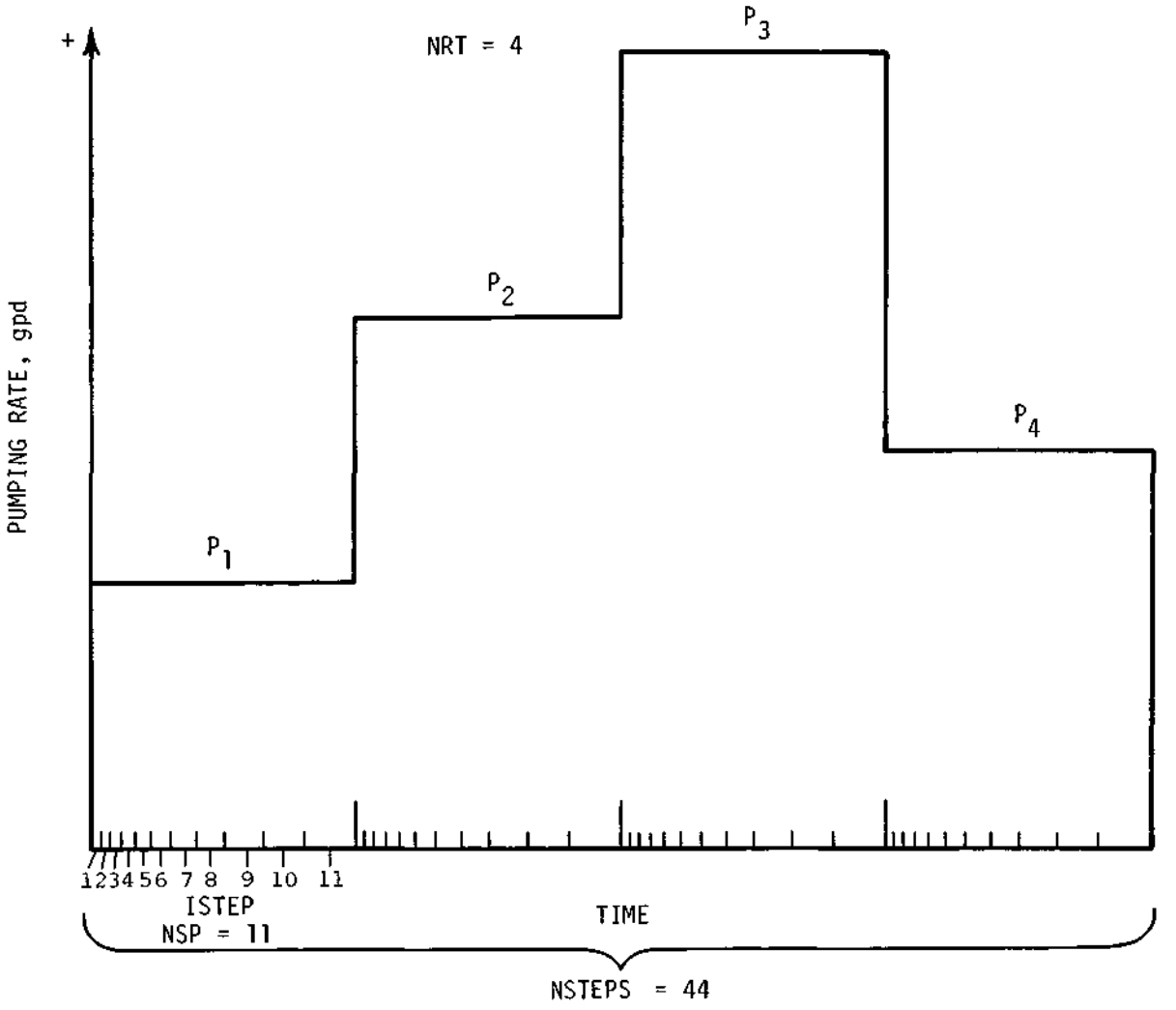


Figure 13. Example variable pumping rate schedule

11C. Next the rate and well location coordinates are entered on pumping schedule cards according to the format shown in figure 11D.

Variable Grid Cards

The varying lengths, DELX(I) and DELY(J), of the finite difference grid, as illustrated in figures 2 and 10, are prepared in accordance with the formats shown in figures HE and 11F. Although it is not necessary, it is suggested that large jumps in grid sizes be held to a minimum to maintain accuracy. It is recommended that length changes from one node to the next be less than double. Having a uniform grid size in the detailed area of interest will also simplify the interpretation of the computer output.

Pollution Initial Conditions Card

There are several mechanisms available in this computer program for generating particles which represent water of various chemical concentrations. There are two principal parameters of concern here: the area and the time period over which the particles spread. Figure 14 illustrates the types of areas for which the computer program can generate particles. The most commonly used area for field problems is the rectangle shown in figure 14C.

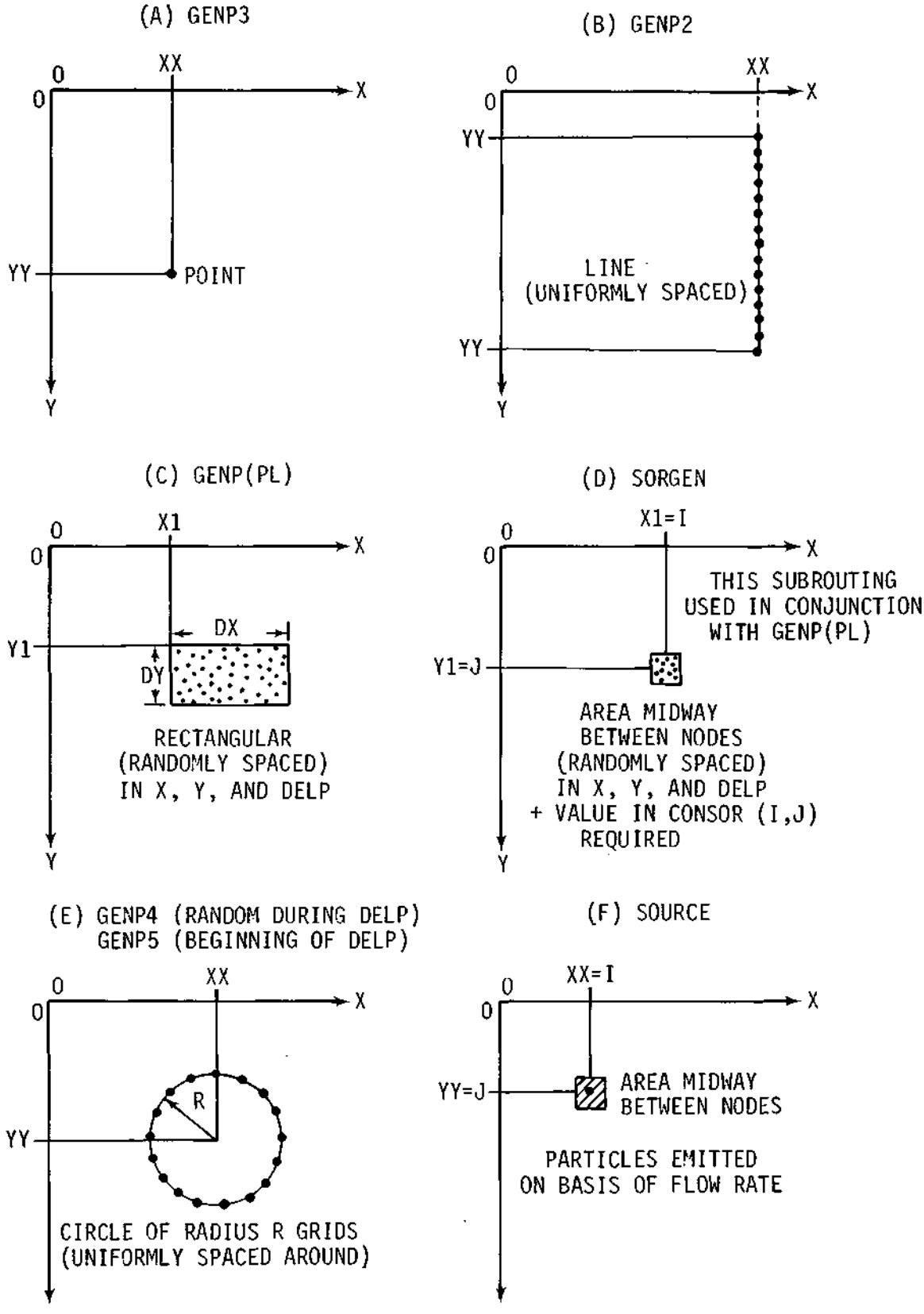


Figure 14. Definition of particle emitting subroutines

Since this type of area would be used most frequently, it is made part of the card input data as opposed to changes being made in some subroutine. The data to be entered on the pollution initial conditions card are thus defined in figure 14C.

At this point, it is essential to realize that in this model the particles are moving on the basis of an x-y coordinate system congruent with and on the same scale as the I-J numbers of the finite difference grid. In other words, if a particle is generated at the location x = 15, y = 4, this particle is at grid coordinates I = 15, J = 4 even though the actual field distance measured from the origin is something else. For instance, refer to figure 10 where the sample discretized area of a streambed is assigned to node I = 7, J = 6. To generate particles within that area you would enter Xl = 6.5, DX = 1.0, Yl = 5.5, and DY = 1.0 on the pollution initial conditions card. Of course, by a change in DX or DY the size of the rectangle can be adjusted from a point (DX = DY = 0) on up to an area the size of the remainder of the model (DX = NC - Xl, DY = NR - Y1).

The last entry on the pollution initial conditions card is the time interval DELP over which particles will be generated. In general, DELP governs the time the particles are allowed to move and goes hand in hand with the time increment DELTA of the flow model.

Further reference to figure 14 will be made later in this report when other ways of generating particle locations will be explained.

Pollution Parameter Card

In most cases, the total pollutant load (PL), which is the total mass of pollutant at 100 percent concentration, is calculated on the basis of the particle mass (PM) and the flow rates or number of particles desired within volumes of the finite difference grid chosen to represent the concentration of the solute. The usual way to proceed is the following: Assume an initial concentration of total dissolved solids (TDS) of 200 mg/l. You suspect that, because of the effects of dispersion and dilution, the concentration will reduce eventually to no more than 1 mg/1. Then, calculate the mass (PM) of a single particle according to the following formula:

$$PM = \frac{\text{Concentration} \times \text{Weight of water}}{10^6 \times \text{Number of particles}} \qquad (15)$$

In terms of the computer model, equation 15 is written

$$PM = \frac{CONC \times (DELX \times DELY \times (H\text{-}BOT) \times APOR \times 62.4)}{10^6 \times NPART} \qquad (16)$$

where

APOR = actual porosity of aquifer
62.4 = pounds per cubic feet of water
CONC= ppm ≈ mg/1
H-BOT = saturated thickness of aquifer
NPART= number of particles representing concentration of TDS in the pertinent volume
DELX, DELY = dimensions of typical finite difference grid
PM= pounds of TDS per particle

Good resolution of the distribution of the TDS as this water moves through the aquifer can be obtained if we arrange to have 200 particles per grid space in the model. (Choice of the number of particles per grid depends upon the resolution desired.) The distribution of the TDS through the aquifer may be obtained from the particle distribution in the model by the following formula:

$$CONC = \frac{PM \times 10^6 \times NPART}{DELX \times DELY \times (H\text{-}BOT) \times APOR \times 62.4} \qquad (17)$$

The computer code has I,J subscripts on all of the parameters of equation 16 except PM and APOR such that the variable nature of the aquifer parameters is accounted for.

The value of PL to be filled in then is the total number of particles wanted times PM. Under most circumstances, PM is the only calculation to be made, and the computer will determine a PL.

The maximum number of particles, MAXP, should seldom exceed 5000. There may be problems that require more, but experience indicates that 5000 is usually adequate.

Values of longitudinal dispersivity, in feet, d_L = DISPL; transverse dispersivity, in feet, d_T = DISPT; effective aquifer porosity EPOR (fraction); actual porosity (APOR); the retardation factor (RD1); the distribution coefficient (KD) for the solute in the particular aquifer material; and the bulk mass density of the aquifer material should then be entered on the pollution parameter card according to the format indicated in figure 11H.

Sink Location Cards

Prepare a card with the I,J coordinate for each sink desired, according to the format of figure 11I. On each of these cards is a location for entering an integer, other than zero, termed MARK. All sinks with the same number in MARK will be treated as a common sink when reporting concentrations. If separate reports on all sinks are wanted, enter different integers on each of the sink location cards in the field MARK.

Source Concentration Cards

Each node of the model that has a source of water with a concentration different from zero needs a source concentration card with the I,J coordinates and concentration CONC, in mg/l, listed according to the format shown in figure 11J.

Node Card Deck

It should be emphasized that the node deck (lines) contains one card (line) for each node that has any aquifer system properties differing from those defined on the default value card (line). If a node card (line) is included, all values must be punched on it even if some of the values are equal to the default values. It is essential to have a copy of the report of Prickett and Lonnquist (1971) to fully understand the details of setting up the flow model.

Barrier Boundary Conditions. Barrier boundaries are simulated by setting PERM (I,J,1) and PERM (I,J,2) equal to zero on individual node cards.

Leaky Artesian Conditions. For leaky artesian conditions calculate the recharge factors ($R_{i,j}$) in gallons per day per foot (gpd/ft) from the following formula:

$$R_{i,j} = (P'/m')\Delta x \Delta y$$

where

P' = vertical hydraulic conductivity of confining bed, in gallons per day per square foot (gpd/ft^2)

m' = thickness of confining bed, in feet

The recharge factor defines the slope of the line given in figure 9C.

Appropriate source bed heads (RH), elevations of the top of the aquifer (CH), and recharge factors are entered on the default value card, and any differing values are entered on the node cards.

Induced Infiltration. For induced infiltration calculate the recharge factors (in gpd/ft) from the following formula:

$$R_{i,j} = (P'/m')A_s$$

where

P' = hydraulic conductivity of the streambed, in gpd/ft^2

m' = thickness of streambed, in feet

A_s = fraction of the area of the streambed assigned to full area node, a fraction

The recharge factor defines the slope of the line shown in figure 9E.

Appropriate values of stream water surface elevations (RH), elevations of the bottom of the streambed (RD), and recharge factors calculated from the above equation are entered on node cards.

Constant Head Conditions. If a constant head is desired in the model, and the concentration of the water flowing into that node to maintain the head is of zero concentration, then set SFI(I,J) at that node equal to 10^{30}. If the concentration is other than zero, set R(I,J) equal to 10^{10} for that node card. This is done to maintain a small resistance to flow from the source to the aquifer enabling the flow rate to be measured.

Storage Factors. In the program described by Prickett and Lonnquist (1971), storage factors were entered on the node cards. This is not necessary here. Just enter the artesian and water table storage coefficients directly into SF1(I,J) and SF2 (I,J) positions on the node and default value cards.

Evapotranspiration. Evapotranspiration is defined by the slope of the line shown in figure 9D and is calculated from field data (Prickett and Lonnquist, 1971). The value of the slope of that line is then entered on node cards in the space reserved for the recharge factor. Set values of both RD and RH equal to one another. RD is defined as the elevation of the water table below which the effects of evapotranspiration cease.

Flow from Springs. Determine a recharge factor (slope of the line shown in figure 9D) from field data of flow versus head changes in the vicinity of the spring. A recharge factor may also be found empirically by matching simulated with observed spring flows. The elevation at which water flows from the spring is recorded in both RH and RD.

PART 3. PROGRAM OPERATIONAL SEQUENCES

The basic operation of the computer program is explained in the following discussion according to the listing given in figure 8 and the flow charts and associated explanations given in figures 15 through 36.

The computer program consists of a MAIN section, 20 subroutines, and 3 functions. As illustrated in figure 15, it first reads and writes all input data. After this, the MAIN program calls a sequence of subroutines, as needed, that 1) solves for the heads for a particular time increment DELTA; 2) prints the heads; 3) generates particles and inserts them in the flow; 4) allows the particles to move a time step DELP; and 5) prints maps and summaries of particle locations and concentration distributions. The sequence of subroutines called by the MAIN program can be changed according to the particular flow and mass transport problem under study. After the desired sequence of subroutines is called, the MAIN program cycles to the second time

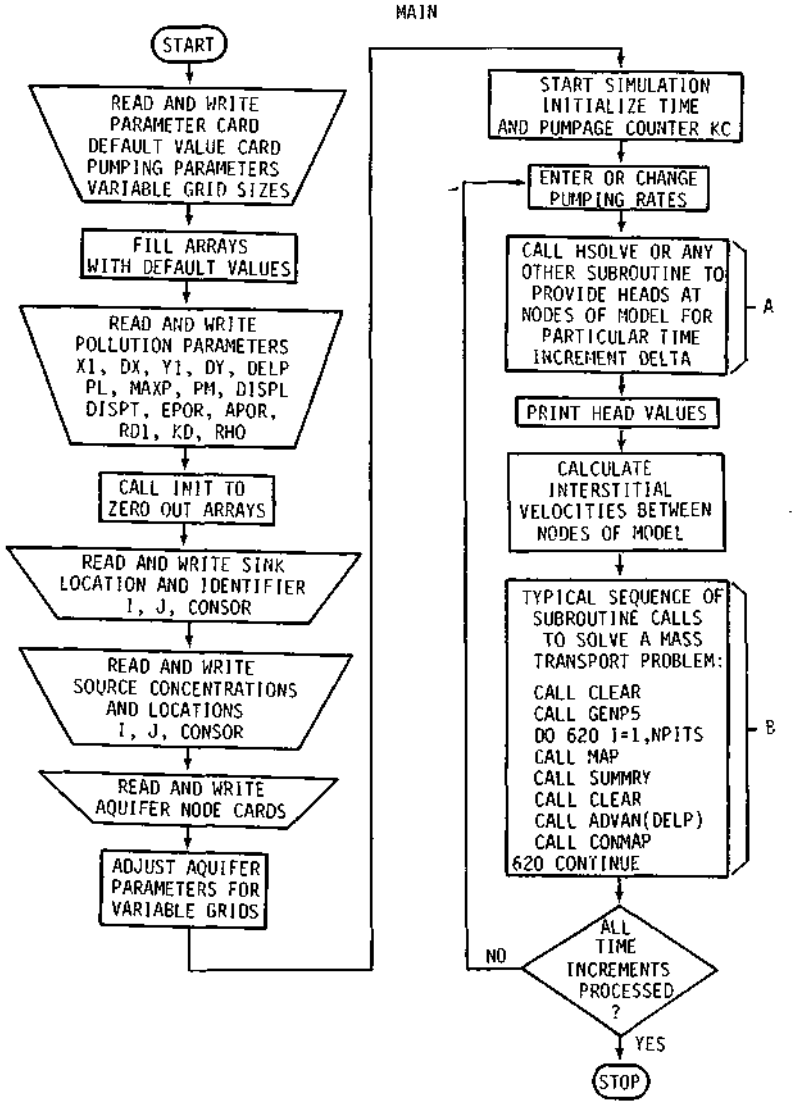


Figure 15. Flow chart for MAIN program

increment of the flow model and performs the above 5 steps repeatedly until all time increments are processed.

There are 5 required subroutines and 3 required functions. There are also 15 optional subroutines to choose from, depending on the transport problem of interest. Of these, there are 3 subroutines for producing the head distribution for the flow model, and 7 subroutines that generate different configurations and numbers of particles. There is 1 subroutine to calculate retardation, and 4 subroutines that print out various maps of particle location, concentration distributions, and summaries of concentration of water flowing into sinks. Table 1 provides brief descriptions of the subroutines and functions. The details on how to mix and match the various subroutines will be given in the example problems section (Part 4).

Required Subroutines and Functions

Function ANORM(0)

Figure 16 gives the flow chart for this function. This function produces a number between -6 and +6 from a normal distribution of numbers having a mean of zero and a standard deviation of one. In figure 8 it can be seen that this function uses yet another function called RANF(O), which is a CDC system subroutine that produces a random number between 0 and 1. If other systems are used, an appropriate substitute random number generator that produces numbers in the same range can be used.

Function V(I,J,K)

As shown in figure 17, whenever calls are made to this function, it is the node-to-node interstitial velocity that is supplied. These velocities are the basis for which an elaborate interpolation scheme is applied to provide velocities anywhere in x-y space.

Subroutine INIT

This subroutine could just as easily have been placed as a part of MAIN. However, by its nature it can be useful as a subroutine for some special applications. As shown in figure 18, this subroutine merely zeros out the TMAP and NP variables and the SOR(I,J), CONSOR(I,J), and MARK(I,J) arrays.

Subroutine CLEAR

As illustrated in the flow chart in figure 19, this subroutine zeros out the NPART(I,J) and TABLE(I) arrays. This is like erasing the blackboard in preparation for keeping track of particle locations each time particles are going to be moved.

Subroutine ADD(XX,YY)

Any time that a new particle is added to the simulation, you must call ADD(XX, YY) and supply the XX, YY coordinates of the particle that you want added. As indicated in figure 20, the particle counter NP is incremented and, as long as NP is less than the maximum number of particles allowed MAXP, the particle XX, YY coordinates are stored in the X and Y arrays. In the event that you are attempting to add particles beyond the limits of the X and Y arrays, the code is written in such a way as to eliminate an addressing error. For many cases, you don't know how many particles will be needed for your simulation since, for instance, the rate of flow from a nearby pollution source may be unknown. In this case, the flow may be sufficient to require a number of particles far in excess of the dimensions of the X and Y arrays. If this happens, the program is written to randomly remove a particle somewhere in the model and make room for the new one. It may be that even the new particle itself gets thrown out. In any case the particle mass (PM) of all other particles is then proportionately increased to maintain conservation-of-mass principles. The variables NP and PM should be checked as the simulation progresses to

Table 1. Srief Descriptions of Subroutines and Functions

ADD(XX,YY):	add single particle at coordinates (XX,YY)
ADVAN(DELP):	advances all particles DELP days
ANORM(O):	produces a single number from a normal distribution of mean 0, σ=1, range +6 to -6
CLEAR:	clears arrays NPART and TABLE
CONMAP:	prints concentrations at all nodes of model
GENP(PL) :	produces PL/PM particles randomly in a rectangle Xl, Yl.DX,DY and randomly in time increment DELP
GENP2:	generates 29 particles along column 30 of model, 1 particle per node
GENP3:	generates 51 particles, all at coordinate XX = 1, YY = 2
GENP4:	generates 360 particles around an R = 0.7 circle with center at coordinates (15,15), randomly during DELP
GENP5:	generates 101 particles around an R = 0.7 circle with center at coordinates (15,15), at the onset of the call
HSOLVE:	ISVVS Bulletin 55 (Prickett and Lonnquist, 1971), composite program
HSOLV2:	calculates linear heads per grid left to right
HSOLV4:	Thiem equation, radial flow from an injection well at coordinates (15,15)
INIT:	initializes or zeros out arrays TMAP, NP, ANC, ANR, SOR, CONSOR, and MARK
MAP:	prints number of particles residing in each zone for whole model
MOVE(XX, YY, DEL) :	moves particles and tabulates in which zone particle resides and removes particles when captured by a sink
RDSOLV(EPOR,RHO,KD,RDl):	calculates retardation factor from input data
SNKCON:	for time DELP, prints concentration at sinks specified by MARK
SORGEN:	produces particles based on either head dependent flow rates or injection wells randomly in space in time during DELP
SOURCE:	produces particles when particle mass has sufficiently accumulated from any source
SUMMRY:	prints out number of particles captured by a sink during time DELP
V:	provides proper velocities to subroutine VELO
VELO(XX, YY, VX, VY) :	calculates interpolated velocities for particle movement

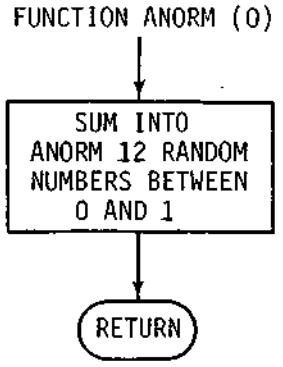


Figure 16. Flow chart for Function ANORM(0)

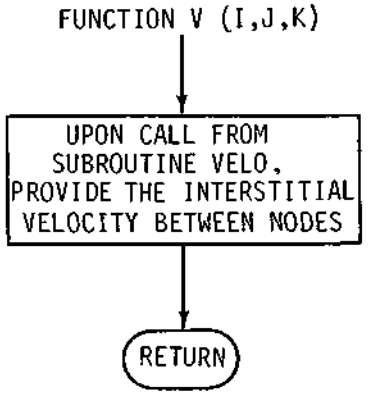


Figure 17. Flow chart for Function V(I,J,K)

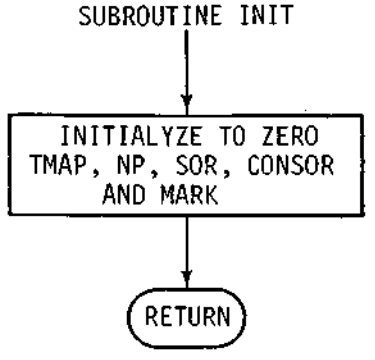


Figure 18. Flow chart for Subroutine INIT

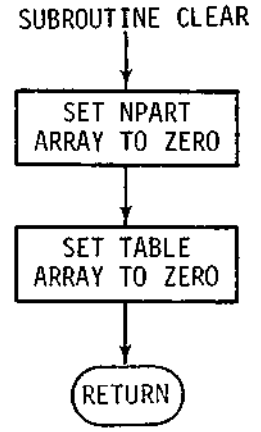


Figure 19. Flow chart for Subroutine CLEAR

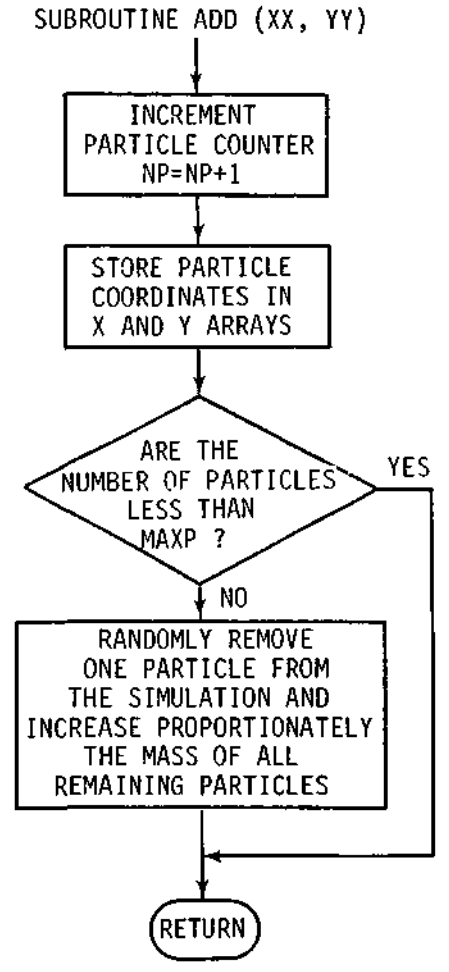


Figure 20. Flow chart for Subroutine ADD(XX,YY)

see if this is happening. If it is, back up the simulation and make necessary changes such that PM stays constant. Suitable changes might be to increase PM or MAXP and the X,Y array dimensions.

Subroutine VELO(XX,YY,VX,VY)

This subroutine (see figure 21) provides velocity vectors for the particles as they move anywhere within the bounds of the model. It is very important that an accurate scheme be employed to produce a continuous velocity field from discrete velocities provided from Function V(I,J,K). One of the main drawbacks of the MOC is its inability to handle the strongly divergent problem. For instance, in the vicinity of injection wells the flow is entirely divergent away from the center of the well.

This divergent flow problem is avoided in the code by borrowing an idea from the finite element method. A set of Chapeau functions is defined for the purpose of interpolating velocities anywhere in the model on the basis of the discrete velocities between nodes. The explanation of how these Chapeau functions work is shown in figures 22 through 24.

The interpolation of velocities is a three-step procedure. The first interpolation involves the use of the four inter-node velocities in the immediate vicinity of the x, y position of the particle, as illustrated in figure 22. The second step of the interpolation is illustrated in figure 23, where the next closest four inter-node velocities are used. The third interpolation is shown in figure 24, where the final velocity vectors VX and VY are calculated.

In the event that the particle is near the model boundaries, the vectors used for interpolation are as shown in figure 25.

The result of this interpolation scheme is a continuous velocity field that provides accurate velocities even for the strongly divergent problem (see figure 42, for example).

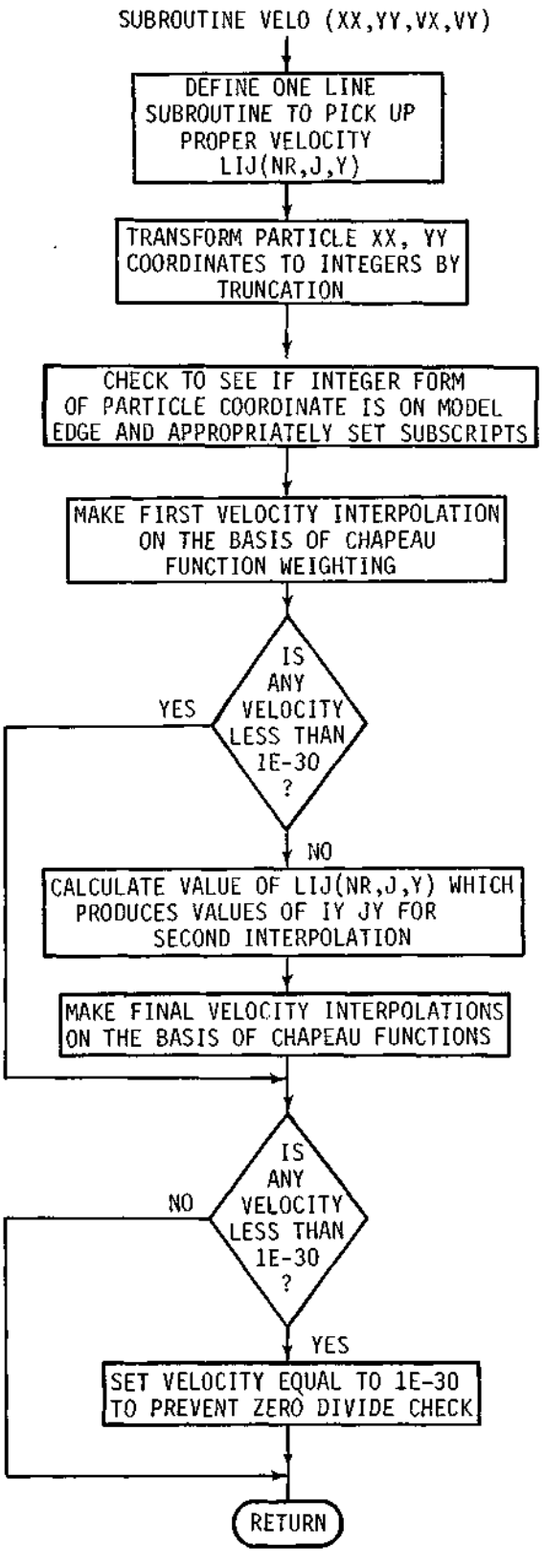
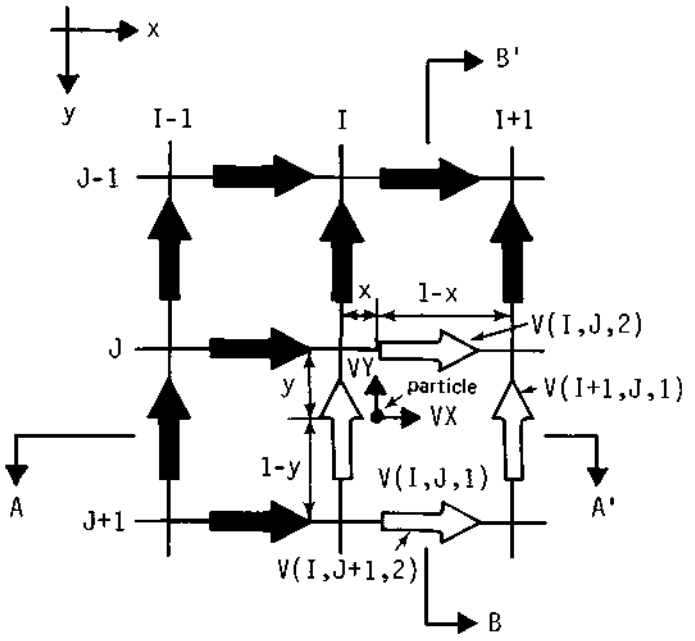
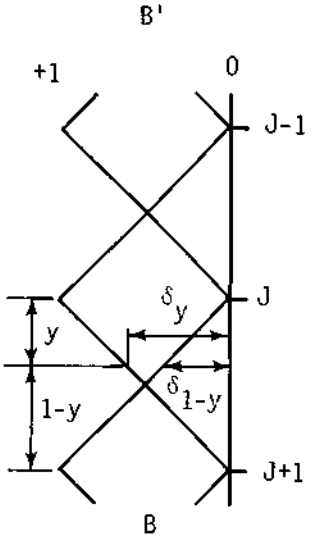
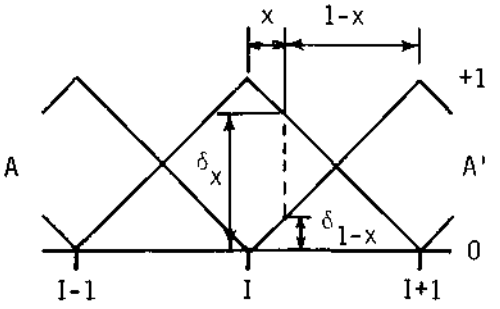


Figure 21. Flow chart for Subroutine VELO(XX,YY,VX,VY)





CHAPEAU FUNCTIONS FOR VELOCITY VECTORS BETWEEN COLUMNS I AND I+1



CHAPEAU FUNCTIONS FOR VELOCITY VECTORS BETWEEN ROWS J AND J+1

$VY = \delta_x V(I,J,1) + \delta_{1-x} V(I+1,J,1)$

since:

$\delta_x = 1-x$

$\delta_{1-x} = x$

$VY = (1-x)V(I,J,1) + xV(I+1,J,1)$

similarily:

$VX = (1-y)V(I,J,2) + yV(I,J+1,2)$

Figure 22. First step in calculating x and y velocity vectors for particle

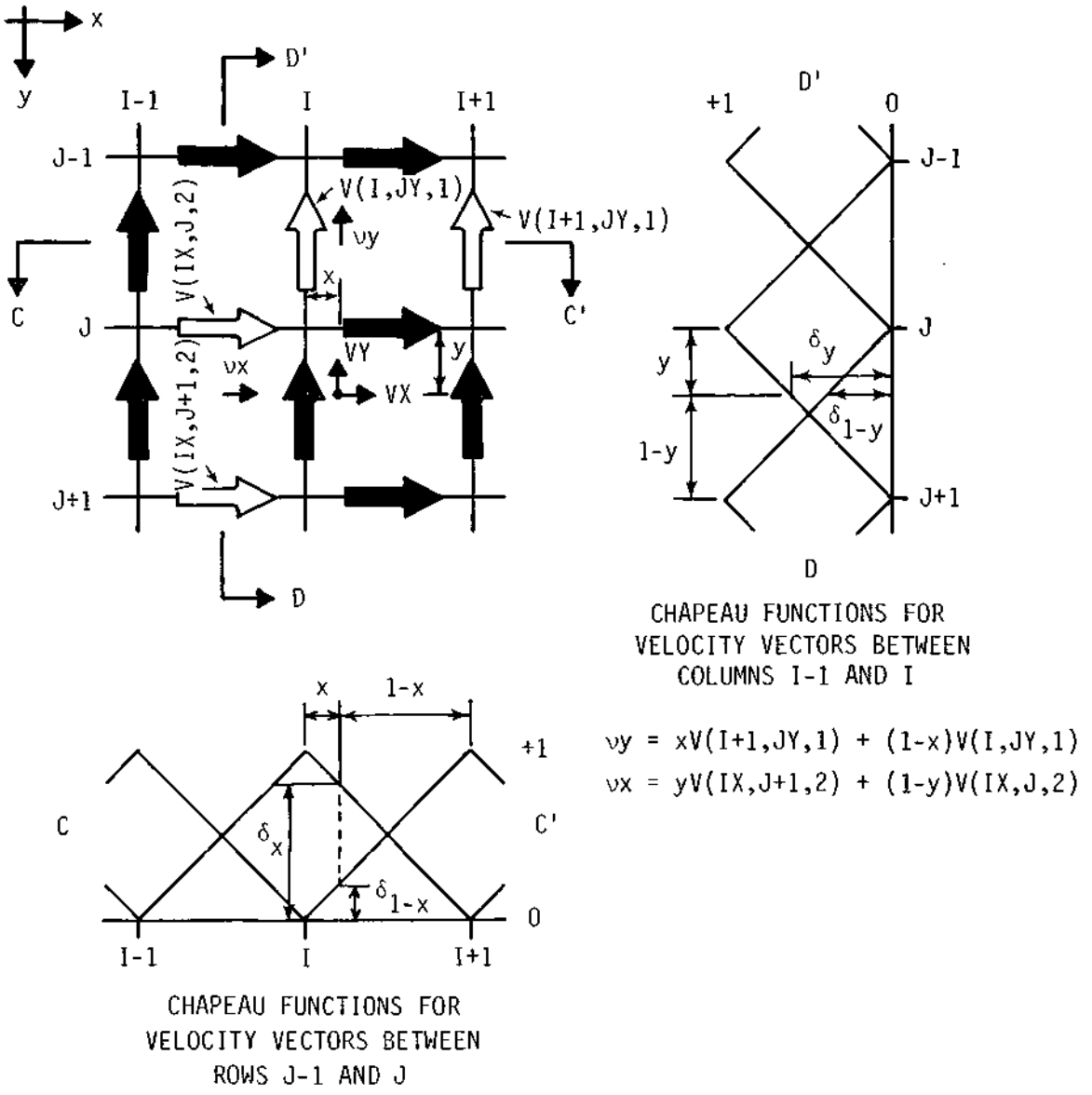


Figure 23. Second step in calculating x and y velocity vectors for particle

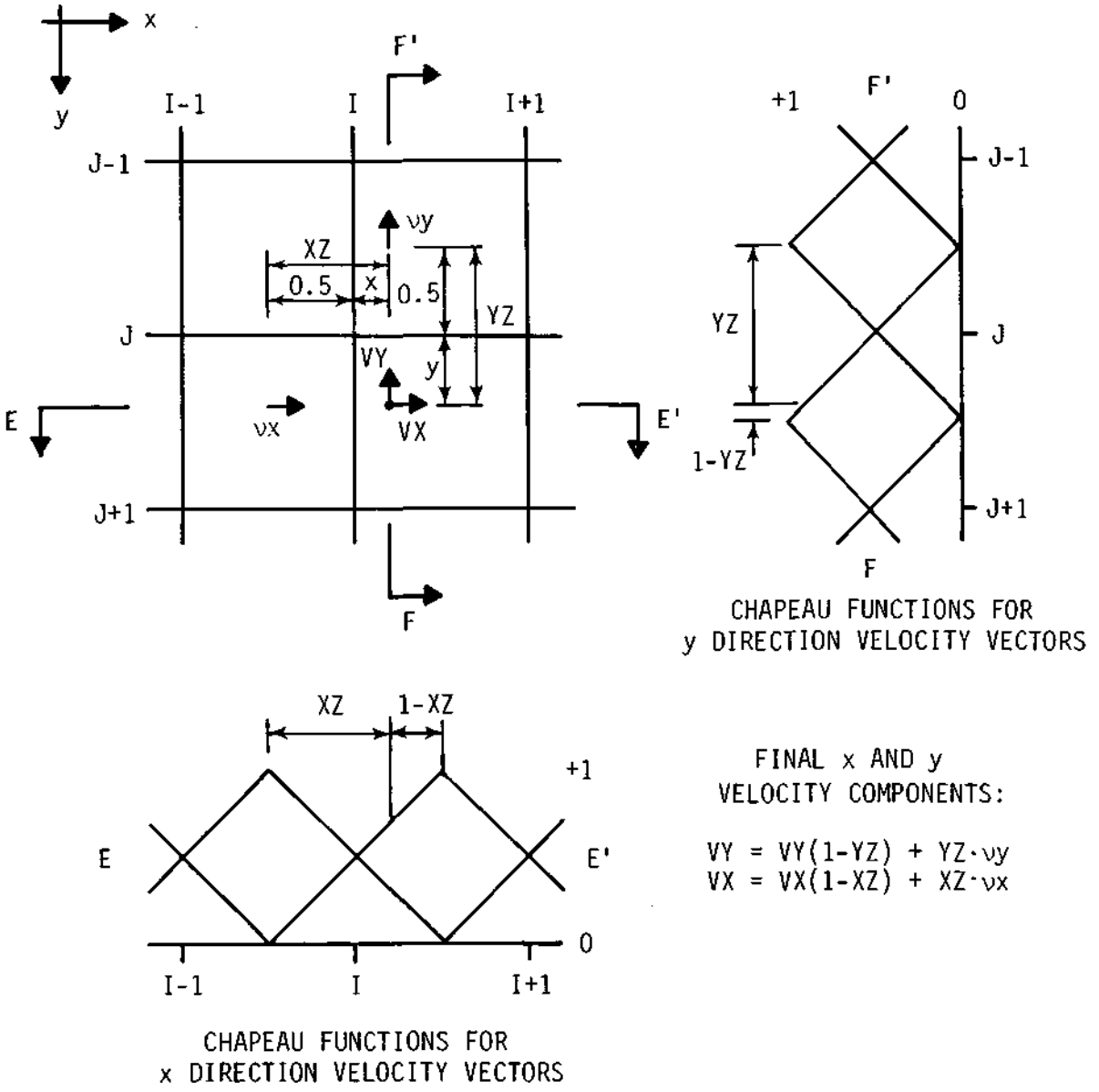


Figure 24. Third step in calculating x and y velocity vectors for particle

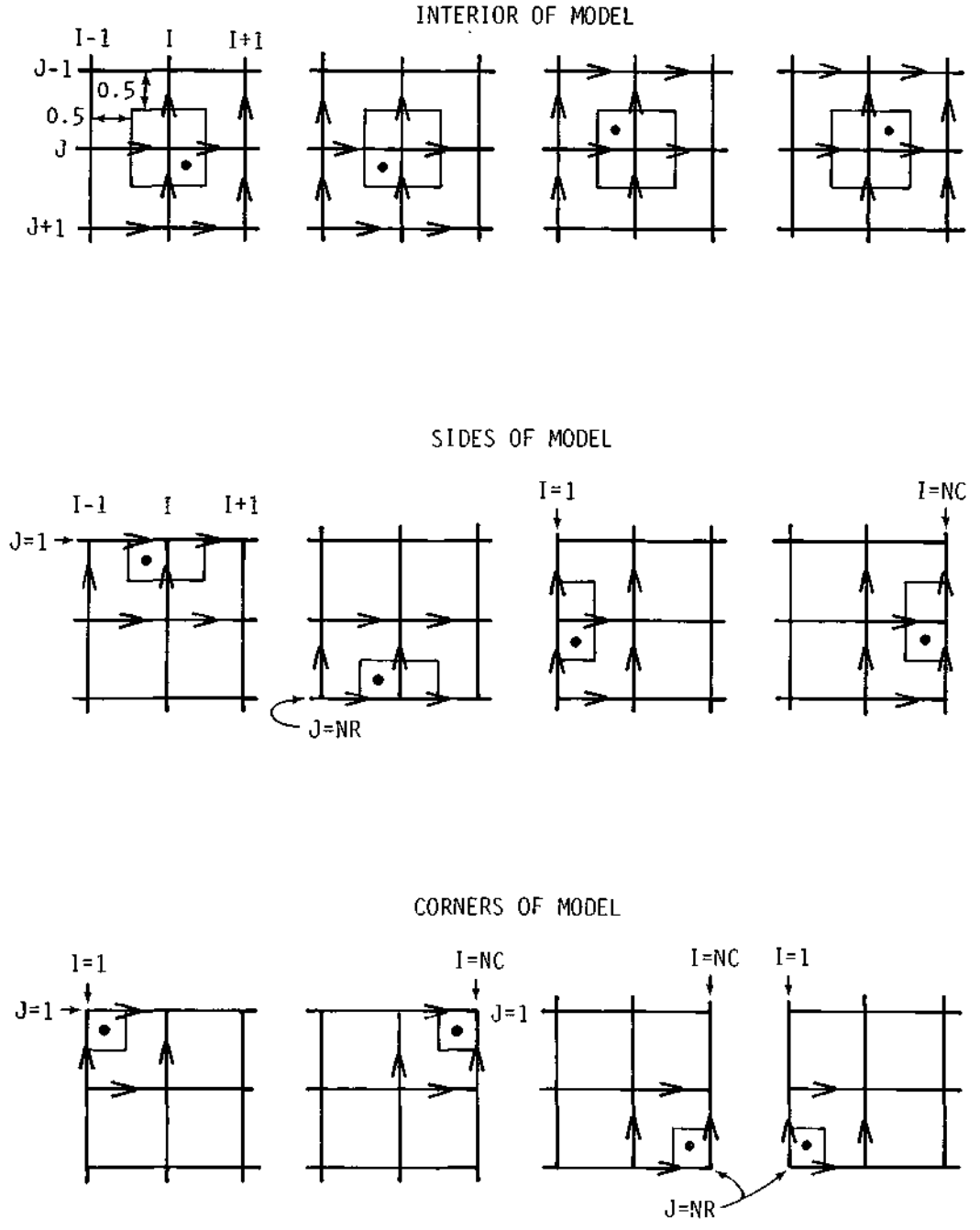


Figure 25. Node to node velocity vectors used in interpolation scheme

Subroutine ADVAN(DELP)

This subroutine controls and advances all particles presently in the flow field and does some bookkeeping on particle x, y coordinate locations as some particles disappear in sinks. The flow chart in figure 26 indicates the sequence of events. The particles are all allowed to move for a time period defined by DELP. For most cases DELP is equal to the flow model time increment DELTA.

However, if it is desired to move the particles more often between DELTA time steps, call ADVAN(DELP) more often and such that the sum of the DELPs equals DELTA. Steady state flow problems are an example, as illustrated in the examples section of this report.

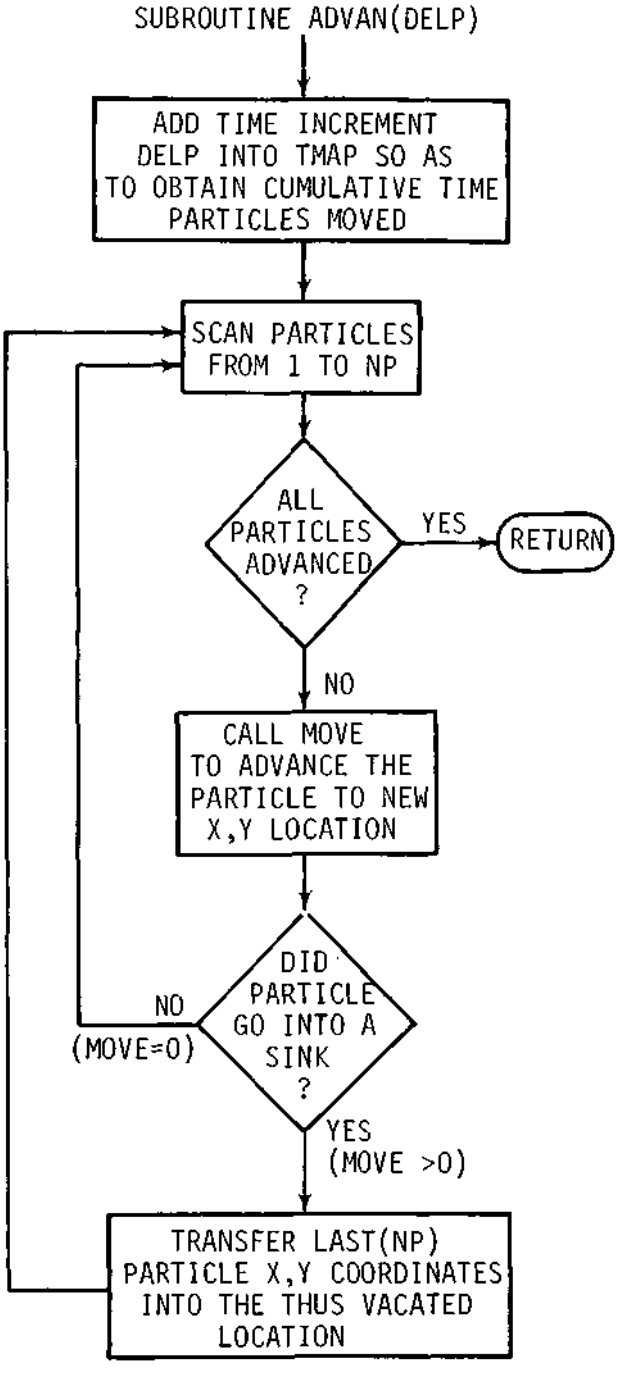


Figure 26. Flow chart for Subroutine ADVAN(DELP)

Function MOVE(XX,YY,DEL)

Every time that this function is called, a particle is moved. Calls will be coming from subrouting ADVAN (DELP) and certain particle generation. As mentioned previously, the particles move on the basis of the I,J coordinate system numbers. The flow chart for function MOVE is given in figure 27 and should be studied along with the listing in figure 8.

The function MOVE moves a particle on the basis of its velocity vectors as calculated from subrouting VELO. New velocity vectors are computed at least 5 times during the time a particle is moving across 1 grid of the model. This is done to maintain high accuracy. There are cases where this restriction can be relaxed, and some cases where it can be tightened up. However, 5 times is a good number for most problems. It is advised that the flow chart in figure 27, the listing in figure 8, and the vector algebra in figure 6 be studied to see how the sequence of events occurs when the function MOVE is called.

The function MOVE performs two other tasks besides moving a particle. First, it checks to see if the particle goes into a sink; i.e., within half a grid of MARK(I,J). If so, the mass of the particle is tabulated in the TABLE(I) array. Secondly, after the particle moves its assigned DELP time, if it doesn't go into a sink, its x,y coordinates are used to tabulate its position

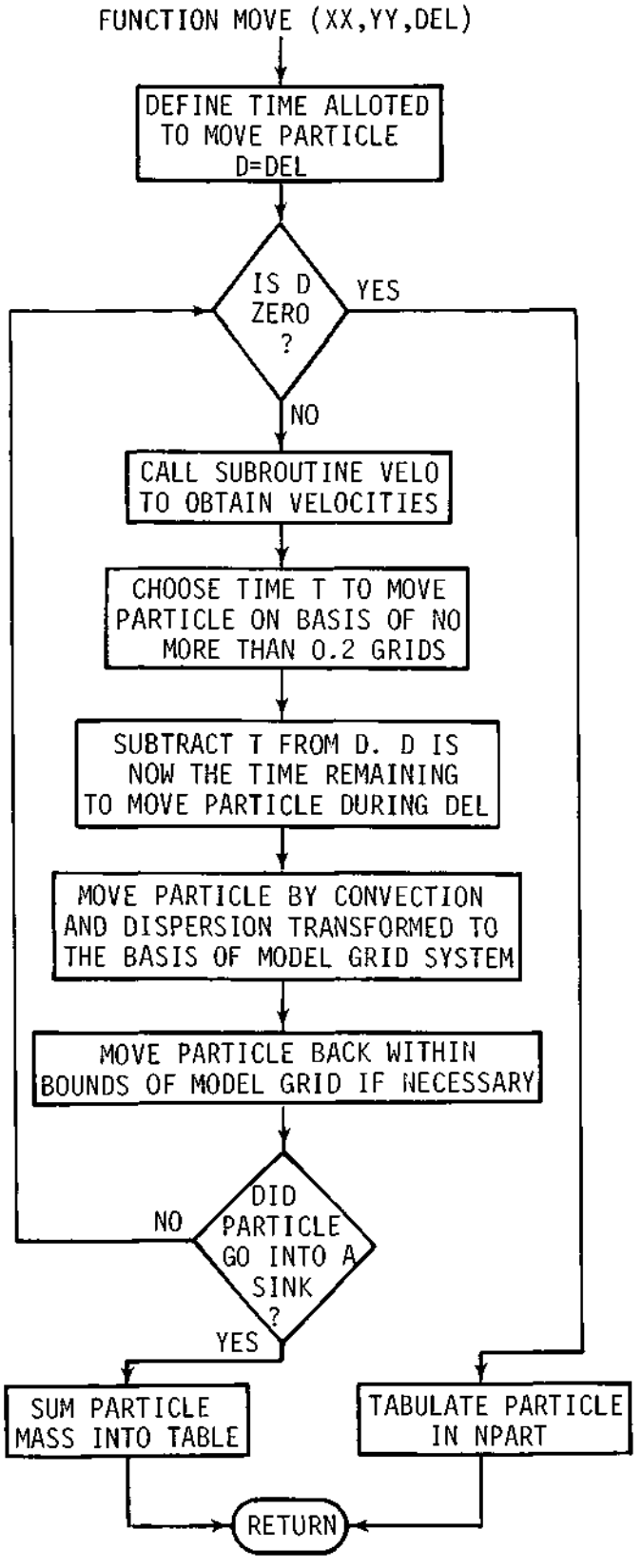


Figure 27. Flow chart for Function MOVE(XX, YY,DEL)

in the NPART (I,J) array. The NPART and TABLE arrays are printed when a map of particle positions and sink histories is desired.

Optional Subroutines That Produce Head Distributions

A set of heads must be obtained for all nodes of the flow model from which to compute the velocity field. The three subroutines discussed here can be used to fill in the head array H(I,J), However, it should be realized that almost any means can be used to do this. A new code could be written, someone else's code could be used, or field data for the head array could be inserted. The three following methods are just examples.

Subroutine HSOLVE

This subroutine comes directly from SWS Bulletin 55 (Prickett and Lonnquist, 1971) and is a portion of their "Composite Aquifer Simulation Program." The section of Part 2 of this report entitled "Job Setup" is based upon using this subroutine as the means to fill in the H(I,J) array. The flow chart in figure 28 is thus abbreviated. A copy of SWS Bulletin 55 is needed to fully understand this. The "Job Setup" section of this report, however, is adequate for a start.

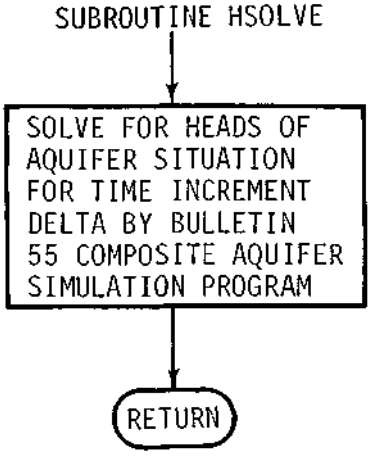


Figure 28. Flow chart for Subroutine HSOLVE

Subroutine HSOLV2

This subroutine calculates heads as a function of column number such that a uniform flow of 1 foot per day in the x direction is realized. Refer to the listing in figure 8 and the flow chart in figure 29 for the details.

This is one example of generating a head distribution directly without the need of solving some complex matrix problem associated with finite difference formulation.

Subroutine HS0LV4

This subroutine fills the H(I,J) array with a solution to the steady-state Theim formula with the center of the well at x = 15 and y = 15. The flow chart in figure 30 and the subrouting code in figure 8 show how this was done. The idea here is to waste no time in producing a head distribution when you know exactly what you want.

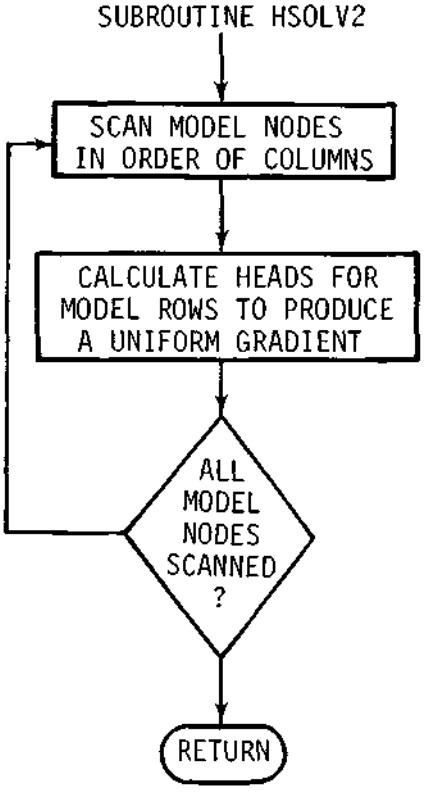


Figure 29. Flow chart for Subroutine HSOLV2

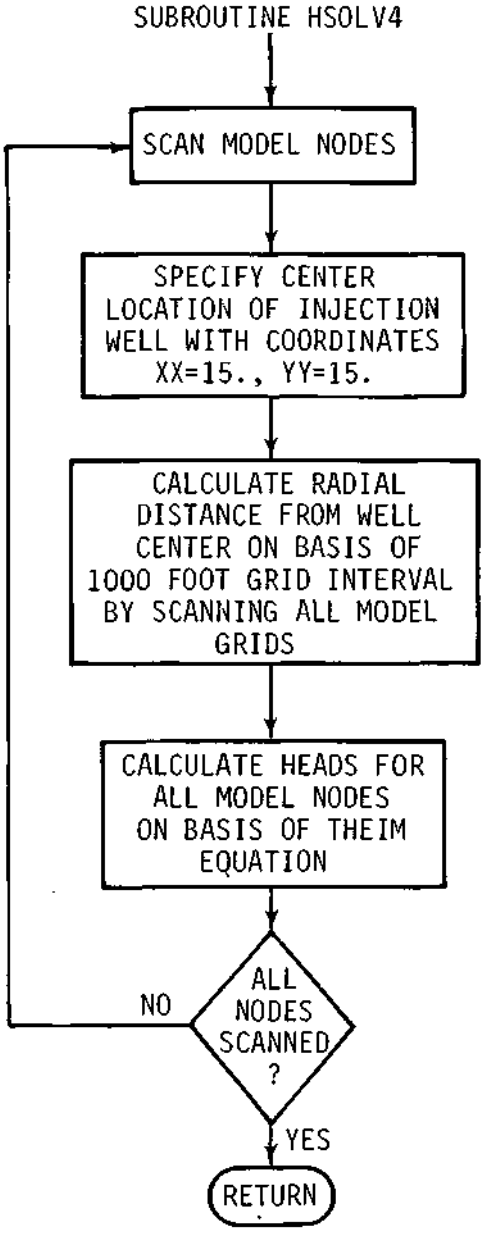


Figure 30. Flow chart for Subroutine HSOLV4

Optional Subroutines That Generate Particles

Seven different subroutines are included as examples of ways particles can be generated in various configurations. Configurations available are points, lines, rectangular areas, and circles. Throughout the following discussion, repeated reference will be

made to figure 14, as this figure indicates the needed input data for generating particles for the various subroutines.

In general, the particle generating subroutines merely generate x, y coordinates in various configurations and then call the subroutine ADD(XX,YY) to get the particle into the flow.

Subroutine GENP2

The flow chart in figure 31, the listing in figure 8, and the illustrated definition in figure 14B shows how to place one particle at each node of the flow model in a line. Again x, y coordinates are generated and then the subroutine ADD(XX,YY) is called.

Subroutine GENP3

Figure 14A and the listing in figure 8 illustrate how one or more particles can be placed at a particular point in the model at one instant of time. The flow chart is given in figure 32. Suitable changes in the xx, yy coordinates of this subroutine could enable the generation of particles at any point in the model.

Subroutine GENP(PL)

This subroutine generates any number of particles randomly located within a rectangular area of configuration given by figure 14C. The necessary Xl, DX, Yl, DY, DELP coordinates may be specified by the initial pollution conditions card or by a call from another subroutine such as SORGEN. The total pollutant load PL can also be specified in the pollution parameter card or by a call to another subroutine such as

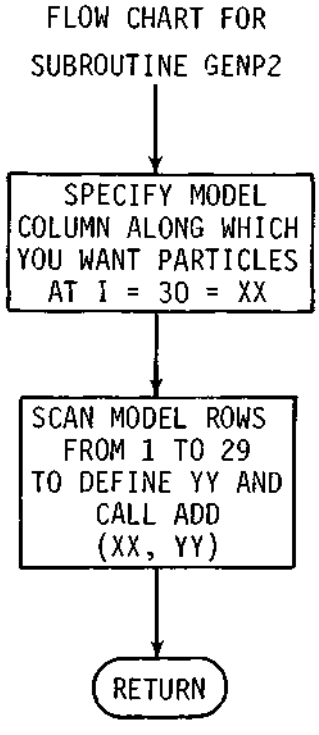


Figure 31. Flow chart for Subroutine GENP2

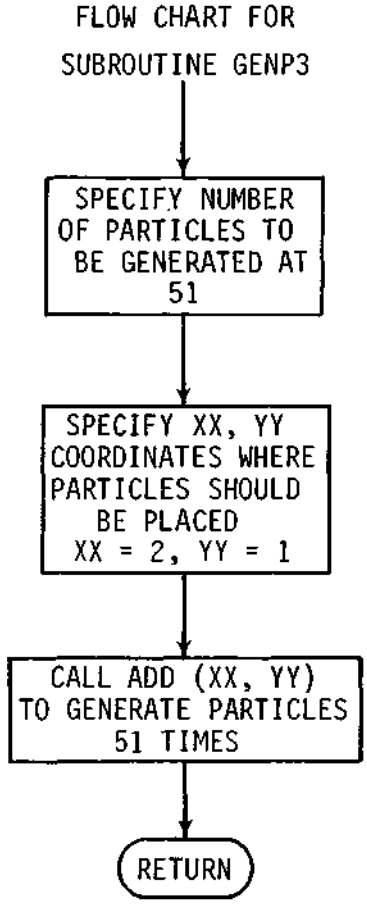


Figure 32. Flow chart for Subroutine GENP3

SORGEN. The idea here is that once PL is supplied, the subroutine breaks up the total into parts PL/PM and spreads particles randomly in the rectangular area specified and randomly over the time increment DELP specified until the total PL is accounted for. The listing in figure 8 and the flow chart in figure 33 give more details.

Subroutines GENP4 and GENP5

The flow chart and listing in figures 34 and 8, respectively, illustrate how a circle of particles is generated according to the configuration shown in figure 14E. Here again, the x and y coordinates are generated and ADD(XX, YY) is called to create the particles. GENP5 generates 101 particles in a circle uniformly spaced at R = 0.7 at the beginning of the time increment DELP. Subroutine GENP4 is the same as GENP5 except that 360 particles are generated randomly during the time increment DELP. The listing in figure 8 shows that this is done by calling RANF.

Subroutine SORGEN

This subroutine is similar to SOURCE in that particles are emitted on the basis of flow rates from sources having a concentration different than zero. A value in the array CONSOR(I,J) is all that is necessary to actuate a particle when this subroutine is called. Both subroutines SORGEN and SOURCE calculate the total mass (PL) involved in the water flowing from the various sources of the model on the basis of the specified particle mass (PM) and the flow rate. As shown in the listing in figure 8, the flow chart in figure 35, and the area definition in figure 14D, particles are emitted by calling GENP(PL) on the basis of calculated PL values. Thus, particles are emitted by subroutine SORGEN randomly in both space and time (DELP) .

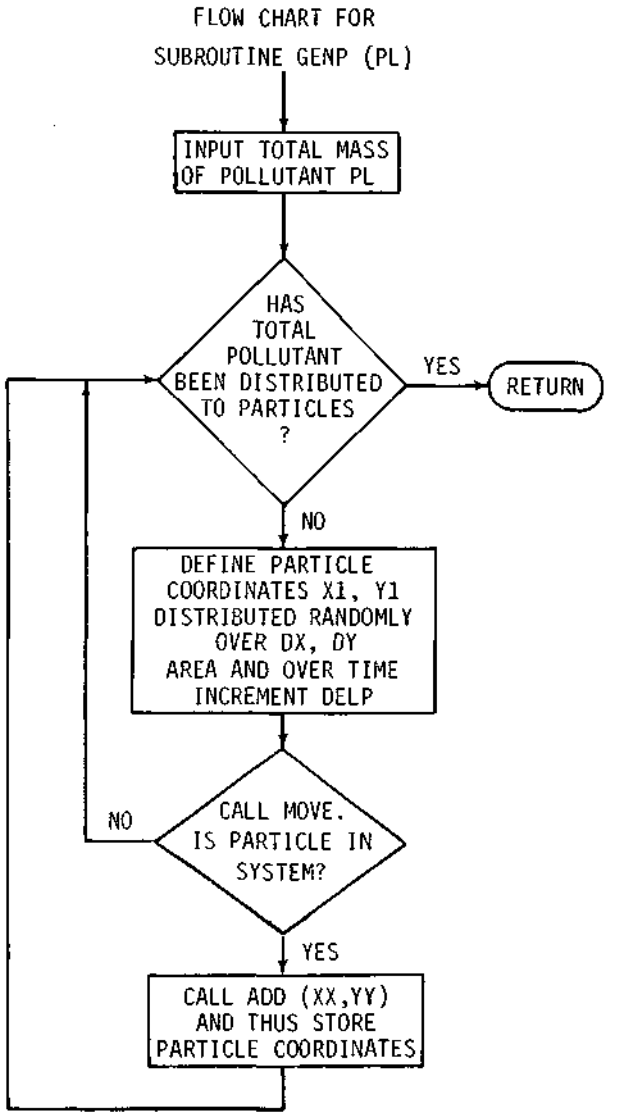


Figure 33. Flow chart for Subroutine GENP(PL)

Subroutine SOURCE

This subroutine is similar to SORGEN with the exception that particles are emitted right at the I,J coordinates of the source and by calling subroutine ADD (XX, YY) only when a sufficient quantity of water has left the source in the amount of PM. In other words, the particle is emitted at the beginning of the front and then no other one is emitted until the specified PM has flowed from the source. Thus, the particles are emitted in uniform lumps at various times during the interval DELP

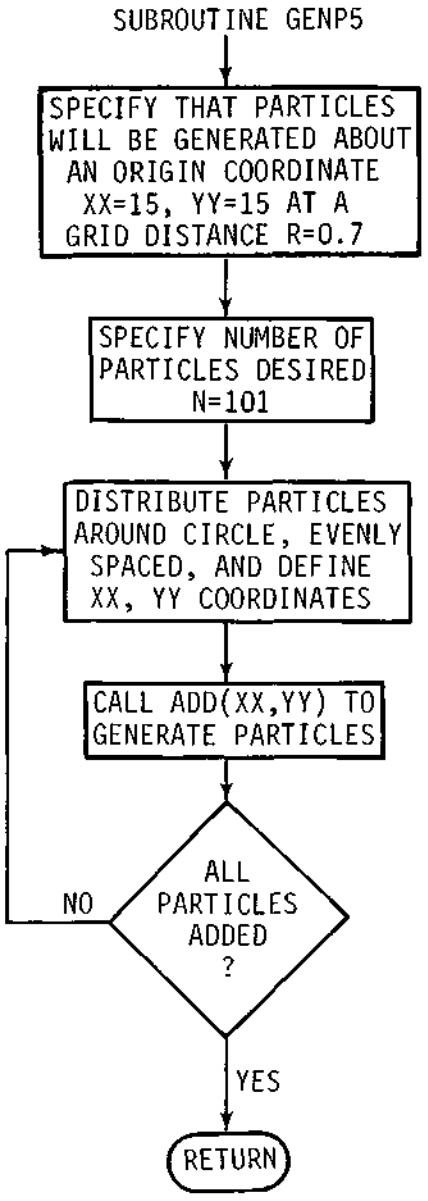


Figure 34. Flow chart for Subroutine GENP5

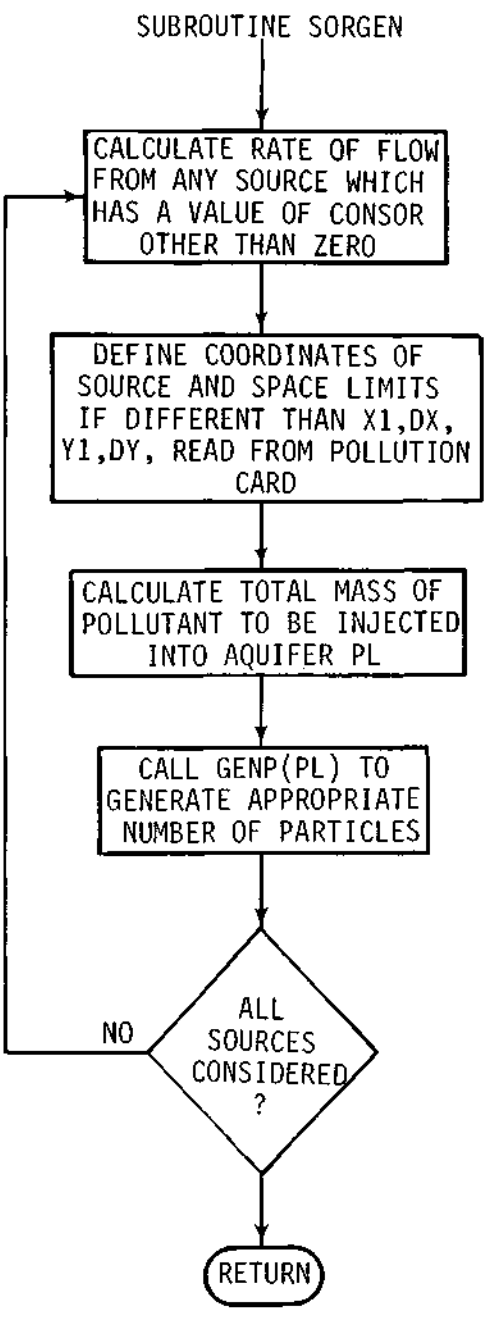


Figure 35. Flow chart for Subroutine SORGEN

when using the subroutine SOURCE. The flow chart, listing, and area configuration are shown in figures 36, 8, and 14F, respectively.

Optional Subroutine for Calculating Particle Retardation

Subroutine RDSOLV

RDSOLV is a simple subroutine which can be used to calculate a retardation factor. It solves the equation

$$RD1 = 1 + (RHO/EPOR) \times KD \qquad (18)$$

only when RD1 is set equal to zero. If RD1 is known and entered as input, the code will use that value. IF KD is known then RD1 will be calculated if 0.0 is the initial value of RD1.

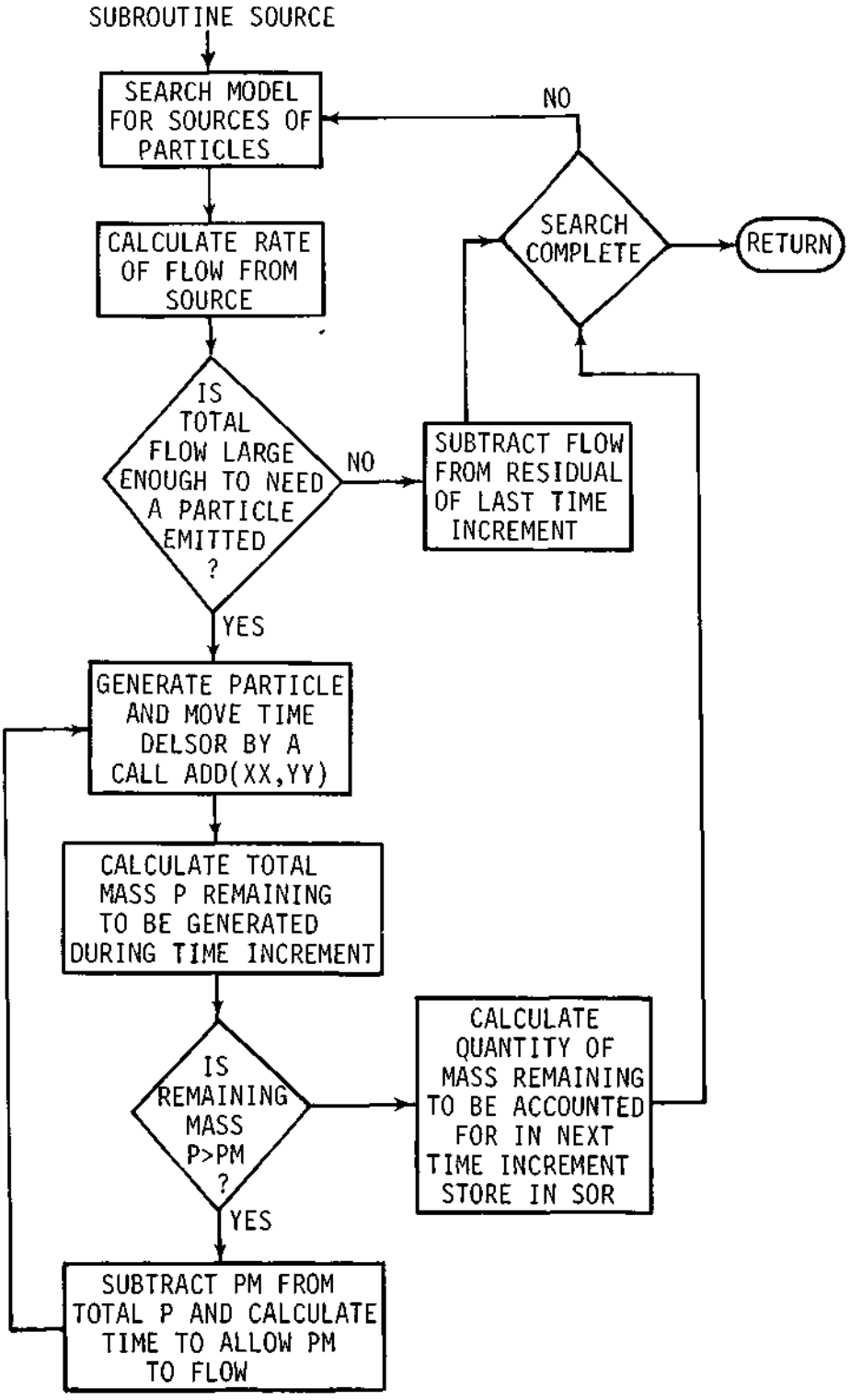


Figure 36. Flow chart for Subroutine SOURCE

Optional Subroutines That Print Results

There are four subroutines that can be called which will print summaries of concentrations, maps of particle locations, and concentrations of water flowing into sinks. Any of these can be called at any time for a report of where and what is going on.

Subroutine MAP

This subroutine prints out PM, NP, and TMAP and the NPART(I,J) array so that the location of particles in the grids can be noted. This map, however, does not report the location of particles within each grid. (It would be necessary to print out the X,Y arrays to obtain this information.) This is a convenient map to contour to get an overall picture of the situation. Figure 37 gives the flow chart for this subroutine, and figure 8 gives the listing.

Subroutine CONMAP

On the basis of equation 16, the concentration distribution is calculated and printed out in numeric table form. Figure 38 shows the flow chart and equation 16 can also be found in the listing in figure 8.

Subroutine SUMMRY

The flow chart for this subroutine is given in figure 39. Any time during the time increment DELP that a particle falls within the capture area (half of a grid either side of a specified sink), the particle is removed from the flow system and is summed into the TABLE(I) array. This subroutine prints out a summary of this action in terms of sink number and total mass deleted during any particular time increment DELP.

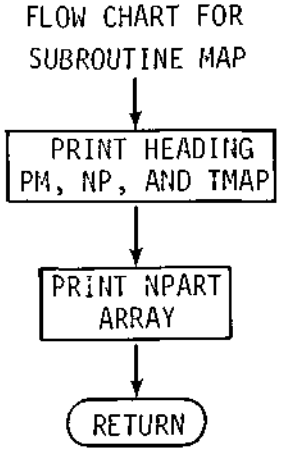


Figure 37. Flow chart for Subroutine MAP

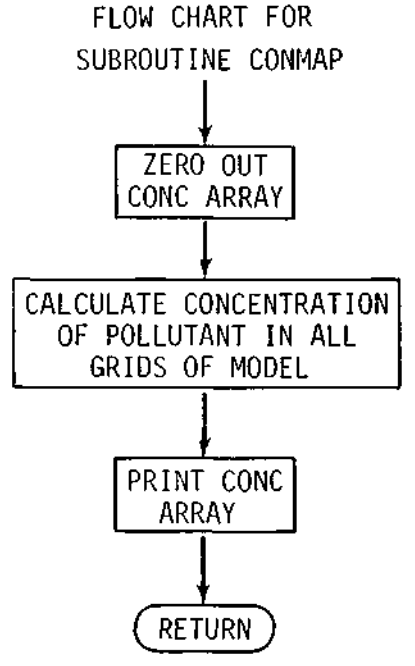


Figure 38. Flow chart for Subroutine CONMAP

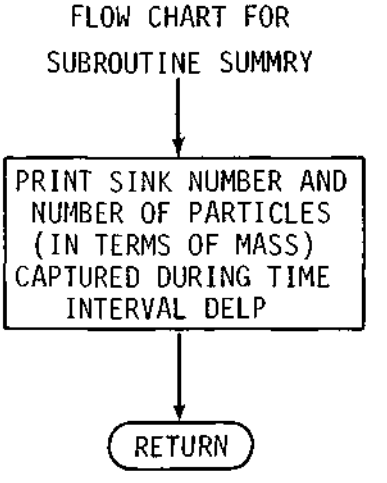


Figure 39. Flow chart for Subroutine SUMMRY

Subroutine SNKCON

This subroutine calculates the concentration of water entering sinks on the basis of whether the sink is pumping (value in the Q array) or is an area of runoff (value in R array). The appropriate formula for pumping sinks is

$$\text{Concentration} = \frac{\text{TABLE(MOVE)} \times 10^6}{\text{Q(I,J)} \times \text{DELP} \times 8.34} \qquad (19)$$

where

TABLE(MOVE) = mass of particles deleted by this sink, mg/l, during DELP

Q(I,J) = pumping rate of sink, in gpd

DELP = time increment over which particles were deleted, days

The appropriate formula for area runoff is

$$\text{Concentration} = \frac{\text{TABLE(MOVE)} \times 10^6}{8.34 \times [\text{H(I,J)} - \text{RH(I,J)}] \times \text{DELP} \times \text{R(I,J)}} \qquad (20)$$

Figure 40 illustrates the flow chart for this subroutine. The listing in figure 8 contains the code for equations 19 and 20.

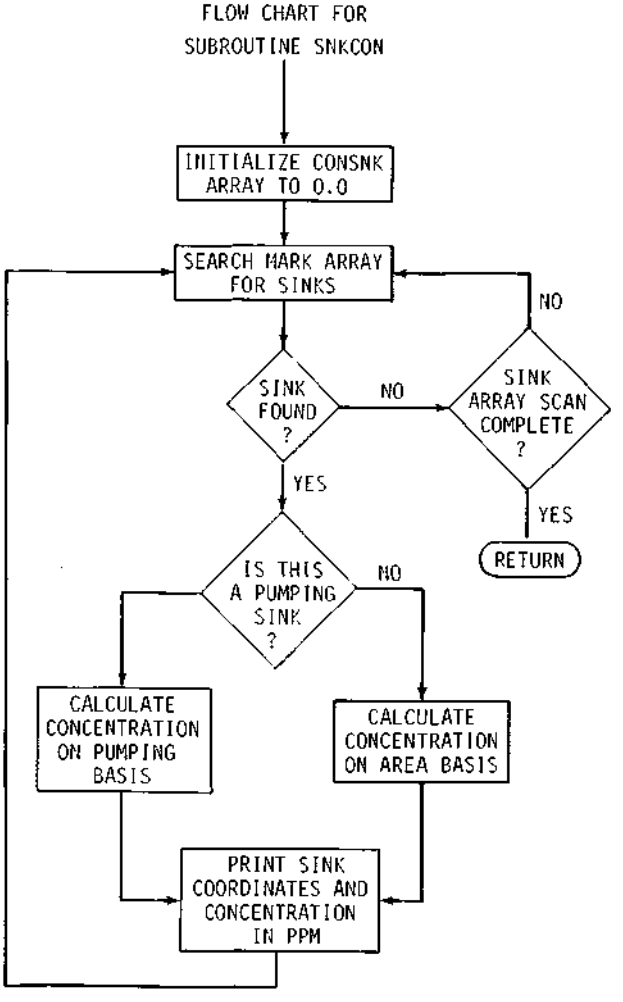


Figure 40. Flow chart for Subroutine SNKCON

PART 4. EXAMPLE PROBLEMS AND COMPARISONS WITH THEORY

The following group of example problems was chosen to illustrate how the computer code is assembled and applied. Where possible, theoretical solutions are compared with the computer output so the reader can observe the accuracy of the technique and gain confidence. Each of the examples includes a description of the problem, a list of the required data input cards, the chosen sequence of subroutine calls, and a comparison of the results with theory.

Divergent Flow from an Injection Well in an Infinite Aquifer without Dispersion or Dilution

This example problem is a convection problem chosen to demonstrate how well the velocity VELO works for a severely divergent problem. The problem is also chosen to demonstrate how a circular distribution of particles appears when reported on the basis of the square grid configuration produced by the subroutine MAP.

Figure 41 gives the input data necessary to do this problem. In actuality, the only data used in the input cards come from the parameter card (NSTEPS = 1), the default value card (NC = 30, NR = 29), the variable grid cards [DELX(I) and DELY(J) = 1000 feet], the pollution initial conditions card where DELP = 100 days, and the pollution parameter card where MAXP = 5000, DISPL = 1E-10, DISPT = 1E-10, and EPOR = 0.2. All other data are read

A

```
1  1E10   .1     30
  30    29   1E5  1E-4   0.0   0.0   0.0   0.0   -30  1E-1   -30   1E4   -40
   1     2     1
25  3 0.0
  1000    1000    1000    1000    1000    1000    1000    1000    1000    1000
  1000    1000    1000    1000    1000    1000    1000    1000    1000    1000
  1000    1000    1000    1000    1000    1000    1000    1000    1000    1000
  1000    1000    1000    1000    1000    1000    1000    1000    1000    1000
  1000    1000    1000    1000    1000    1000    1000    1000    1000    1000
  1000    1000    1000    1000    1000    1000    1000    1000    1000
     1.          0.          2.          0.         100.
 100.   5000   500.  1E-10  1E-10     .2     .2    0.0    0.0    2.0
30  2  1
BLANK LINE
BLANK LINE
```

B

```
      CALL HSOLV4

C***  CALL APPROPRIATE SUBROUTINES
C***  FOR AQUIFER SITUATION
C

      CALL GENP5

      DO 620 I=1,NPITS

      CALL SUMMRY
      CALL SNKCON
      CALL CLEAR
      CALL ADVAN(DELP)
      CALL MAP
      CALL CONMAP

 620  CONTINUE
```

Figure 41. Input data for divergent flow problem (A) and subroutine sequence (B)

and written but not used since all this information is supplied by subroutines HSOLV4 and GENP5. Refer to figures 8 and 42 for these other data. The extra data cards are included to avoid re-writing the code in MAIN .

Figure 41B gives the sequence of subroutine calls that will solve this problem. The subroutine calls in figure 15A and B are merely replaced by those in 41B. Subroutine HSOLV4 produces the steady-state head distribution, GENP5 generates 101 particles in a circle around the injection well, and the *DO* 620 loop prints out the particle distribution as the particles follow along the velocity vectors every 100 days.

Figure 42 shows some of the results of the particle positions as produced by subroutine MAP as a function of time. The theoretical solution of particle location and density is also given in figure 42. One should recall that MAP is reporting on the basis of particles found in an area 1/2 grid either side of the I,J coordinates. If the x, y coordinates of the particles themselves were printed, a smooth curve drawn through them would still be a circle. A Calcomp subroutine would be useful here, but that is left to the user to implement.

Pumping from a Well Near a Line Source of Contaminated Water, with Dilution but without Dispersion

Consider pumping 1 million gallons of water per day from a well located 5000 feet from a constant line source of contaminated water of concentration 200 mg/l greater than the resident groundwater. Assume a semi-infinite aquifer and steady-state flow. Also assume that the line source becomes polluted at time zero and that you are interested in the time-concentration curve for the pumped well. A cross-sectional view of this situation is shown in figure 43 along with other data.

Subroutine HSOLVE and the input data (figure 44) were used to solve this problem. An NC = 30 by NR = 29 variable grid model was used and a pumping well located at I = 25, J = 15 was specified. A time increment DELTA = 1E10 was used to produce the steady-state head distribution; a sink was specified at I = 25, J = 15 and MARK = 1; and source concentration data were prepared along column 30. Node lines were prepared specifying recharge factors R(I,J) of 1E10 to obtain a constant head along column 30 and a means of measuring flow from that source. The particle mass PM was calculated from equation 15 as 500 so that 50 particles found in one of the 1000' x 1000' grids would represent 200 mg/1. This was judged to be sufficient for resolution of the distribution of the concentration.

Figure 44B shows the sequence of subroutines that was chosen to solve this problem. Here again, the subroutine calls in figure 15A and B are replaced by those in figure 44B. Subroutine HSOLVE produces the steady-state head distribution, subroutine SORGEN generates the correct number of particles on a node-for-node basis according to their individual flows, and the printout subroutines give reports on concentration and particle locations as the simulation progresses DELP days at a time.

Figure 45A shows the results of the sink concentration as a function of time as compared to theory. (See function MOVE in figure 8 for the definition of D.) Note the lumpy character of the output. Engineering judgment is needed in drawing a curve through the computer data. From a simple knowledge of the aquifer conditions, it should follow that the concentration curve for this well should be a smooth monotonically increasing function of time approaching 200 mg/1. Knowing in general what the solution should look like greatly aids drawing curves through the computer data. If no idea is known of what the solution should look like, simply decrease the particle mass and time increment DELP until the computer answers achieve the desired smoothness.

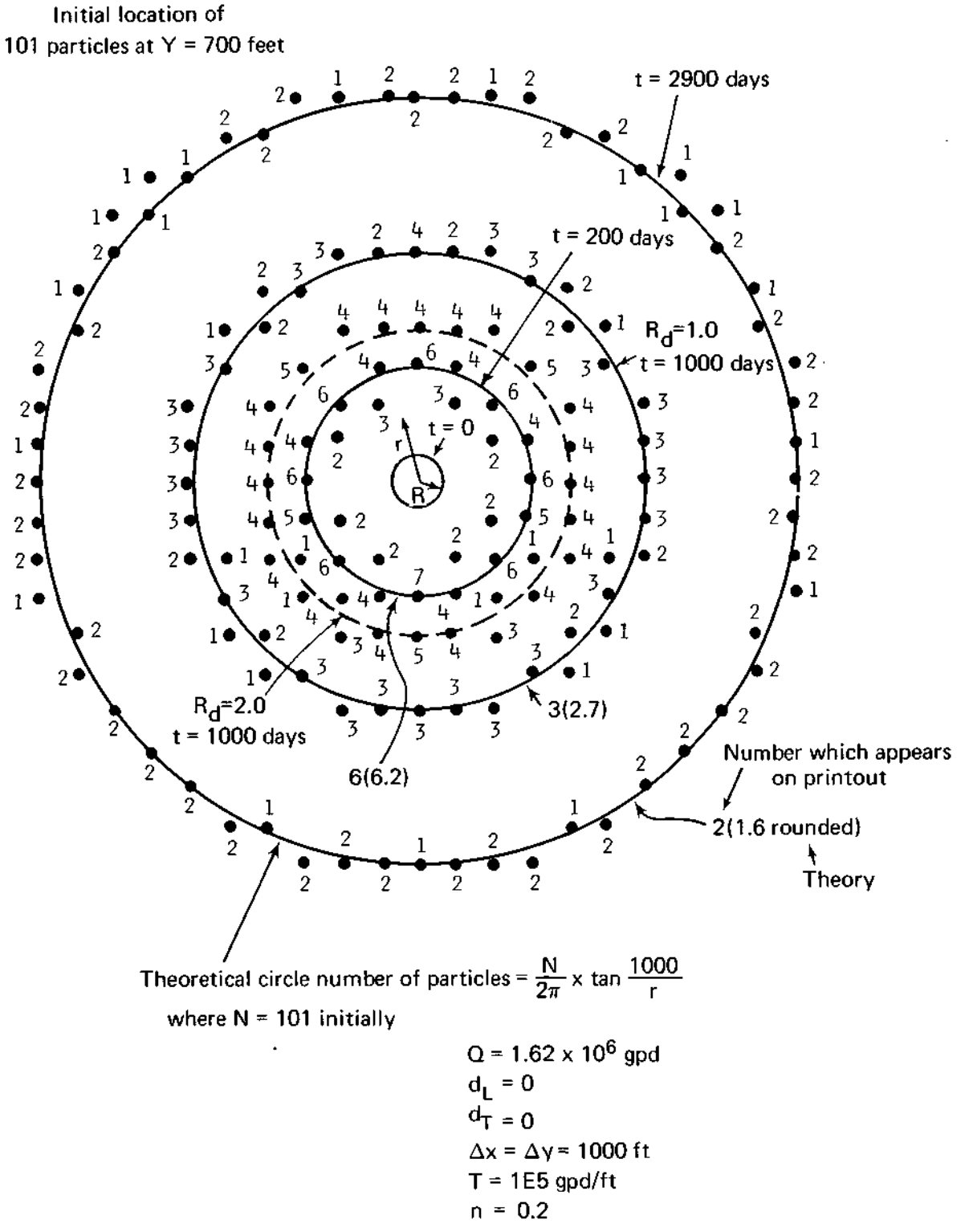


Figure 42. Outward radial flow of particles from an injection well

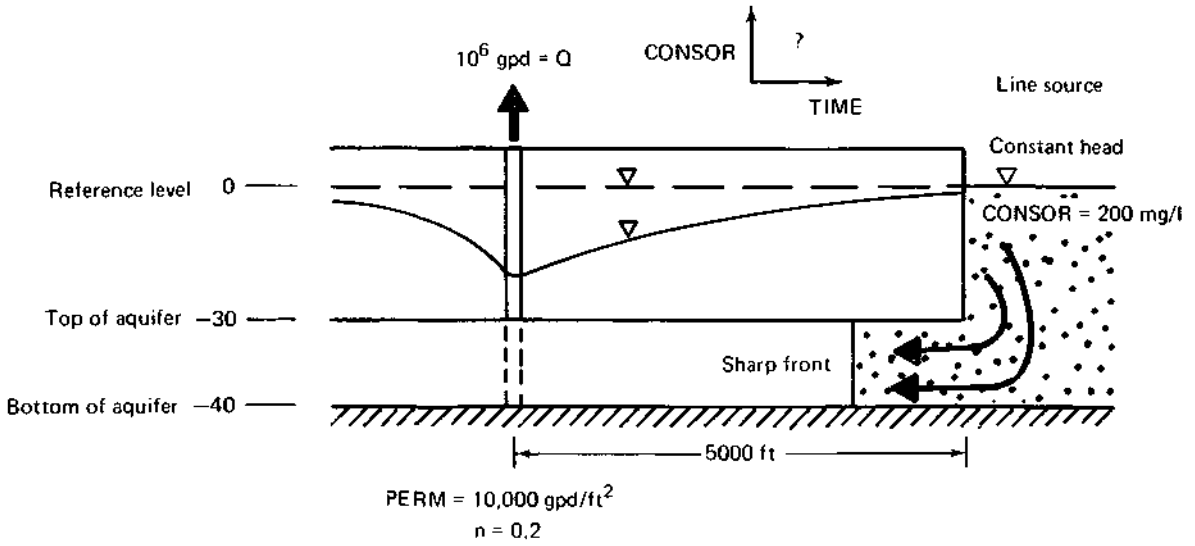


Figure 43. Cross section view of well pumping near a line source of pollution

This may be an expensive method to use, however.

With practice and some experience, it may be possible to obtain a valid solution from very few particles and rather large time increments DELP. For instance, the above problem was run with PM = 500 and DELP = 100 days, as shown in figure 45B. Another run was made with PM = 1500 and DELP = 100 days, as shown in figure 45C, and another was made with PM = 2497 and DELP = 200 days (figure 45D). An examination of the differences between the computed results of figures 45A through 45D indicates what is really needed to define the concentration curve.

Figure 45E shows that some loss of accuracy occurs when velocities of particles are computed only 2.5 times per grid movement.

Figure 46 shows a plan view of the same aquifer condition as would be depicted by the subroutine MAP just at the time of breakthrough to the well. The map was produced with the same data input as was shown in figure 44A, but with a different sequence of subroutines, as shown in figure 47.

In the plan view the difference between the actual x, y location of the particles and the method of printing out the NPART matrix on a square pattern must be realized. This explains most of the printed numbers less than 50 along the front depicted in figure 46. Here again a Calcomp routine would eliminate the need of interpretation.

A third example of the same condition is given by calling the sequence of subroutines shown in figure 48. The question of interest answered in this example is the shape of the interface as the front moves from the source to the well as a function of time. The subroutine GENP2 is called, which generates a single particle per node along the line source. Then the sequence of calling subroutines MAP, CLEAR, ADVAN (DELP) produces a series of maps as shown in figure 49. The front is easily described in this fashion with only 29 particles. An excellent definition of the front can be obtained by adding more particles through calls to GENP (PL) with Xl = 30, DX = 0, Yl = 1, DY = 29, PL = 5000, and PM = 1 and printing out the X,Y coordinate history of the particles.

A

```
 1  1E10   .1     56
 30    29   1E5  1E-4   0.0   0.0   0.0   0.0   -30  1E-1   -30   1E4   -40
  1     2     1
25 15   1E6
 64000   32000   16000    8000    4000    2000    1000    1000    1000    1000
  1000    1000    1000    1000    1000    1000    1000    1000    1000    1000
  1000    1000    1000    1000    1000    1000    1000    1000    1000    1000
 64000   32000   16000    8000    4000    2000    1000    1000    1000    1000
  1000    1000    1000    1000    1000    1000    1000    1000    1000    1000
  1000    1000    1000    2000    4000    8000   16000   32000   64000
     30.      0.       1.      28.      50.
 100.  5000   500.  1E-10  1E-10    .2      .2    0.0     0.0     2.0
25 15  1
BLANK LINE
30  1          200.
30  2          200.
30  3          200.
30  4          200.
30  5          200.
30  6          200.
30  7          200.
30  8          200.
30  9          200.
30 10          200.
30 11          200.
30 12          200.
30 13          200.
30 14          200.
30 15          200.
30 16          200.
30 17          200.
30 18          200.
30 19          200.
30 20          200.
30 21          200.
30 22          200.
30 23          200.
30 24          200.
30 25          200.
30 26          200.
30 27          200.
30 28          200.
30 29          200.
BLANK LINE
30  1   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30  2   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30  3   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30  4   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30  5   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30  6   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30  7   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30  8   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30  9   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 10   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 11   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 12   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 13   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 14   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 15   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 16   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 17   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 18   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 19   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 20   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 21   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 22   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 23   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 24   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 25   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 26   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 27   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 28   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
30 29   1E5  1E51E-4 0.0   0.0  1E10   0.0   -30  1E30   -30   1E4   1E4   -40
```

B

```
      CALL HSOLVE

C***  CALL APPROPRIATE SUBROUTINES
C***  FOR AQUIFER SITUATION
C

      DO 620 I=1,NPITS

      CALL SUMMRY
      CALL SNKCON
      CALL CLEAR
      CALL ADVAN(DELP)
      CALL SORGEN
      CALL MAP
      CALL CONMAP

 620  CONTINUE
```

Figure 44. Input data for pumping in the vicinity of a polluted line source (A) and subroutine sequence (B)

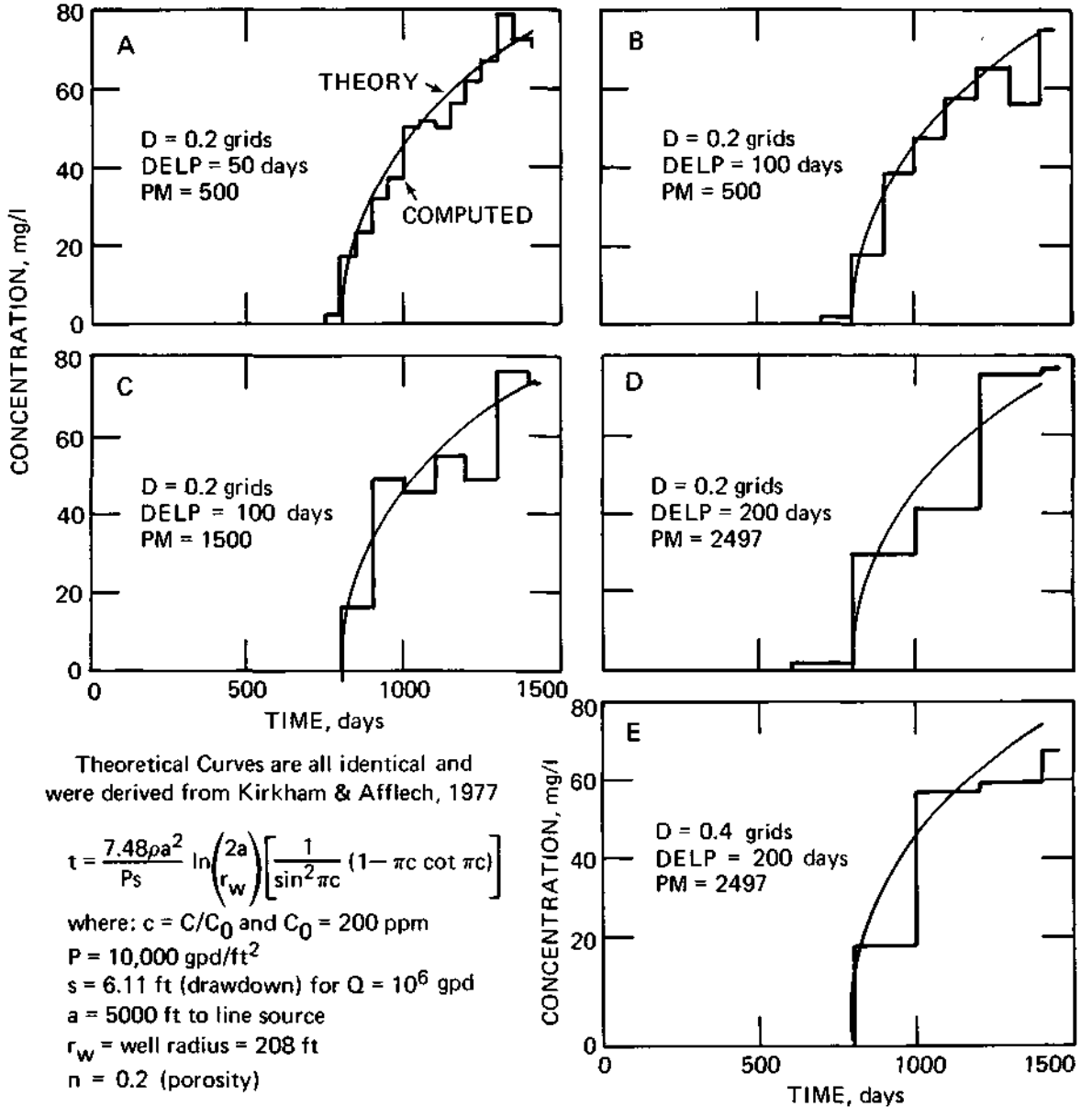
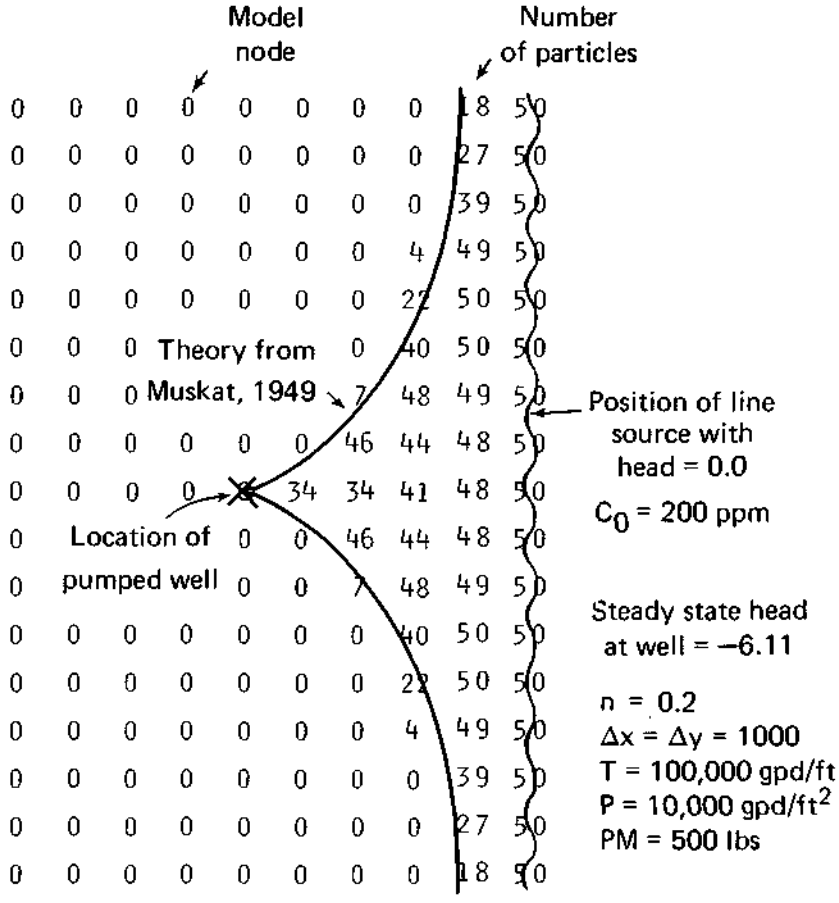


Figure 45. Concentration of water derived from a well pumping near a line source of pollution



200 ppm = 50 particles in one grid

$$DELX * DELY * (CH - BOT) * n * \frac{gal}{ft^3} * \frac{lbs}{gal}$$

$$\frac{MASS}{grid} = 1000 \times 1000 \times 10 \times 0.2 \times 7.48 \times 8.3453 = \frac{lbs}{grid}$$

$$\frac{50 \times 500}{MASS/grid} = 200 \text{ ppm}$$

Figure 46. Distribution of particles in the vicinity of a well pumping near a line source of pollution

```
      CALL HSOLVE

C***  CALL APPROPRIATE SUBROUTINES
C***  FOR AQUIFER SITUATION
C

      DO 620 I=1,NPITS

      CALL SUMMRY
      CALL SNKCON
      CALL CLEAR
      CALL ADVAN(DELP)
      CALL SOURCE
      CALL MAP
      CALL CONMAP

 620  CONTINUE
```

Figure 47. Subroutine sequence to produce figure 46

```
      CALL HSOLVE

C***  CALL APPROPRIATE SUBROUTINES
C***  FOR AQUIFER SITUATION
C

      CALL GENP2
      DO 620 I=1,NPITS

      CALL SUMMRY
      CALL SNKCON
      CALL CLEAR
      CALL ADVAN(DELP)
      CALL MAP
      CALL CONMAP

 620  CONTINUE
```

Figure 48. Subroutine sequence to produce figure 49

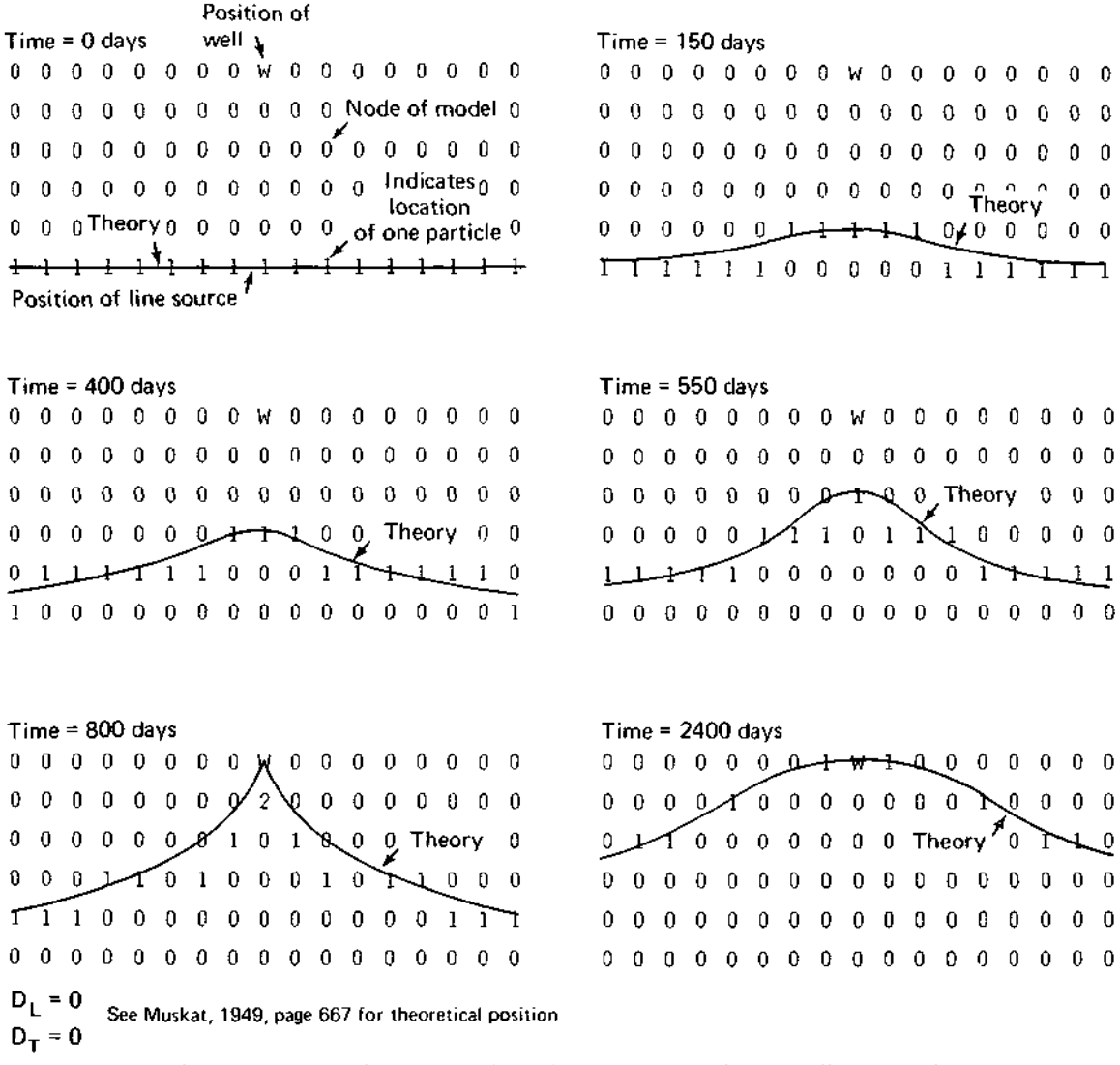


Figure 49. Advance of a front toward a well pumping near a line source

Longitudinal Dispersion in Uniform One-Dimensional Flow with Continuous Injection at X = 0.0

The data deck for this situation is shown in figure 50A. An NC = 30 and NR = 3 flow model is used for this problem. Longitudinal dispersivity is set at 4.5 feet and the transverse dispersivity at zero. The total pollutant PL is set at 100. The sequence of subroutine calls for this problem is shown in figure 50B. The subroutine HSOLV2 is used to provide the head distribution such that the x-direction velocity of flow is 1 foot/day. A grid interval of 10 feet is used to map the distribution of particles as they are generated at x = 1,y = 2 randomly over the time increment DELP. This approximates the continuous injection scheme desired and fully randomizes the flow.

The resulting set of maps provided the necessary data to plot the graphs of figure 51. The theoretical curves were derived from equations given by Fried (1975). Again, note the lumpiness of the results. Remedies could include using more particles than 100 per time increment, and smaller grid sizes. Likewise, application of engineering judgment would allow fewer particles. Notice in figure 51 that there is no "overshoot" in the traditional

A

```
 1   1E10    .1      20
 30      3    1E5   1E-4    0.0    0.0    0.0    0.0   -30   1E-1   -30    1E4    -40
  1      2      1
25   3 0.0
      10       10       10       10       10       10       10       10       10       10
      10       10       10       10       10       10       10       10       10       10
      10       10       10       10       10       10       10       10       10       10
      10       10       10
        1.          0.          2.          0.          10.
  100.    5000       1.     4.5   1E-10      .2       .2     0.0     0.0     2.0
30   2   1
BLANK LINE
BLANK LINE
```

B

```
      CALL HSOLV2

C***  CALL APPROPRIATE SUBROUTINES
C***  FOR AQUIFER SITUATION
C

      DO 620 I=1,NPITS

      CALL SUMMRY
      CALL SNKCON
      CALL CLEAR
      CALL ADVAN(DELP)
      CALL GENP(PL)
      CALL MAP
      CALL CONMAP

 620  CONTINUE
```

Figure 50. Input data for dispersion in a uniform one-dimensional flow with continuous injection at x=0 (A) and subroutine sequence (B)

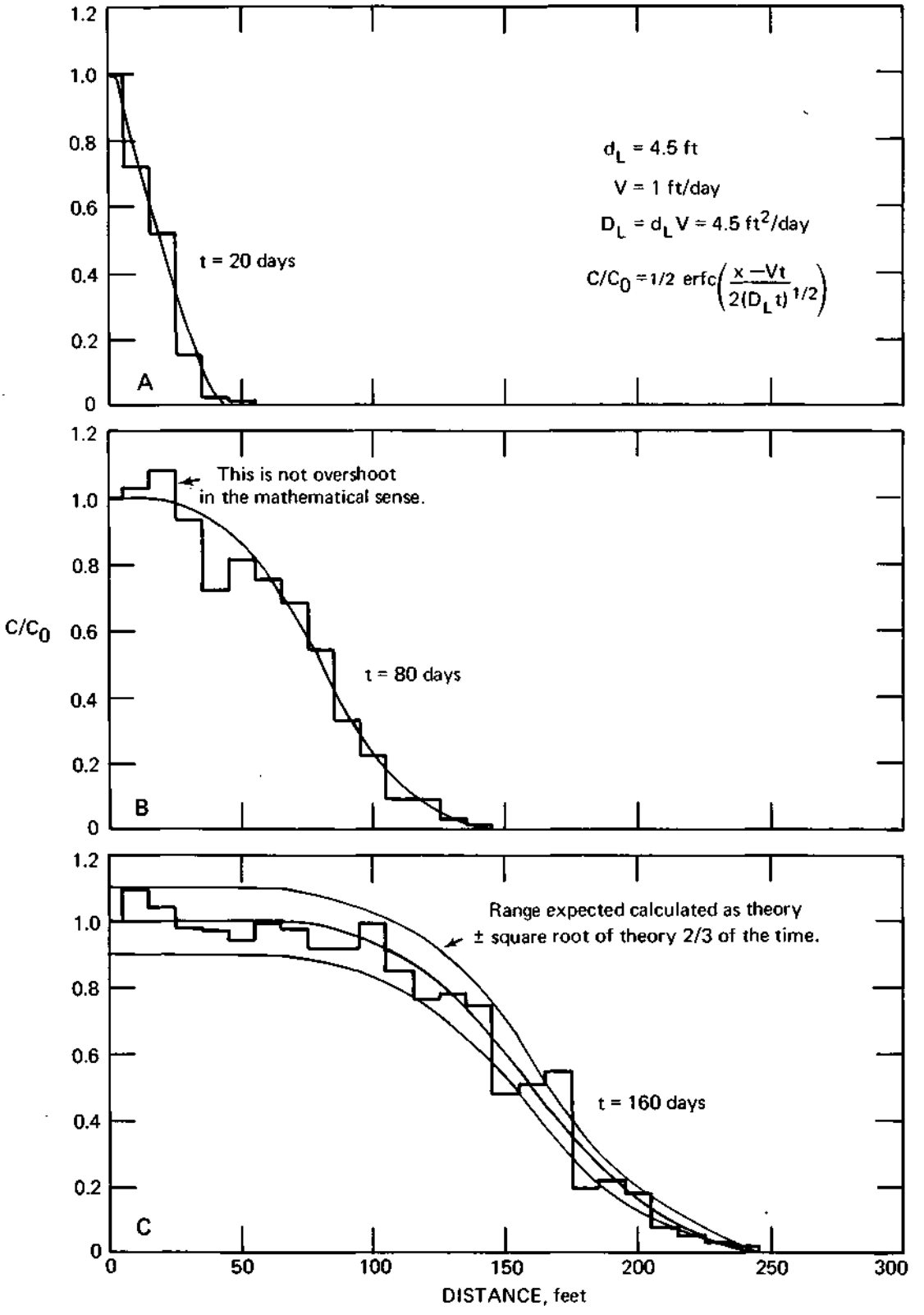


Figure 51. Longitudinal dispersion in uniform one-dimensional flow with continuous injection at x=0

mathematical sense of the word. An increase in the number of particles of the simulation will reduce the magnitude of the $C/C_o > 1$ phenomenon shown in figures 51B and 51C. No number of additional particles would cure such a phenomenon in the MOC.

Statisticians note that it is possible to express the probability of the variation in magnitude of the number of particles either side of the number actually printed. With this type of simulation, the possible range in the number of particles printed varies according to a Poissan distribution, and the magnitude of that possible variation would vary according to the square root of the number printed. Thus, if the number 10 shows up, its actual value could range from $(10\text{-}10^{½})$ to $(10+10^{½})$, or 6.8 to 13.2. This fact helps in deciding what may be significant when interpreting printout in this type of simulation. Use of this concept, applied to the theoretical curve, shows that nearly all simulated values of figure 51C fall within the expected range.

Longitudinal Dispersion in Uniform One-Dimensional Flow with a Slug of Tracer Injected at X = 0.0

The data for this problem are the same as those used in figure 50. The only difference between the previous continuous injection scheme and this problem of an injected slug is in the sequence of subroutines called. Figure 52 shows that GENP3 is called once to get the slug injected and the remainder of the subroutine calls advance the particles and print out the results as the slug moves downstream.

Figure 53 gives the results of this simulation compared with the theoretical results derived by an equation given by Bear (1972). The effects of injecting slugs of increasing numbers of particles are illustrated by comparing figures 53A, 53B, and 53C. Figures 53A through 53C can actually be produced with a single simulation by calling GENP3 applicable to three separate rows of the flow model. Then, the concentration curve of figure 53A would be plotted on the basis of the first row data. Figure 53B would be plotted by superposing data from rows 1 and 2. And finally, figure 53C would be plotted on the basis of the sum of all three rows.

Longitudinal Dispersion in a Radial Flow System Produced by an Injection Well

Figure 54A gives the input data for an NC = 30 by NR = 29 model with longitudinal dispersivity set to 450 feet (a high dispersivity like this is unheard of in the traditional sense of dispersion theory; however, it could be attributed to the effects of aquifer stratification). A grid interval of 1000 feet is used to give a severe test to the discrete particle technique.

Figure 54B illustrates the choice of subroutine calls to solve this problem. The subroutine HSOLV4 defines the head distribution by the Theim formula, and GENP4 generates the particles in the vicinity of the injection well.

Figure 55 gives the results of this problem compared with theory derived from an equation given by Bear (1972). A plan view map of the concentration distribution at time = 350 days is given in figure 56 along with the theory for comparison. If more infor-

```
      CALL HSOLV2

C***  CALL APPROPRIATE SUBROUTINES
C***  FOR AQUIFER SITUATION
C

      CALL GENP3
      DO 620 I=1,NPITS

      CALL SUMMRY
      CALL SNKCON
      CALL CLEAR
      CALL ADVAN(DELP)
      CALL MAP
      CALL CONMAP

  620 CONTINUE
```

Figure 52. Subroutine sequence for longitudinal dispersion in a uniform one-dimensional flow field with a slug of tracer injected at x=0

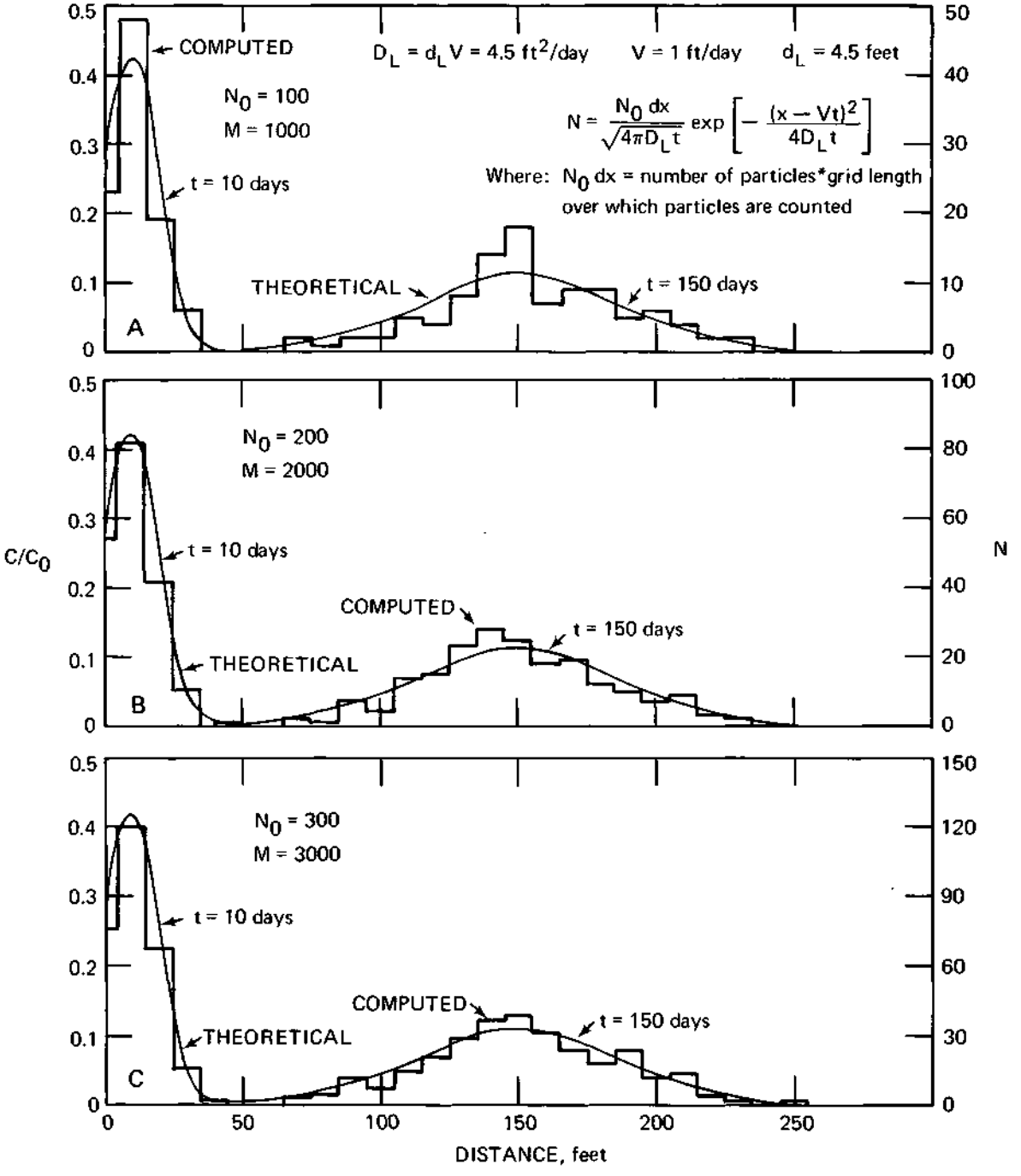


Figure 53. Longitudinal dispersion in a uniform one-dimensional flow with a slug of tracer injected at x=0

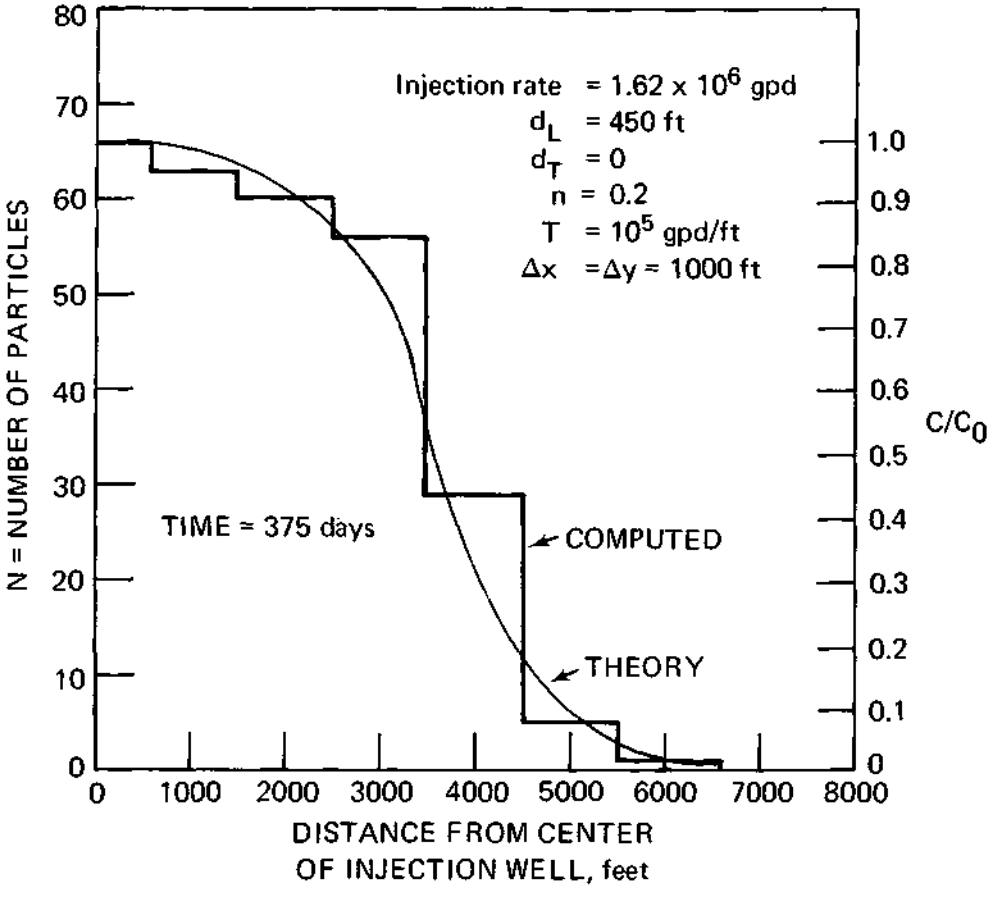
A

```
 1  1E10   .1     10
 30    29   1E5  1E-4   0.0   0.0   0.0   0.0   -30  1E-1   -30   1E4   -40
  1     2     1
25  3 0.0
   1000    1000    1000    1000    1000    1000    1000    1000    1000    1000
   1000    1000    1000    1000    1000    1000    1000    1000    1000    1000
   1000    1000    1000    1000    1000    1000    1000    1000    1000    1000
   1000    1000    1000    1000    1000    1000    1000    1000    1000    1000
   1000    1000    1000    1000    1000    1000    1000    1000    1000    1000
   1000    1000    1000    1000    1000    1000    1000    1000    1000
      1.        0.        2.        0.       50.
  100.   5000   500.   450.  1E-10     .2     .2    0.0    0.0    2.0
30  2  1
BLANK LINE
BLANK LINE
```

B

```
      CALL HSOLV4

C***  CALL APPROPRIATE SUBROUTINES
C***  FOR AQUIFER SITUATION
C

      DO 620 I=1,NPITS

      CALL SUMMRY
      CALL SNKCON
      CALL CLEAR
      CALL ADVAN(DELP)
      CALL GENP4
      CALL MAP
      CALL CONMAP

 620  CONTINUE
```

Figure 54. Input data for longitudinal dispersion in a radial flow pattern (A) and subroutine sequence (B)



$$C/C_0 = 1/2 \ \text{erfc}\left[\frac{r - \bar{r}}{\sqrt{4/3\, d_L \bar{r}}}\right]$$

Where:

r = average radius of injected water

Relation of C/C_0 to number of particles

$$C_0 = \frac{360 \text{ particles}}{1.62 \times 10^6 \text{ gal/day} \times 50 \text{ days}} \times \frac{1000 \times 1000 \times 7.48 \times 0.2 \text{ gal}}{\text{grid}} = \frac{66 \text{ particles}}{\text{grid}}$$

Figure 55. Radial dispersion from an injection well

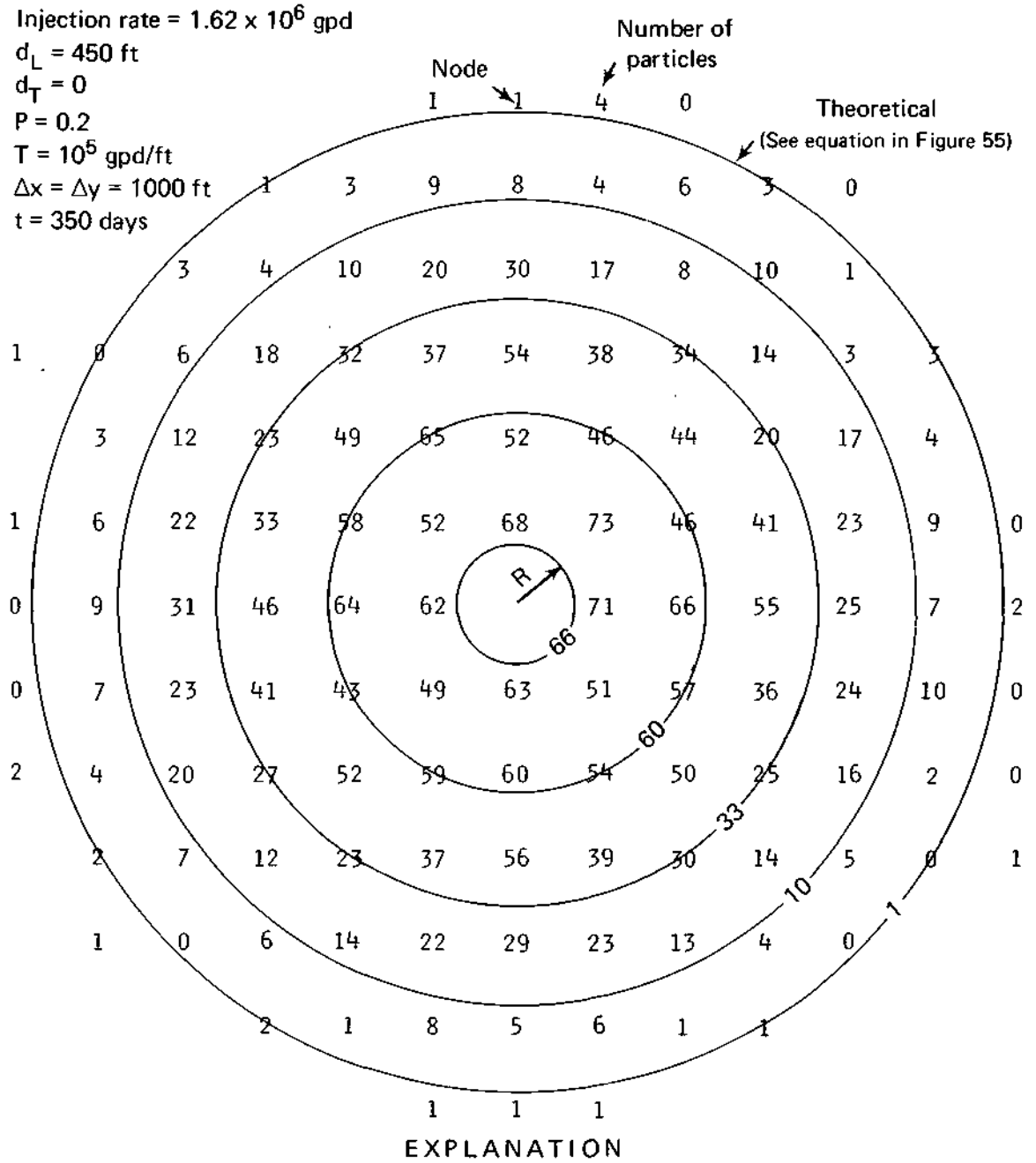


Figure 56. Map of theoretical versus computed radial dispersion from an injection well

mation is desired to better define the concentration distribution in figure 56, the simulation should be rerun with a finer grid interval and more particles .

Longitudinal and Transverse Dispersion in Uniform One-Dimensional Flow with a Slug of Tracer Injected at X = 0.0

Figure 57A gives the input data for this problem with a longitudinal dispersivity of 4.5 ft and a 1.125-ft transverse dispersivity. The subroutine calls are given in figure 57B, showing that HSOLV2 is used to produce the head distribution and GENP3 is called once to generate the slug. Before this problem is run, GENP3 has to be modified to the following: 1) DO 10 I = 1,200; 2) YY = 15; and 3) XX = 5. Figure 58 gives the computer results compared with a theoretical solution derived from an equation given by Fried (1975). In this example, concentration is replaced simply by the distribution of particles found in the model. The agreement between computer results and theory is excellent.

Figures 59 and 60 illustrate in plan view the output for this problem as produced by subroutine MAP when the number of particles is varied between 500 and 2000.

A

```
 1  1E10   .1     20
30    29   1E5  1E-4   0.0   0.0   0.0   0.0   -30  1E-1   -30   1E4   -40
 1     2     1
25  3 0.0
     10      10      10      10      10      10      10      10      10      10
     10      10      10      10      10      10      10      10      10      10
     10      10      10      10      10      10      10      10      10      10
     10      10      10      10      10      10      10      10      10      10
     10      10      10      10      10      10      10      10      10      10
     10      10      10      10      10      10      10      10      10
       1.          0.          2.          0.          10.
  100.   5000     1.    4.5  1.125     .2     .2    0.0    0.0    2.0
30  2  1
BLANK LINE
BLANK LINE
```

B

```
      CALL HSOLV2

C***  CALL APPROPRIATE SUBROUTINES
C***  FOR AQUIFER SITUATION
C

      CALL GENP3

      DO 620 I=1,NPITS

      CALL SUMMRY
      CALL SNKCON
      CALL CLEAR
      CALL ADVAN(DELP)
      CALL MAP
      CALL CONMAP

 620  CONTINUE
```

Figure 57. Input data for longitudinal and transverse dispersion (A) and subroutine sequence (B)

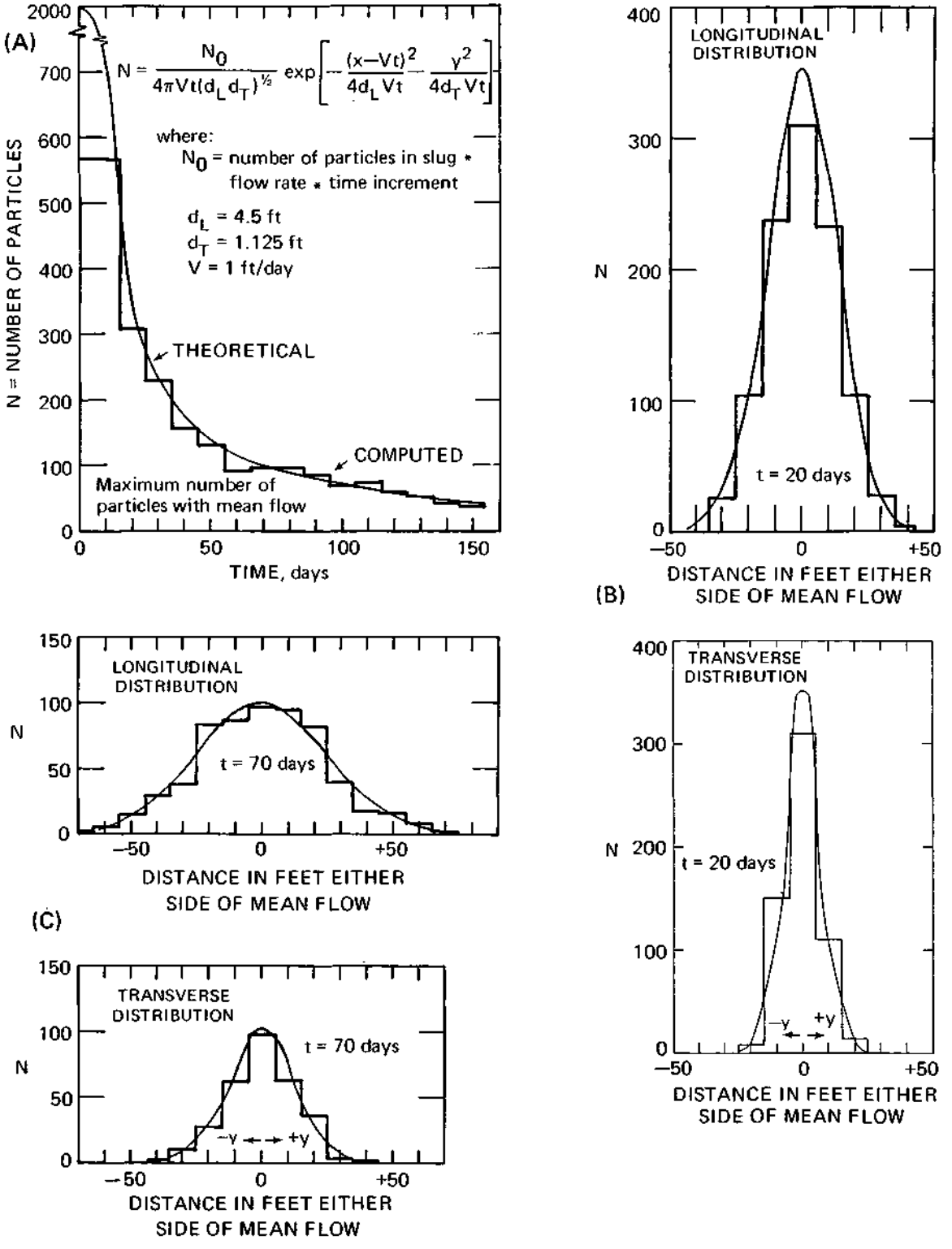


Figure 58. Dispersion of a slug injected in a uniform one-dimensional flow in the x direction

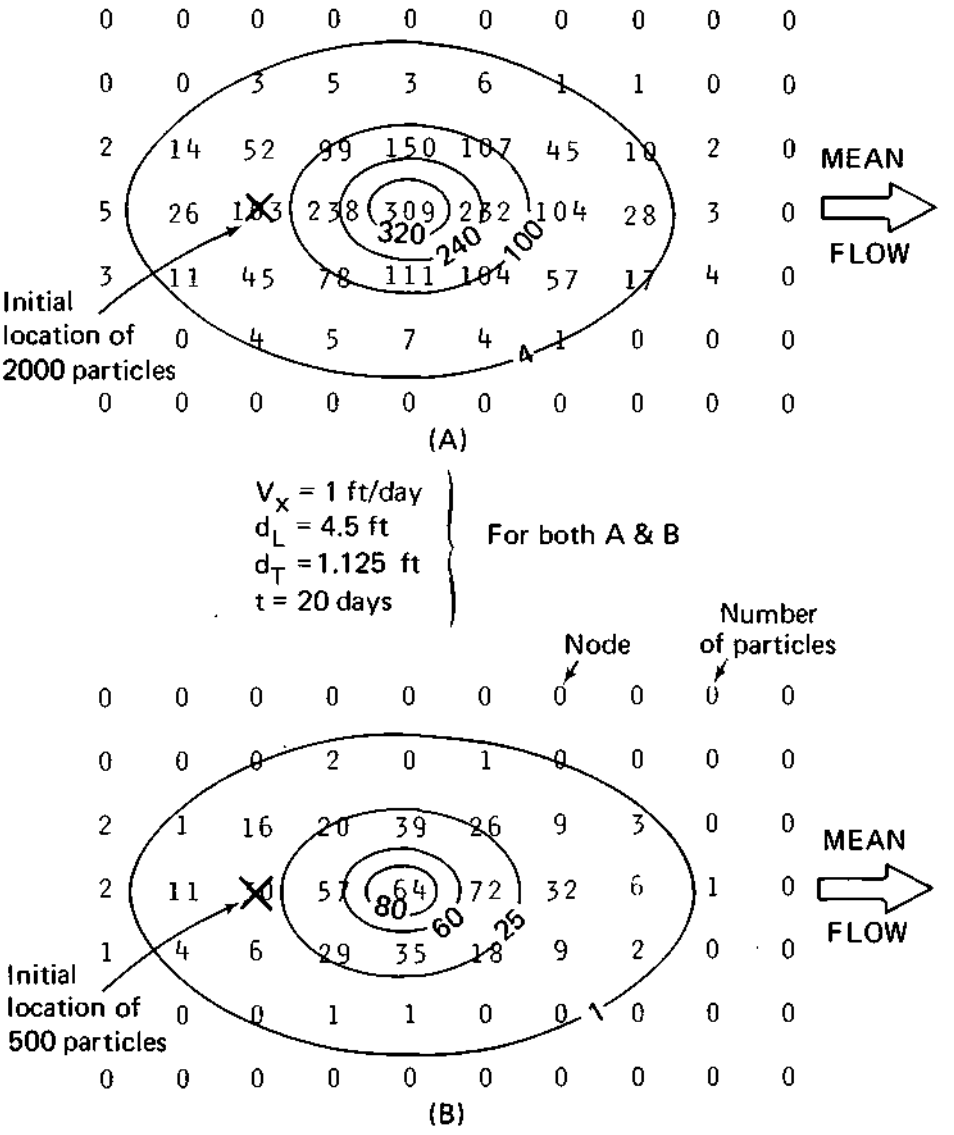


Figure 59. Map of the number of particles of the model grid 20 days after injection of a slug into a uniform flow compared to theory

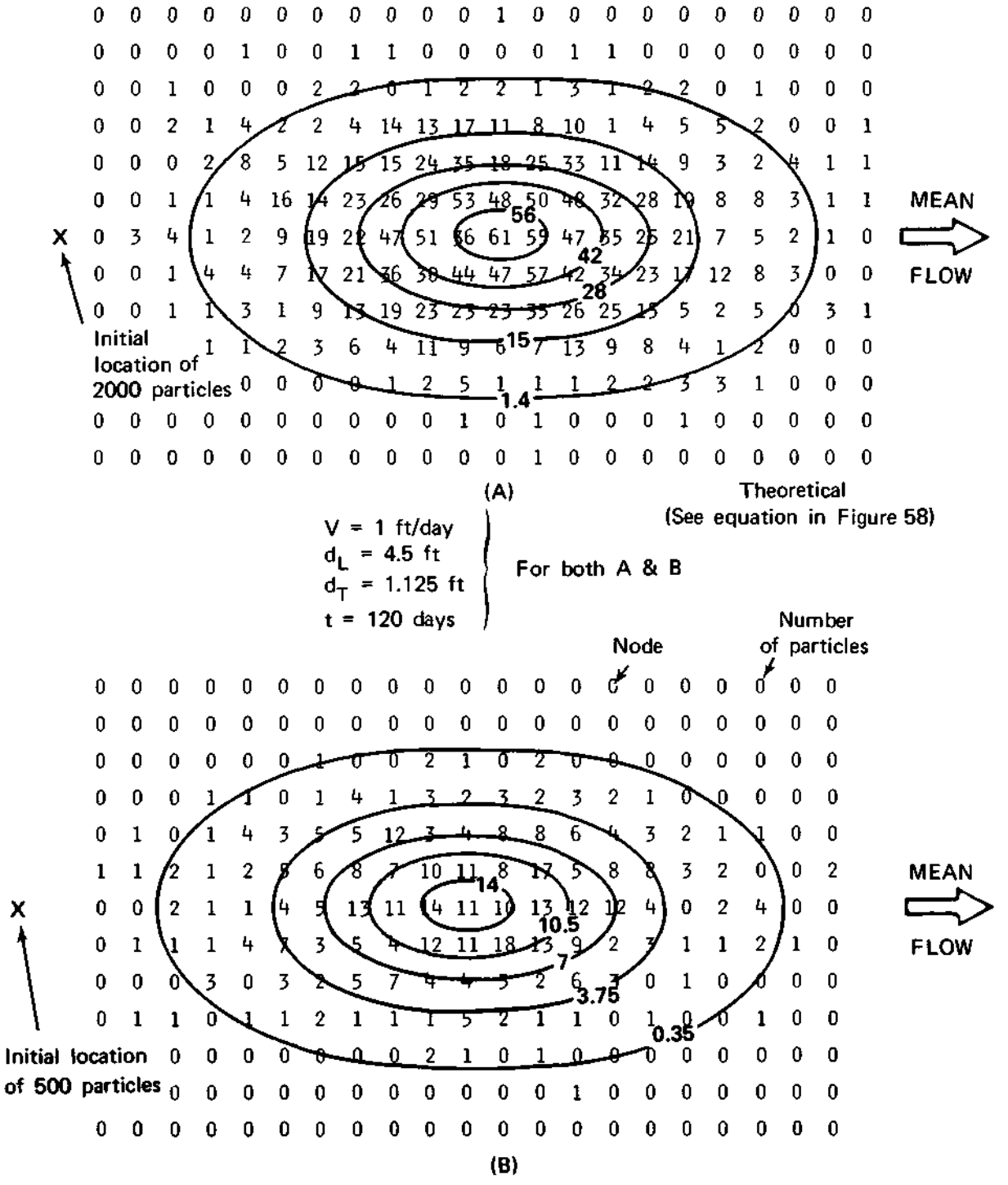


Figure 60. Map of the number of particles of the model grid 70 days after injection of a slug into a uniform flow compared with theory

PART 5. FIELD APPLICATION; MEREDOSIA, ILLINOIS

The field problem is located in Morgan County, Illinois, south of Meredosia, near the Illinois River (figure 61). The area is industrialized, with petroleum and agricultural chemical companies occupying most of the property. The contamination problem was first detected in 1978 at a distribution terminal for liquid ammonia fertilizer. This facility requires 1000 gpm (5420 m^3/day) groundwater supply to operate condensers in the refrigeration system. The water is fed directly into the system from high capacity production wells. During 1978 the pump bowls of the production wells started binding during operation because of calcium carbonate precipitation. The purpose of this study was to identify and characterize the newly developed groundwater condition in order to recommend corrective measures.

Groundwater Flow System

The site outlined in figure 62 is about 850 ft (257.4 m) by 625 ft (189.2 m). The major structures are the terminal, the office building, and two large storage tanks. Underlying the site is 90 ft (27.2 m) of unconsolidated sand and gravel on top of Pennsylvanian age bedrock (Heigold and Ringler, 1979). There is a gentle slope (.006 ft/ft) in land surface from east to west across the site. The water table, contoured from data for 12 wells (figure 62), reflects a gentle,

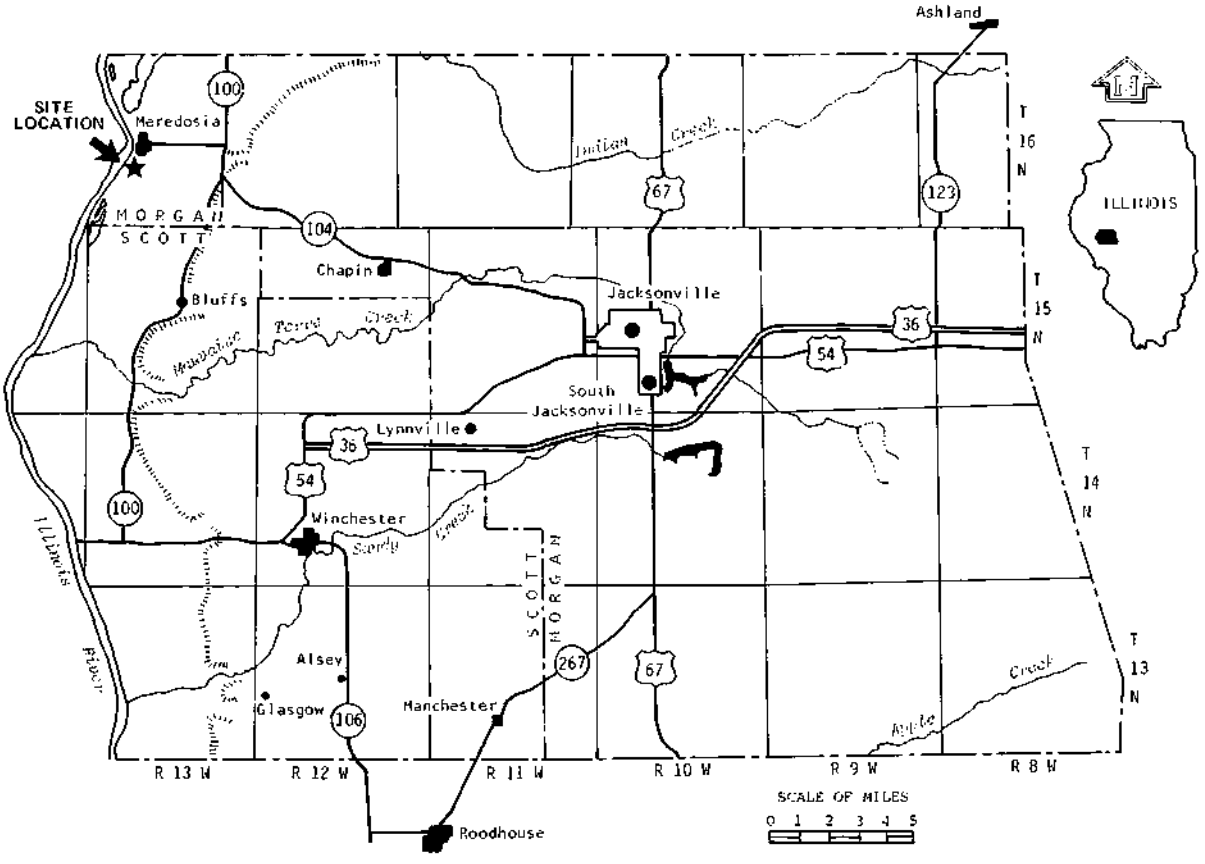


Figure 61. Location of field site

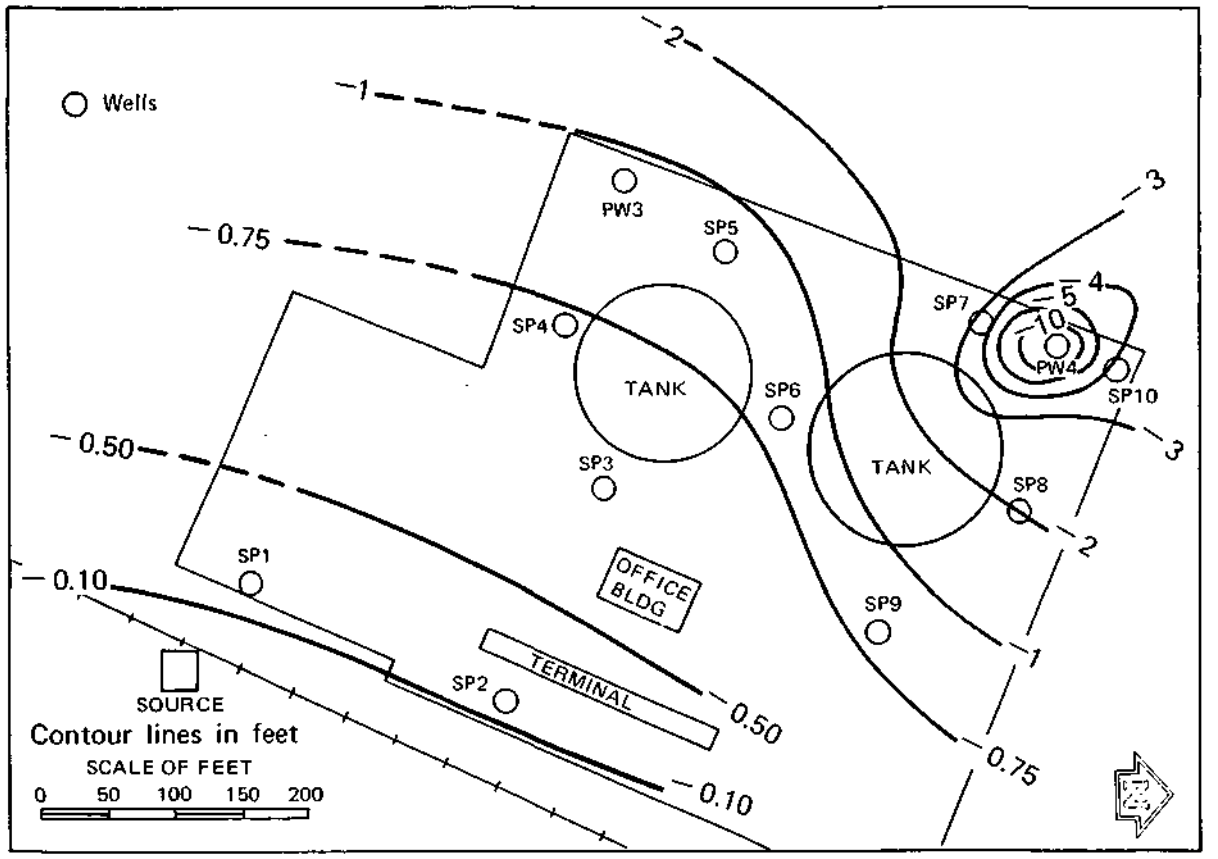


Figure 62. Water table contours and control wells at the field site (The zero datum for the contours is the water table level inferred at the source)

natural gradient toward the Illinois River (west) and illustrates the cone of depression from the production well, PW4. Aquifer transmissivities range from 150,000 gpd/ft (1863 m^2/day) to 300,000 gpd/ft (3727 m^2/day) in this part of the Illinois River bottomlands .

The pump binding problem first developed in two wells housed in the west end of the office building (figure 62). Leached chemical constituents from an uncovered chemical fertilizer bin (SOURCE) migrated down-gradient, tothe west-northwest, and were drawn into the cone of depression of the production wells in the office building, which are now abandoned. The source consisted of ammonium sulfate, nitrates, and traces of inorganic phosphates salvaged from fire-damaged fertilizer processing plants. It had been in place for at least a year prior to the development of pump-binding problems and nearly three years before this study was initiated .

Quality of Native and Contaminated Groundwater

The native groundwater at the site is of good chemical quality. The partial chemical analysis in table 2 shows that the water is only moderately mineralized although levels of nitrate are somewhat higher than the USEPA primary standard of 10 mg/1.

On the site, ten sand-point wells--SP1 through SP10 (figure 62)--were driven to check groundwater quality and levels. Wells SP1, SP4, and SP10 were

Table 2. Chemical Composition of Uncontaminated Groundwater

	(mg/l)
Calcium	68.0
Magnesium	24.0
Sodium	7.5
Potassium	2.0
Iron	0.4
Manganese	0.14
Ammonia	0.04
Bicarbonate	278.0
Sulfate	33.0
Chloride	11.0
Nitrate	13.4
Fluoride	0.10

permanently installed for the purposes of tracer experiments and monitoring. Two production wells, PW3 and PW4, were also used as monitoring wells. PW4 was producing 1000 gpm (5420 m^3/day) during the study, while PW3 was pumped only for sampling and a groundwater tracer experiment. On the basis of the chemical analysis of groundwater taken from 11 wells (table 3), the groundwater quality was classified into three groups: 1) interior plume, 2) marginal plume, and 3) distal or native groundwater.

The groundwater in the plume interior had ammonia concentrations ranging from 285 mg/1 to 2100 mg/1 (figure 63). Nitrate ranged from 570 mg/1 to 1885 mg/1 in the interior, while calcium and magnesium concentrations were well below concentrations of native groundwater. Sulfate, potassium, chloride, phosphate, and iron were found in concentrations above background (figure 64). The pH of the interior plume water ranged from 8.6 to 8.9. The pH of the native groundwater is 8.0 or slightly less.

The groundwater on the margins of the plume was identified as having concentrations noticeably above background, but considerably lower than interior concentrations (figures 63 and 64). Observation wells to the north, farthest from the source, had the lowest concentrations and were termed distal.

A. chemical analysis was made on the scale deposit which bound the pump. Calcium carbonate was the major constituent (95 percent), and magnesium carbonate comprised 0.5 to 5 percent of the scale sample by weight. The rapid precipitation of $CaCO_3$ on the pump bowls was probably caused by the high pH and by degassing due to the agitation of the groundwater in the pump bowls.

Table 3. Chemical Composition of Groundwater Samples from Study Site (mg/1)

		SP1	SP2	SP3	SP4	SP5	SP6	SP7	SP8	SP9	SP10	PW3
Potassium	K	60.0	34.0	42.0	62.0	22.0	65.0	31.0	6.00	2.00	8.00	25.0
Calcium	Ca	2.80	6.40	4.40	68.0	104.0	10.8	84.0	64.0	68.0	104.0	108.0
Magnesium	Mg	3.90	5.10	11.0	34.0	39.0	8.20	27.0	27.0	24.0	150.0	150.0
Iron	Fe	0.60	2.00	0.55	0.20	0.30	0.25	0.27	0.40	0.40	0.30	<0.02
Chloride	Cl^-	59.0	26.0	29.0	47.0	14.0	49.0	12.0	5.00	7.00	6.00	13.0
Sulfate	SO	196.	141.	140.	112.	89.0	159.	103.	49.0	33.0	42.0	90.0
Phosphate	PO	20.7	12.3	8.80	3.60	<0.10	16.5	<0.10	<0.10	<0.10	<0.10	<0.10
Nitrate	NO^-	1380.	680.	1280.	570.	480.	1180.	390.	60.0	55.0	220.	440.
Ammonia (total)	NH°	2114.	457.	801.	282.	187.	967.	191.	1.30	0.04	11.6	137.

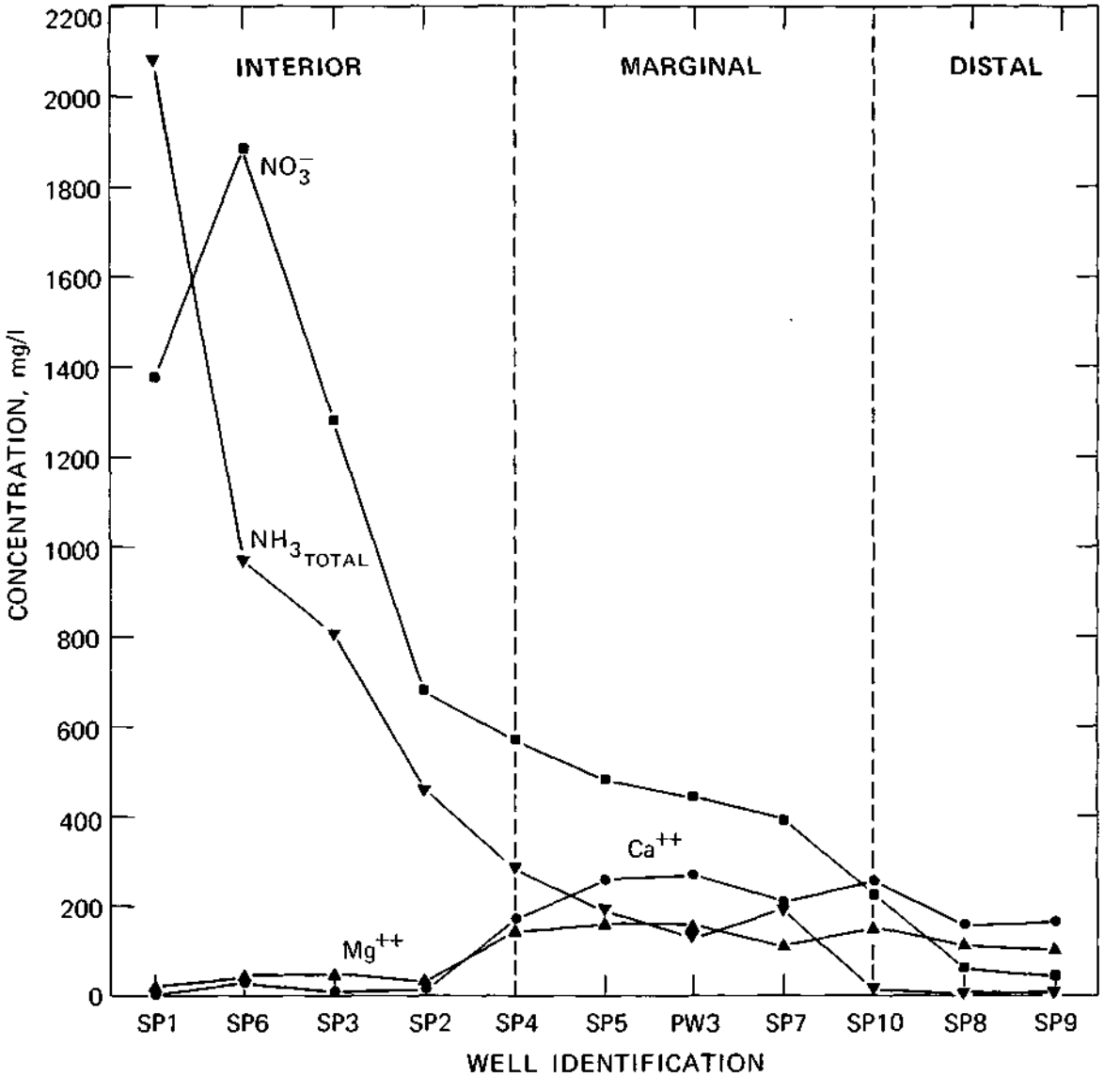


Figure 63. Partial chemical analysis from the 11 non-pumping control wells

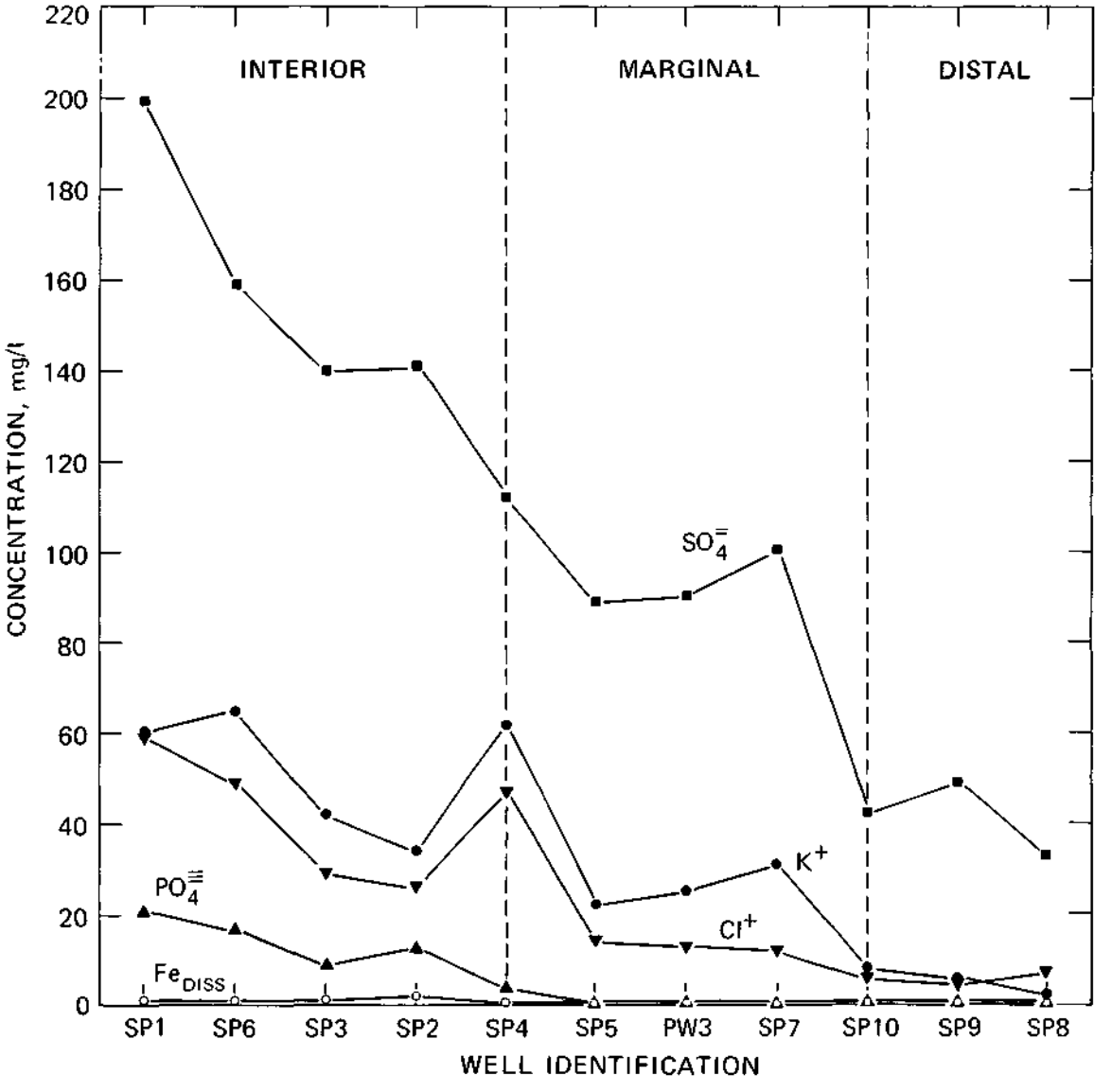


Figure 64. Partial chemical analysis from the 11 non-pumping control wells

A plume configuration was developed based on the chemical data. Considerations of chemical equilibria involved then provided insight into the transport and transformation of ammonia and minor constituents of the chemical fertilizer.

Contaminant Plume

The ammonia concentrations were used to determine a two-dimensional configuration for the plume. By contouring ammonia values (figure 65) two conclusions were drawn: 1) the plume was migrating in a generally western direction, and 2) part of the plume was being drawn into the cone of depression of PW4. Wells SP1, SP4, SP10, PW3, and PW4 were sampled periodically over nine months. Concentrations at SP1 averaged 2020 mg/l with a standard deviation of 79 mg/1 (N = 6). Concentrations in the other wells showed no significant change (<10 percent) in ammonia concentration during the same period. Periodic flushing events are probable, but at the scale of the study site, a steady-state groundwater quality condition prevailed.

Two additional investigations were carried out to improve the characterization of the plume: 1) a study of the source solubility, and 2) tracer experiments near the two production wells, PW3 and PW4. To assimilate the ammonia concentration in the groundwater beneath the source, samples of the solid fertilizer were brought into the laboratory and washed with distilled water, and successive washes were analyzed for total ammonia. Leachate concentrations averaged from 2000-2200 mg/1. A more elaborate leaching experiment would

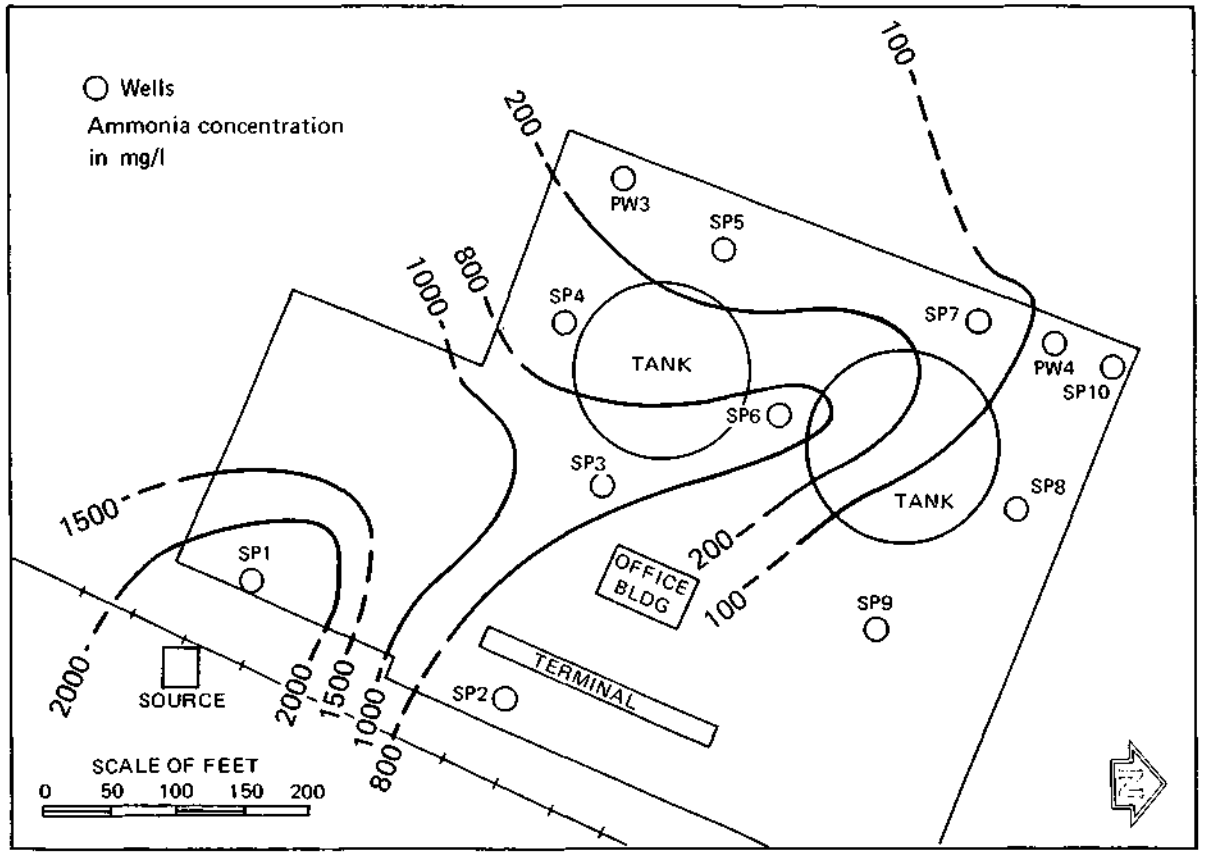


Figure 65. Ammonia concentrations (mg/l) in the groundwater at the field site

have revealed no more information because the variables affecting the leach rate at the source are unknowns. Therefore, concentrations encountered in the observation well nearest the source (SP1) were used as the source concentration.

Two tracer experiments were performed to determine groundwater flow rates near the production wells: one near PW4 at an injection distance of 50 ft (15.2 m) and one near PW3 at 82 ft (25 m) injection distance. Rhodamine WT, a fluorescent dye developed specifically for tracing work, was used because of its low adsorption on mineral and organic materials (Smart and Laidlaw, 1977; Aulenbach, Bull, and Middlesworth, 1978). The dye was introduced and flushed into SP4 and SP10 while PW3 and PW4 were producing 1000 gpm (5420 m^3/day) each. Wells PW3 and PW4 were sampled at 10-minute intervals. The initial breakthrough time of the dye was 150 minutes at well PW3 and 40 minutes at PW4. The peak concentration at PW4 occurred at 65 minutes, and the tracer concentration was still increasing slightly in PW3 after the termination of the experiment (200 minutes). Assuming that the interstitial velocity (V) can be calculated from the average arrival time of a nonreactive tracer, then V equals 0.77 ft/min (0.23 m/min) between SP10 and PW4.

The Meredosia Solute Transport Model

The model developed for the study area south of Meredosia, Illinois, covered an area of 56.7 acres (22.95 ha). The model (figure 66) was 1770 ft (539.6 m) along the x-axis, roughly north-south, and 1395 ft (425.3 m) along the y-axis. It contained 61 cells in the x-direction and 36 cells in the y-direction, progressively increasing in size beyond the site toward the north and west boundaries of the model.

The boundary conditions for the groundwater flow portion of the model consisted of fixed head and flux conditions. Head conditions were taken from the water table map which was constructed from field data, and flux conditions used along the north and west boundaries were calculated from gradients in the unconsolidated aquifer material. A flux withdrawal of 1000 gpm (5420 m^3/day) was used at PW4. Calibration of the flow portion of the model was not difficult because of the high density of data in this small area. From the contoured data, fixed head values were used on the model boundaries. The flow model was calibrated and used in the steady-state mode. The steady-state flow computation was chosen because: 1) there was no appreciable change in the concentration of contaminants in the groundwater throughout the study time (9 months), thus reducing the importance of flushing events at the source, and 2) the hydraulic gradient remained constant during the study time. The second condition seems reasonable, with only 5 ft (1.54 m) of topographic relief within the site and a hydraulic conductivity of about 3000 gpd/ft (1.42×10^{-3} m/sec) in the unconsolidated material.

The boundary conditions for solute transport were: 1) a constant influx at the source of 2000 mg/l, as calibrated against observed ammonia concentrations; and 2) sink conditions at PW4 and along the down-gradient boundaries (north and west), in order to observe the ammonia concentrations in the groundwater exiting from the model boundaries.

Ammonia was used as the contaminant plume indicator and for the purpose of transport calibration. Figure 67 demonstrates the nature of computer output after 400 days of transport simulation. The numbers within the model boundaries represent the number of particles residing in each cell, with one particle per cell equaling a concentration of 200 mg/1 ammonia. As shown, the plume has migrated mainly in the down-gradient direction, but it also has been drawn toward production well PW4. The larger numbers of particles near

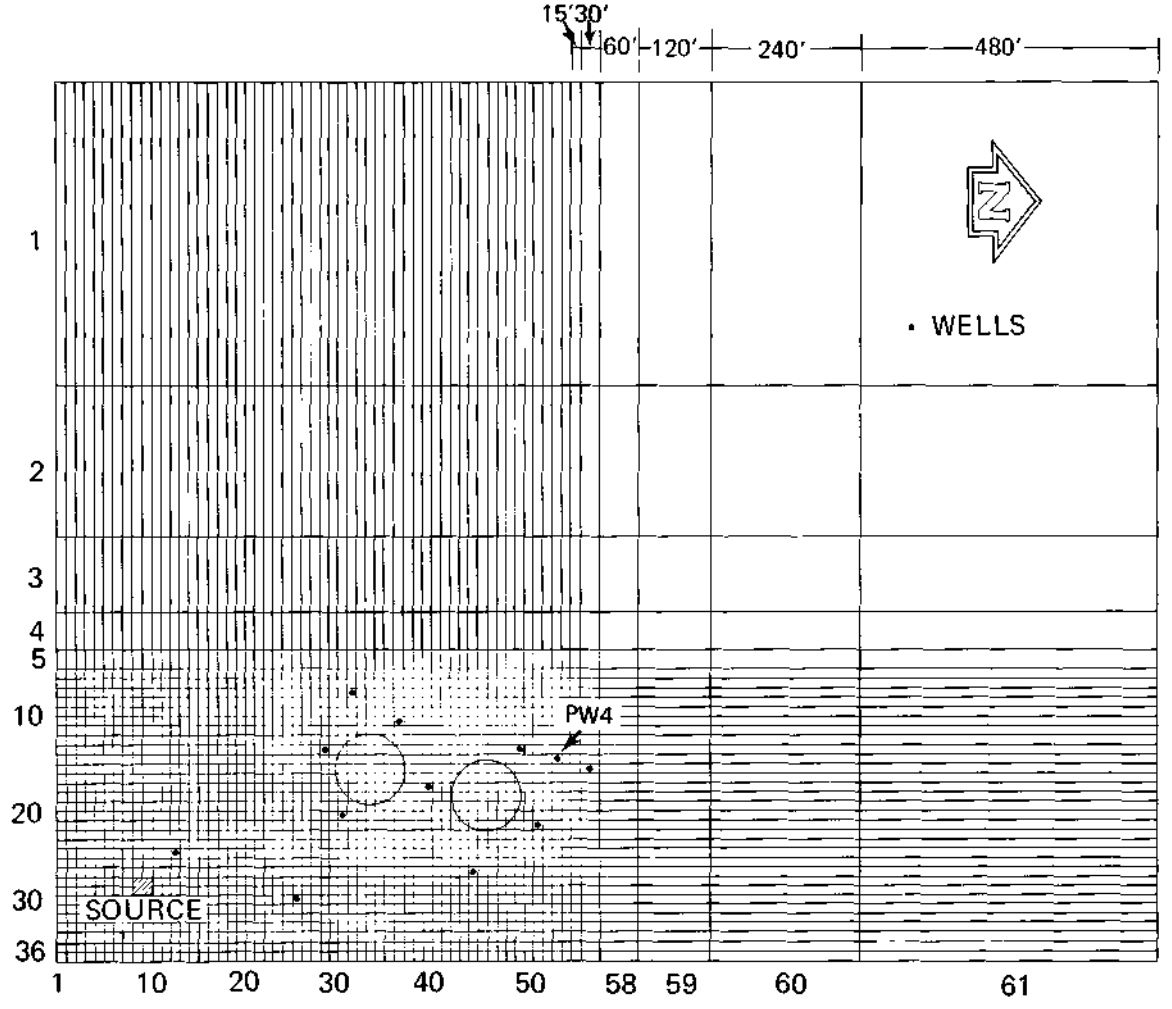


Figure 66. Particle cells superposed over the study area

the west boundary reflect the increase in cell size. The model was calibrated to known ammonia concentrations with a series of simulations in which the longitudinal and transverse dispersivity were varied.

Solute Transport Simulations

The approach used in the transport simulations was: 1) to calibrate the model and to investigate the unknowns--longitudinal and transverse dispersivity, 2) to obtain the steady plume configuration within the boundaries of the model, and 3) to investigate the most feasible remedial action scheme to flush the aquifer of the undesirable water.

The first step in the transport model calibration involved sorting out the parameters to which the model was sensitive. This was done within the range of uncertainty of the parameters at this particular site. Most of the parameters were well-known because of the existing data and field experiments. The model was most sensitive to longitudinal and transverse dispersivity. Even the effects from varying the recharge rate were small when compared to those caused by dispersivity variation. Longitudinal dispersivity (d_L) was varied from 0.0 to 11 ft (3.35 m) and transverse dispersivity (d_T) from 0.0 to 6.5 ft (1.98 m). The most reasonable ranges for calibration to the field-measured plume geometry were 7 ft

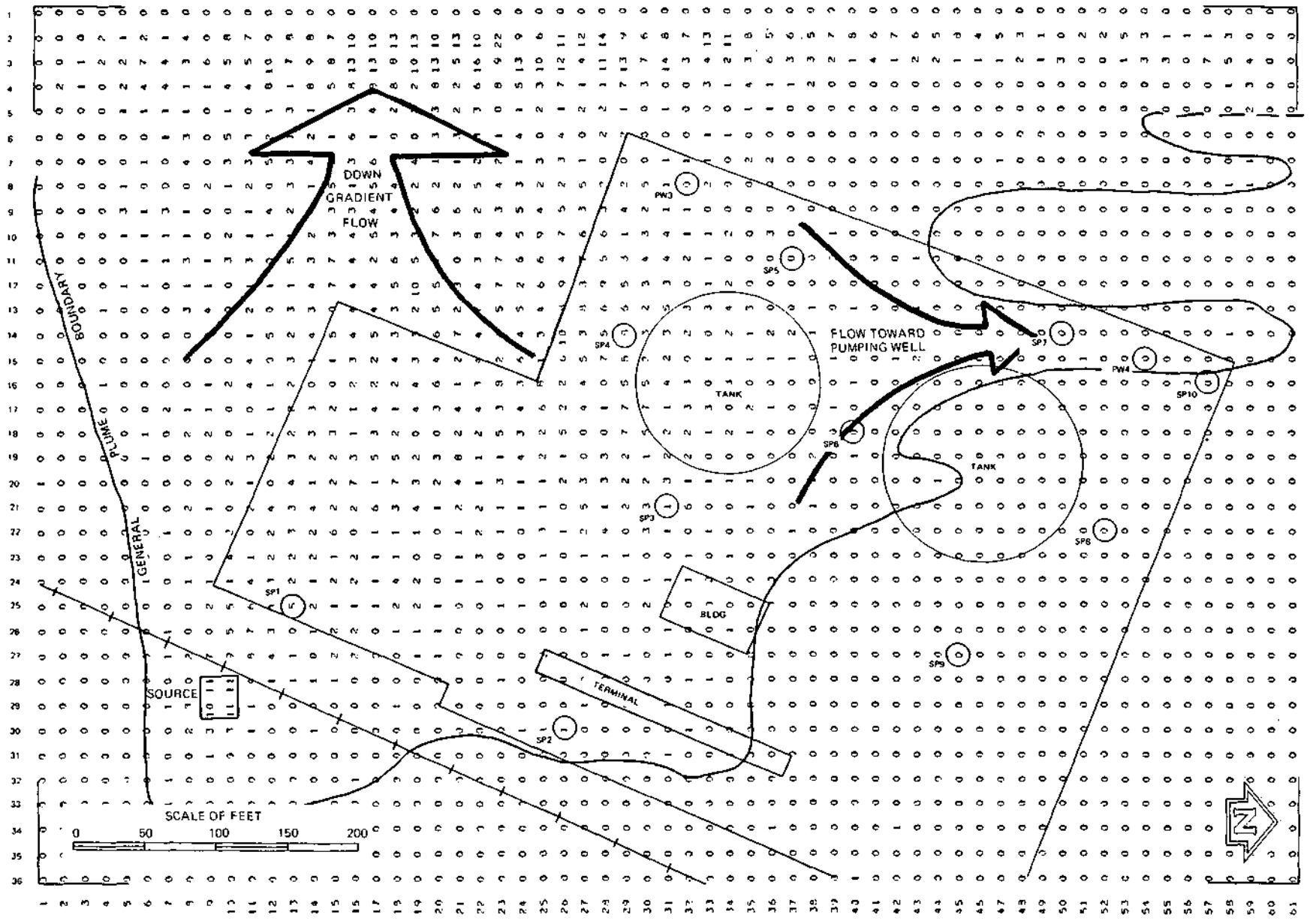


Figure 67. Distribution of particles residing in the model at 400 days of simulation

(2.13 m) < d_L < 11 ft (3.35 m) and 2 ft (0.61 m) < d_T < 3 ft (0.915 m).

The second issue addressed was whether or not the plume had reached a steady-state configuration within the boundaries of the model or was increasing in concentration with time toward the down-gradient boundaries. Long-term (1500 days) simulations were made within the range of reasonable d_L and d_T values, and the concentration of ammonia entering the sink boundaries was plotted against time. A steady concentration appeared after about 360 days at the down-gradient boundaries and production well using the smallest reasonable d_L, and d_T values, 7 ft (2.13 m) and 2 ft (0.61 m), respectively. This represents the upper limit or longest time it would take the contaminants to establish a constant plume geometry within the boundaries of the model. In other words, the contamination problem will remain the same at the site until a source or sink boundary condition is altered.

The most likely change in the boundary conditions and the one that would produce the most desirable results is removal of the source. This was investigated using the model and evaluated in terms of ammonia concentration at the down-gradient boundaries and the production well (PW4). With a steady-state plume condition existing at 360 days from the start of the simulation, the source flux was removed from the model at the next time step, 390 days. The ammonia concentrations at the sinks diminished to nearly zero at 810 days (figure 68). Therefore, 420 days is approximately the time it would take for the study area and production well to naturally flush itself of the contaminant.

The application of solute transport modeling to this groundwater contamination problem has been useful in determining the extent of the problem. It seems unlikely that the present condition will worsen at the study site, and natural processes should flush the plume within fourteen months of source removal.

As part of continuing investigations, the effect of the contaminant plume at down-gradient wells will be emphasized. Additional sampling and modeling efforts will serve to further define controlling subsurface processes and to improve the usefulness of the approach.

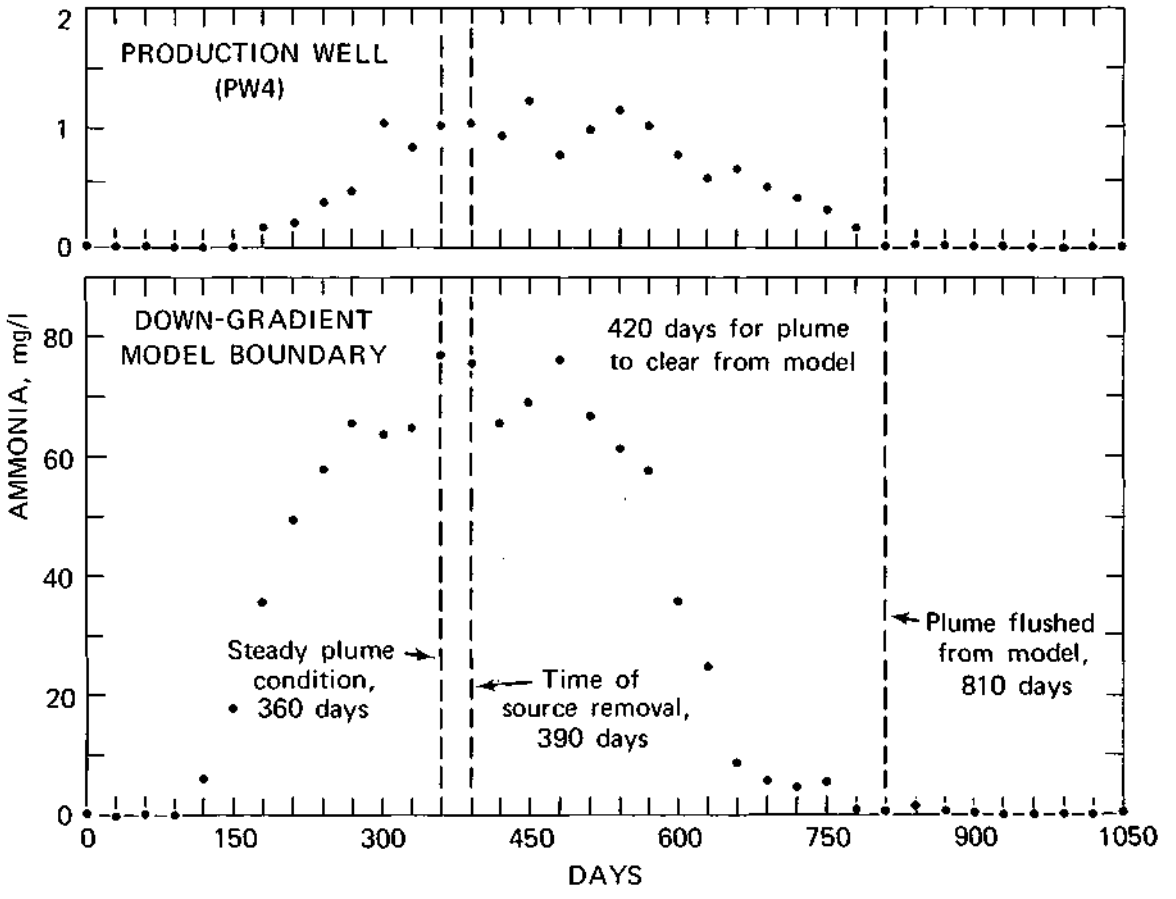


Figure 68. Ammonia concentration at the production well and down-gradient boundary after source removal

REFERENCES

Aulenbach, D. B., J. H. Bull, and B. C. Middlesworth. 1978. *Use of tracers to confirm groundwater flow.* Ground Water, v. 16(3): 149-157.

Baetsle, L. H. 1967. *Computational methods for the prediction of underground movement of radio-nuclids.* Journal of Nuclear Safety, 8(6):576-588.

Baetsle, L. H. 1969. *Migration of radionuclide in porous media.* Progress in Nuclear Energy, Series XII, Health Physics, A.M.F. Duhamel, Ed., Pergamon Press, Elmsford, NY, pp. 707-730.

Bear, Jacob. 1972. *Dynamics of fluids in porous media.* American Elsevier Publishing Company, New York, New York, 764 pages.

Borg, I. Y., R. Stone, H. B. Levy, and L. D. Ramspott. 1976. *Information pertinent to the migration of radionuclides in groundwater at the Nevada Test Site, Part 1: Review and analysis of existing information.* Lawrence Livermore Laboratory, Report UCRL 52078, 216 p.

Freeze, R. A., and J. A. Cherry. 1979. *Groundwater.* Prentice-Hall, Inc., 604 p.

Fried, Jean J. 1975. *Groundwater pollution.* Elsevier Scientific Publishing Company, Amsterdam, 330 p.

Heigold, P. C, and R. W. Ringler. 1979. A seismic *refraction survey of the lower Illinois Valley bottomlands.* Illinois State Geological Survey Circular 507, 18 p.

Helfferich, F. 1962. *Ion exchange.* McGraw-Hill, New York.

Higgins, G. H. 1959. *Evaluation of the groundwater contamination hazard from underground nuclear explosives.* Journal of Geophysical Research, 64: 1509-1519.

Jacob, C. E. 1950. *Flow of groundwater. In* Engineering Hydraulics, Hunter Rouse, Ed., John Wiley & Sons, Inc., New York.

Kimmel, G. E.. and O. C. Braids. 1980. *Leachate plumes in ground water from Babylon and Islip Landfills, Long Island, New York.* U.S.G.S. Prof. Paper 1085, Department of the Interior, Superintendent of Documents, No. I19.16:1085.

Kirkham, D., and S. B. Affleck. 1977. *Solute travel times to* welts. Ground Water Journal, v. 15(3) :231-242.

Mood, A. M., and F. A. Graybill. 1963. *Introduction to the theory of statistics.* McGraw-Hill Book Company, Inc., New York, New York.

Muskat, M. 1939. *The flow of homogeneous fluids through porous media.* McGraw-Hill Book Company Inc., 763 p.

Ogata, A. 1970. *Theory of dispersion in granular medium.* U.S.G.S. Prof. Paper 411-I.

Prickett, T. A., and C. G. Lonnquist. 1971. *Selected digital computer techniques for groundwater resource evaluation.* Illinois State Water Survey Bulletin 55, 66 p.

Smart, P. L., and I. M. S. Laidlaw. 1977. *An evaluation of some fluorescent dyes for water tracing.* Water Resources Research, v. 13(1): 15-33.