PB85-214450

PLUME 2D: TWO-DIMENSIONAL PLUMES IN UNIFORM GROUND WATER FLOW

Oklahoma State University
Stillwater, OK

Jun 85

U.S. DEPARTMENT OF COMMERCE
National Technical Information Service
NTIS

EPA/600/2-85/065
June 1985

PLUME2D

TWO-DIMENSIONAL PLUMES IN UNIFORM GROUND WATER FLOW

by

Jan Wagner and Stephanie A. Watts
School of Chemical Engineering
Oklahoma State University
Stillwater, Oklahoma 74078

Douglas C. Kent
Department of Geology
Oklahoma State University
Stillwater, Oklahoma 74078

CR811142

Project Officer

Carl G. Enfield
Robert S. Kerr Environmental Research Laboratory
U. S. Environmental Protection Agency
Ada, Oklahoma 74820

ROBERT S. KERR ENVIRONMENTAL RESEARCH LABORATORY
OFFICE OF RESEARCH AND DEVELOPMENT
U.S. ENVIRONMENTAL PROTECTION AGENCY
ADA, OK 74820

TECHNICAL REPORT DATA

(Please read Instructions on the reverse before completing)

1. REPORT NO.	2.	3. RECIPIENT'S ACCESSION NO.
EPA/600/2-85/065		PB85 [illegible] /AS

4. TITLE AND SUBTITLE	5. REPORT DATE
PLUME2D: TWO-DIMENSIONAL PLUMES IN UNIFORM GROUND WATER FLOW	June 1985
	6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S)	8. PERFORMING ORGANIZATION REPORT NO.
Jan Wagner, Stephanie Watts, Douglas Kent	
9. PERFORMING ORGANIZATION NAME AND ADDRESS	10. PROGRAM ELEMENT NO.
Oklahoma State University Stillwater OK 74078	ABRD1A
	11. Coop. Agr. CR-811142
12. SPONSORING AGENCY NAME AND ADDRESS	13. TYPE OF REPORT AND PERIOD COVERED
Robert S. Kerr Environmental Research Laboratory Office of Research and Development U.S. Environmental Protection Agency Ada, OK 74820	Final 9/83 - 2/85
	14. SPONSORING AGENCY CODE EPA/600/15

15. SUPPLEMENTARY NOTES

Carl G. Enfield, Project Officer

16. ABSTRACT

A closed-form analytical solution for two dimensional plumes was incorporated in an interactive computer program. The assumption of an infinite aquifer depth and uniform source mass rate and source location was overcome by using the principal of superposition in space and time. The source code was written in a subset of FORTRAN 77 and can be compiled with FORTRAN IV, FORTRAN 66 as well as FORTRAN 77. As a result, the code is nearly independent of hardware and operating system. The model can be solved for either vertically or horizontally averaged conditions.

17. KEY WORDS AND DOCUMENT ANALYSIS

a. DESCRIPTORS	b. IDENTIFIERS/OPEN ENDED TERMS	c. COSATI Field/Group

18. DISTRIBUTION STATEMENT	19. SECURITY CLASS *(This Report)*	21. NO. OF PAGES
RELEASE TO PUBLIC	UNCLASSIFIED	95
	20. SECURITY CLASS *(This page)*	22. PRICE
	UNCLASSIFIED	

EPA Form 2220-1 (Rev. 4-77) PREVIOUS EDITION IS OBSOLETE

DISCLAIMER

The information in this document has been funded wholly or in part by the United States Environmental Protection Agency under assistance agreement number CR-811142 to Oklahoma State University. It has been subject to the Agency's peer and administrative review, and it has been approved for publication as an EPA document.

FOREWORD

EPA is charged by Congress to protect the Nation's land, air, and water systems. Under a mandate of national environmental laws focused on air and water quality, solid waste management and the control of toxic substances, pesticides, noise, and radiation, the Agency strives to formulate and implement actions which lead to a compatible balance between human activities and the ability of natural systems to support and nurture life.

The Robert S. Kerr Environmental Research Laboratory is the Agency's center of expertise for investigation of the soil and subsurface environment. Personnel at the Laboratory are responsible for management of research programs to: (a) determine the fate, transport and transformation rates of pollutants in the soil, the unsaturated zone and the saturated zones of the subsurface environment; (b) define the processes to be used in characterizing the soil and subsurface environment as a receptor of pollutants; (c) develop techniques for predicting the effect of pollutants on ground water, soil and indigenous organisms; and (d) define and demonstrate the applicability and limitations of using natural processes, indigenous to the soil and subsurface environment, for the protection of this resource.

This project was initiated to develop an interactive computer model which could be utilized to predict toxic chemical fate in homogeneous aquifers. The model should be useful in making comparisons, between chemicals, for idealized homogeneous aquifers. This model is not intended for addressing site specific problems where there is significant heterogeneity in the aquifer.

Clinton W. Hall
Director
Robert S. Kerr Environmental
Research Laboratory

ABSTRACT

A closed-form analytical solution for two dimensional plumes was incorporated in an interactive computer program. The assumption of an infinite aquifer depth and uniform source mass rate and source location was overcome by using the principal of superposition in space and time. The source code was written in a subset of FORTRAN 77 and can be compiled with FORTRAN IV, FORTRAN 66 as well as FORTRAN 77. As a result, the code is nearly independent of hardware and operating system. The model can be solved for either vertically or horizontally averaged conditions.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

Relatively simple analytical methods can often be used to evaluate ground-water contamination problems, depending upon the complexity of the system and the availability of field data. Analytical models can also serve as valuable tools in developing parameters for more sophisticated numerical models. Although the numerical evaluation of an analytical solution to a ground-water problem may be mathematically complex, analytical models are well suited for interactive use on digital computers. Many analytical solutions to ground-water contamination problems can be coded on programmable hand-held calculators. In general, very few input parameters are required to define a given problem and numerical results can be calculated in a few seconds.

This report presents analytical solutions to two ground-water pollution problems -- two-dimensional plumes in uniform ground-water flow. An interactive computer code has been developed which enables the user to modify the definition of a given problem, and thus gain some insight into the effects of various parameters on the extent of a contaminant plume.

SECTION I

MATHEMATICAL DEVELOPMENT

The differential equation describing the conservation of mass of a component in a saturated, homogeneous aquifer with uniform, steady flow in the x-direction can be written as

$$\frac{\partial C_T}{\partial t} + V \frac{\partial (C)}{\partial x} = D_x \frac{\partial^2(\Theta C)}{\partial x^2} + D_y \frac{\partial^2(\Theta C)}{\partial y^2} + D_z \frac{\partial^2(\Theta C)}{\partial z^2} - r_t \qquad (1)$$

where

C	=	component mass per unit of fluid phase	M/L^3
C_T	=	total component mass per unit volume of aquifer	M/L^3
D_x	=	dispersion coefficient in x-direction	L^2/t
D_y	=	dispersion coefficient in y-direction	L^2/t
D_z	=	dispersion coefficient in z-direction	L^2/t
r_t	=	rate of degradation of mass per unit volume of aquifer	M/L^3t
V	=	Darcy, or seepage, velocity in the x-direction	L/t
$x,y,z,$	=	rectangular coordinates at the point of interest	L
Θ	=	porosity of porous media	L^3/L^3

The total mass of a component per unit volume of aquifer is distributed as dissolved solute in the fluid phase and adsorbed solute on the solid matrix. Let

C_s = component mass per unit mass of solid M/M

and

ρ_B = bulk density of the aquifer, or the mass of solids per unit volume of the aquifer M/L^3.

The total component mass per unit volume of aquifer can be expressed as

$$\frac{\text{Mass}}{\text{Unit Volume}} = \frac{\text{Volume of voids}}{\text{Unit Volume of aquifer}} \; \frac{\text{Component Mass}}{\text{Volume of voids}}$$

$$+ \frac{\text{Mass of solids}}{\text{Unit volume of aquifer}} \; \frac{\text{Component Mass}}{\text{Mass of solids}}$$

or

$$C_T = \Theta C + \rho_B C_s \tag{2}$$

and, the rate of accumulation of mass in the aquifer becomes

$$\frac{\partial C_T}{\partial t} = \Theta \frac{\partial C}{\partial t} + \rho_B \frac{\partial C_s}{\partial t} \tag{3}$$

In general, $C_s = f(C)$ and

$$\frac{\partial C_s}{\partial t} = \frac{dC_s}{dC} \frac{\partial C}{\partial t} \tag{4}$$

For a linear equilibrium adsorption isotherm,

$$\frac{dC_s}{dC} = K_d \; \frac{M/M}{M/L^3} \tag{5}$$

where K_d is a distribution constant.

The change in concentration per unit volume of porous media, $\partial C_T/\partial t$, can be written in terms of fluid phase concentration, C, by substituting Equations 4 and 5 into Equation 3. Therefore,

$$\frac{\partial C_T}{\partial t} = \Theta \frac{\partial C}{\partial t} + \rho_B K_d \frac{\partial C}{\partial t}$$

or

$$\frac{\partial C_T}{\partial t} + (\Theta + \rho_B K_d) \frac{\partial C}{\partial t} \qquad (6)$$

The rate of degradation of component mass per unit volume of porous media is also distributed between the solid and liquid phases, or

$$\frac{\text{Rate of mass degraded}}{\text{Unit volume of aquifer}} = \frac{\text{Rate of mass degraded}}{\text{Unit volume of fluid}} \; \frac{\text{Volume of fluid}}{\text{Volume of aquifer}}$$

$$+ \frac{\text{Rate of mass degraded}}{\text{Unit mass of solid}} \; \frac{\text{Mass of solid}}{\text{Volume of aquifer}}$$

Now, the rate of change in total mass per unit volume of aquifer due to reaction can be written as

$$r_t = \frac{\partial C_T}{\partial t} = \Theta \frac{\partial C}{\partial t} + \rho_B \frac{\partial C_s}{\partial t} \qquad (7)$$

The concentration on the solid, C_s, is related to the concentration in the liquid, C, through the linear adsorption isotherm assumed previously, and

$$r_t = \frac{\partial C_T}{\partial t} = (\Theta + \rho_B K_d) \frac{\partial C}{\partial t} \qquad (8)$$

Assuming first order decay kinetics, the rate of decrease in fluid phase and solid phase concentrations due to reaction can be expressed as

$$\frac{\partial C}{\partial t} = \lambda C \qquad (9)$$

and

$$\frac{\partial C_s}{\partial t} = \lambda C_s$$

respectively, where λ is a rate constant (1/t), and

$$r_t = (\Theta + \rho_B K_d)\lambda C \qquad (10)$$

Equation 1 can now be written in terms of the fluid concentration. Substituting Equations 6 and 10 and recalling that for a homogeneous porous medium the porosity, Θ, is constant, Equation 1 becomes

$$(1 + \frac{\rho_B}{\Theta} K_d) \frac{\partial C}{\partial t} + V^* \frac{\partial C}{\partial x} = D_x \frac{\partial^2 C}{\partial x^2} + D_y \frac{\partial^2 C}{\partial y^2} + D_x \frac{\partial^2 C}{\partial z^2} - (1 + \frac{\rho_B}{\Theta} K_d)\ \lambda C \qquad (11)$$

where $V^* \equiv \frac{V}{\Theta}$ is the average interstitial, or pore, velocity. Defining a "retardation coefficient" as

$$R_d = 1 + \frac{\rho_B}{\Theta} K_d \qquad (12)$$

the differential equation describing the conservation of mass in the aquifer becomes

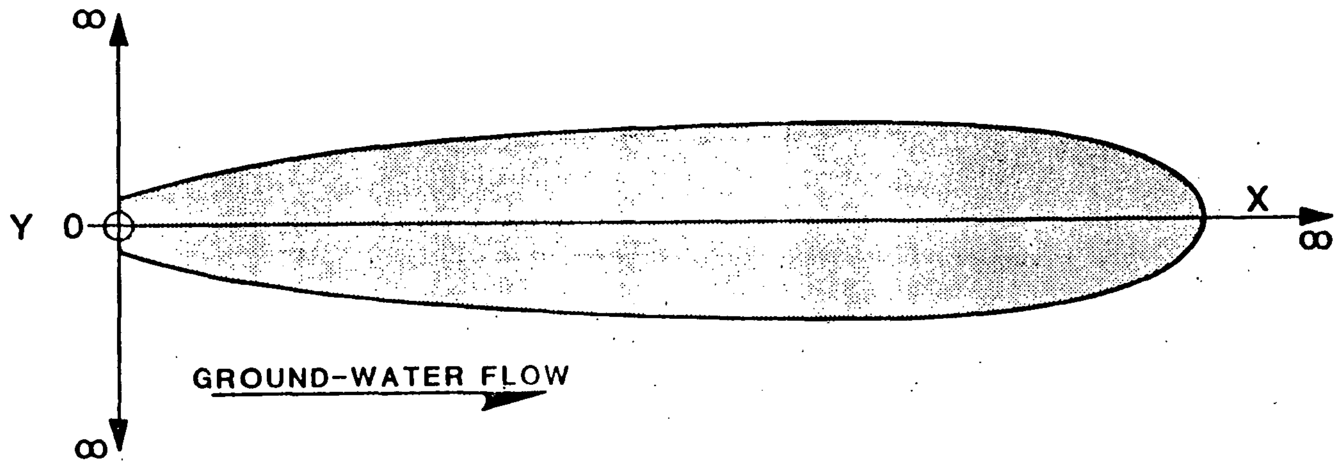


Figure 1. Coordinate system for vertically averaged solution.

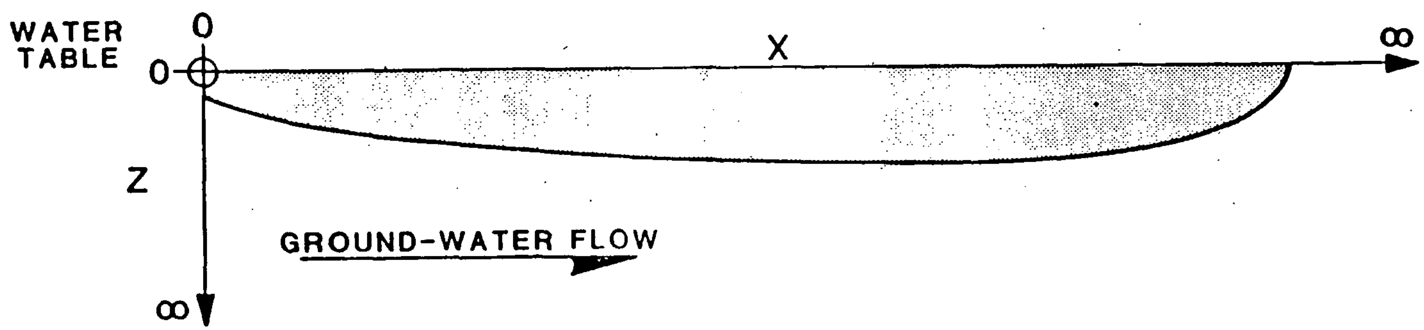


Figure 2. Coordinate system for horizontally averaged solution.

$$R_d \frac{\partial C}{\partial t} + V^* \frac{\partial C}{\partial x} = D_x \frac{\partial^2 C}{\partial x^2} + D_y \frac{\partial^2 C}{\partial y^2} + D_z \frac{\partial^2 C}{\partial z^2} - R_d \lambda C \quad (13)$$

Equation 13 is a linear partial differential equation which can be integrated analytically to yield an expression for concentration as a function of time and position.

Solutions of Equation 13 for two types of ground-water contamination problems are presented in the following paragraphs. The first is a vertically-averaged solution which describes a contaminant plume in the x-y plane (Figure 1). The second is a horizontally-averaged solution, describing a contaminant plume in the x-z plane (Figure 2).

Vertically-averaged solution. The vertically-averaged solution applies to a homogeneous aquifer of infinite aerial extent and finite depth. The contaminant is assumed to be well mixed over the saturated thickness. The source of contaminant is a vertical line source located at the origin of a coordinate system in the x-y plane. This conceptual model would apply to an injection well which fully penetates the saturated zone.

Wilson and Miller (1978) have also applied this solution downstream from a contaminant source at the surface of the water table. For a relatively thin saturated zone, vertical dispersion will result in mixing vertically. The concentration distribution can be considered as being two-dimensional in a horizontal plane at distances downstream of the source sufficient for the concentration distribution to become uniform with depth. Mathematically, the problem is treated as an infinite aquifer with a line source at the origin. The vertically-averaged formulation of Equation 13 is

$$R_d \frac{\partial C}{\partial t} + V^* \frac{\partial C}{\partial x} = D_x \frac{\partial^2 C}{\partial x^2} + D_y \frac{\partial^2 C}{\partial y^2} - R_d \lambda C \quad (14)$$

The boundary conditions can be stated mathematically as follows

$$C(x,y,0) = 0 \tag{15a}$$

$$C(x,\pm\infty,t) = 0 \tag{15b}$$

$$C(\pm\infty,y,t) = 0 \tag{15c}$$

A solution to Equation 14 with the Equation 15 boundary conditions and a continuous source of strength C_oQ' can be written as (Hunt, 1978; Wilson and Miller, 1978):

$$C = \frac{C_o Q' \, EXP\left(\frac{V^* x}{2D_x}\right)}{4\pi\Theta \, (D_x D_y)^{0.5}} \; W(U,B) \tag{16}$$

where

$$U = \frac{\left(\frac{Vx}{D_x}\right)^2 + \frac{D_x}{D_y}\left(\frac{V^* y}{D_x}\right)^2}{\frac{4 V^{*2} t}{R_d D_x}} \tag{17}$$

and

$$B = \frac{1}{2}\left[\left(\frac{V^* x}{D_x}\right)^2 + \frac{D_x}{D_y}\left(\frac{V^* y}{D_x}\right)^2\right]^{1/2}\left[1 + \frac{4D_x R_d \lambda}{V^{*2}}\right]^{1/2} \tag{18}$$

and C_0Q' (M/t/L) is the contaminant source rate per unit depth of the saturated zone.

The function W(U,B) is defined as

$$W(U,B) = \int_U^\infty \frac{1}{\xi} \text{EXP} \left(-\xi - \frac{B^2}{4\xi}\right) d\xi \qquad (19)$$

where ξ is a dummy integration variable. This function is often referred to as the "well function for leaky artesian aquifers" (Hantush; 1956, 1964).

The steady-state solution of Equations 14 and 15 can be obtained by noting as $t \rightarrow \infty$, $U \rightarrow 0$ and the well function (Hantush, 1956) can be expressed as

$$W(0,B) = 2K_0(B) \qquad (20)$$

where $K_0(B)$ is the modified Bessel function of the second kind of order zero. At steady-state the vertically-averaged solution can be written as

$$C = \frac{C_0Q' \ \text{EXP} \left(\frac{V^* x}{2D_x}\right)}{2\pi\theta \ (D_x D_y)^{0.5}} \ K_0(B) \qquad (21)$$

The units of the variables in Equations 16 and 21 can be eliminated by defining the following dimensionless groups:

$$\text{Modified Peclet Numbers} \sim \frac{\text{Convective mass transport}}{\text{Dispersive mass transport}}$$

$$Pe_x = \frac{V^* x}{D_x} \tag{22}$$

$$Pe_y = \frac{V^* y}{D_x} \tag{23}$$

Damkohler Group II $\sim \frac{\text{Mass decay rate}}{\text{Mass dispersion rate}}$

$$D_k = \frac{D_x R_d \lambda}{V^{*2}} \tag{24}$$

Number of Pore Volumes Injected $\sim \frac{\text{Mass transport rate}}{\text{Mass accumulation rate}}$

$$I = \frac{V^{*2} t}{R_d D_x} \tag{25}$$

Dimensionless Source Term $\sim \frac{\text{Mass injection rate}}{\text{Mass diffusion rate}}$

$$\Gamma = \frac{Q'}{\Theta(D_x D_y)^{0.5}} \tag{26}$$

Dimensionless Concentration

$$\gamma = \frac{C}{C_o} \tag{27}$$

Note that the number of pore volumes injected can be written as

$$I = \frac{V^2 t}{D_x R_d} = \left(\frac{V^* L}{D_x}\right)^2 \frac{D_x t}{R_d L^2} \tag{28}$$

where L is a characteristic length defined as

$$L^2 = x^2 + \frac{D_x}{D_y} y^2 \tag{29}$$

The first group on the right-hand-side of Equation 28 is the Modified Peclet number

$$Pe_{xy} = \left[\left(\frac{V^* x}{D_x}\right)^2 + \frac{D_x}{D_y} \left(\frac{V^* y}{D_x}\right)^2\right]^{1/2}$$

or

$$Pe_{xy} = (Pe_x^{\,2} + \beta\, Pe_y^{\,2})^{1/2} \tag{30}$$

where

$$\beta = \frac{D_x}{D_y}$$

The second group on the right-hand-side of Equation 28 is a dimensionless time variable,

$$\tau = \frac{D_x t}{R_d L^2} \tag{31}$$

The transient and steady-state solutions to Equations 14 which are given by Equations 16 and 21 can be written in terms of the dimensionless variables defined above. The transient solution is

$$\gamma = \frac{\Gamma}{4\pi} \text{EXP}(\frac{1}{2} Pe_x) \; W(U,B) \tag{32}$$

and at steady state

$$\gamma = \frac{\Gamma}{2\pi} \text{EXP} (\frac{1}{2} Pe_x) \; K_0(B) \tag{33}$$

with

$$U = \frac{Pe_{xy}^{2}}{4I} \tag{34}$$

and

$$B = \frac{1}{2} Pe_{xy}(1 + 4D_k)^{\frac{1}{2}} \tag{35}$$

The values of dimensionless concentrations evaluated using Equation 32 or Equation 33 are valid for any consistent set of units. Using dimensionless variables also tends to "scale" numerical values when working in various systems of units.

Horizontally-averaged solution. Consider a homogeneous aquifer with a continuous line source of infinite length located at the water table and normal to the direction of ground-water flow as shown in Figure 2. In other words the tracer is assumed to be well mixed over the width of the aquifer. A problem which might fit this conceptual model is seepage from a trench perpendicular to the direction of ground-water flow.

The horizontally-averaged formulation of Equation 13 is

$$R_d \frac{\partial C}{\partial t} + V^* \frac{\partial C}{\partial x} = D_x \frac{\partial^2 C}{\partial x^2} + D_z \frac{\partial^2 C}{\partial z^2} - R_d \lambda C \tag{36}$$

For an aquifer of infinite depth and a uniform continuous line source, the appropriate boundary conditions can be written as follows

$$C(x,z,0) = 0 \tag{37a}$$

$$C(\pm\infty,z,t) = 0 \tag{37b}$$

$$C(x,\infty,t) = 0 \tag{37c}$$

$$\frac{\partial C(x,0,t)}{\partial z} = 0 \tag{37d}$$

A solution to Equation 36 with the Equation 37 boundary conditions and a continuous line source of strength C_oQ' is

$$C = \frac{C_oQ' \; EXP(\frac{V^* x}{2D_x})}{2\pi\Theta \; (D_xD_z)^{0.5}} \; W(U,B) \tag{38}$$

At steady state the horizontally-averaged solution can be written as

$$C = \frac{C_oQ' \; EXP \; (\frac{V^* x}{2D_x})}{\pi\theta \; (D_xD_z)^{0.5}} \; K_o(B) \tag{39}$$

where

$$U = \frac{\left(\frac{V^* x}{D_x}\right)^2 + \frac{D_x}{D_z}\left(\frac{V^* z}{D_x}\right)^2}{\frac{4 V^{*2} t}{R_d D_x}} \qquad (40)$$

$$B = \frac{1}{2}\left[\left(\frac{V^* x}{D_x}\right)^2 + \frac{D_x}{D_z}\left(\frac{V^* z}{D_x}\right)^2\right]^{1/2}\left[1 + \frac{4 D_x R_d \lambda}{V^{*2}}\right]^{1/2} \qquad (41)$$

and Q' ($L^3/t/L$) is the volumetric contaminant source rate per unit width of the aquifer (or unit length of the line source).

Changing subscripts, the definition of the dimensionless groups leads to

$$Pe_z = \frac{V^* z}{D_x} \qquad (42)$$

and

$$\Gamma = \frac{Q'}{\Theta(D_x D_z)0.5} \qquad (43)$$

with

$$Pe_{xz} = (Pe_x{}^2 + \beta\, Pe_z{}^2)^{1/2} \qquad (44)$$

where

$$\beta = \frac{D_x}{D_z} \qquad (45)$$

By substituting the dimensionless groups described in vertically-averaged solution and those defined above, Equations 38 through 41 can be written in terms of dimensionless variables.

The transient solution becomes

$$\gamma = \frac{\Gamma}{2\pi} \mathrm{EXP}\left(\frac{1}{2} Pe_x \right) W(U,B) \tag{46}$$

and at steady state, the horizontally-averaged solution is

$$\gamma = \frac{\Gamma}{\pi} \mathrm{EXP} \left(\frac{1}{2} Pe_x\right) K_o(B) \tag{47}$$

where

$$U = \frac{{Pe_{xz}}^2}{4I} \tag{48}$$

and

$$B = \frac{1}{2} Pe_{xz}(1 + 4D_k)^{\frac{1}{2}} \tag{49}$$

The similarity of the solutions of the vertically-averaged and horizontally-averaged problems facilitates their numerical evaluation using a common computational algorithm. For the same numerical values of the independent variables, concentration values for the horizontally-averaged solutions are obtained by doubling the vertically-averaged solution values.

Assumptions and Limitations

Equations 32-33 and 46-47 can be used to calculate the concentrations in leachate plumes under the following assumptions and limitations:

1. The ground-water flow regime is completely saturated.
2. All aquifer properties are constant and uniform throughout the aquifer.
3. All ground-water flow is horizontal, continuous, and uniform throughout the aquifer.
4. The aquifer is infinite in extent for the vertically-averaged solution, or semi-finite in extent for the horizontally-averaged solution.
5. The leachate source is a line located at the origin of the coordinate system.
6. The mass flow rate of the source is constant.
7. At zero time the concentration of leachate in the aquifer is zero.

The assumptions of an infinite aquifer depth and a uniform source mass rate can be overcome by using the principles of superposition in space and time, respectively (Walton, 1962). Both of these provisions have been incorporated in the computer program described in the next section.

Superposition

The differential equation describing component mass concentration in a porous medium, Equation 1, is a linear partial differential equation. The principal of superposition can be used directly to solve complex ground-water contamination problems in terms of the simplier solutions described above. Unfortunately, the scattered applications of this principle are not explained in any single reference. Some texts indicate that superposition means that any sum of solutions is also a solution. Superposition is commonly used to generate a linear no-flow boundary condition through the use of "image wells" or to simulate multiple sources and sinks (Walton; 1962, 1970). The principle of superposition is also complicated by referring to the "Duhamel theorem," the "Faltung integral," and/or "convolution integrals." These terms often have no apparent physical interpretation. For the purposes of this report, "superposition in space" will refer to the approximation of sources of finite area or volume as the sum of a finite number of point sources or the generation of no-flow boundaries using image wells. "Superposition in time" will refer to the approximation of a variable source rate of contamination as the sum of a finite number of constant source rates distributed in time.

Both the horizontally-averaged and vertically-averaged solutions can be used to simulate aquifers of finite width or depth, respectively, or plane sources of finite width. Applications of this type require a thorough understanding of the physical interpretation of the principal of superposition.

Some applications are relatively straight forward, and the computer program provides for the approximation of a non-uniform source rate using superposition in time. Multiple sources and aquifers of finite thickness are also included using superposition in space.

Consider the variable source of contamination shown in Figure 3. The solutions of the governing differential equation presented in this report are of the form

$$C(x,z,t) = C_o Q' \ f(x,z,t) = \dot{Q}' \ f(x,z,t) \qquad (50)$$

where $\dot{Q}'$ is the source mass rate per unit length. The principle of superposition in time can be written for any position as

$$C(x,z,t) = \sum_{i=1}^{n} \dot{Q}_i' \; f(x,z,t_i) \tag{51}$$

Now, the variable rate schedule shown in Figure 3a can be decomposed into a series of positive and negative mass rates as shown in Figure 3b. The concentration at a point x,y,z at the end of the simulation, t_s, can be evaluated as

$$\begin{aligned} C(x,y,z,t) = {} & \dot{Q}_1' \; f(x,y,z,t_1) - \dot{Q}_1' \; f(x,y,z,t_2) \\ & + \dot{Q}_2' \; f(x,y,z,t_2) - \dot{Q}_2' \; f(x,y,z,t_3) \\ & + \dot{Q}_3' \; f(x,y,z,t_3) - \dot{Q}_3' \; f(x,y,z,t_4) \\ & + \dot{Q}_4' \; f(x,y,z,t_4) \end{aligned} \tag{52}$$

In general terms

$$C(x,y,z,t_s) = \sum_{i=1}^{n} (\dot{Q}_i' - \dot{Q}_{i-1}') \; f(x,y,z,t_i) \tag{53}$$

with $\dot{Q}_0' = 0$

Note the time corresponding to a given source rate, t_i, is the period beginning with the start of the given rate to the end of the simulation period; <u>time is not the duration of a given rate</u>. For ease of application, Equation 53 can be rewritten as

$$C(x,y,z,t_s) = \sum_{k=1}^{n} (\dot{Q}'_k - \dot{Q}'_{k-1}) \; f(x,y,z,t_s - t_{k-1}) \tag{54}$$

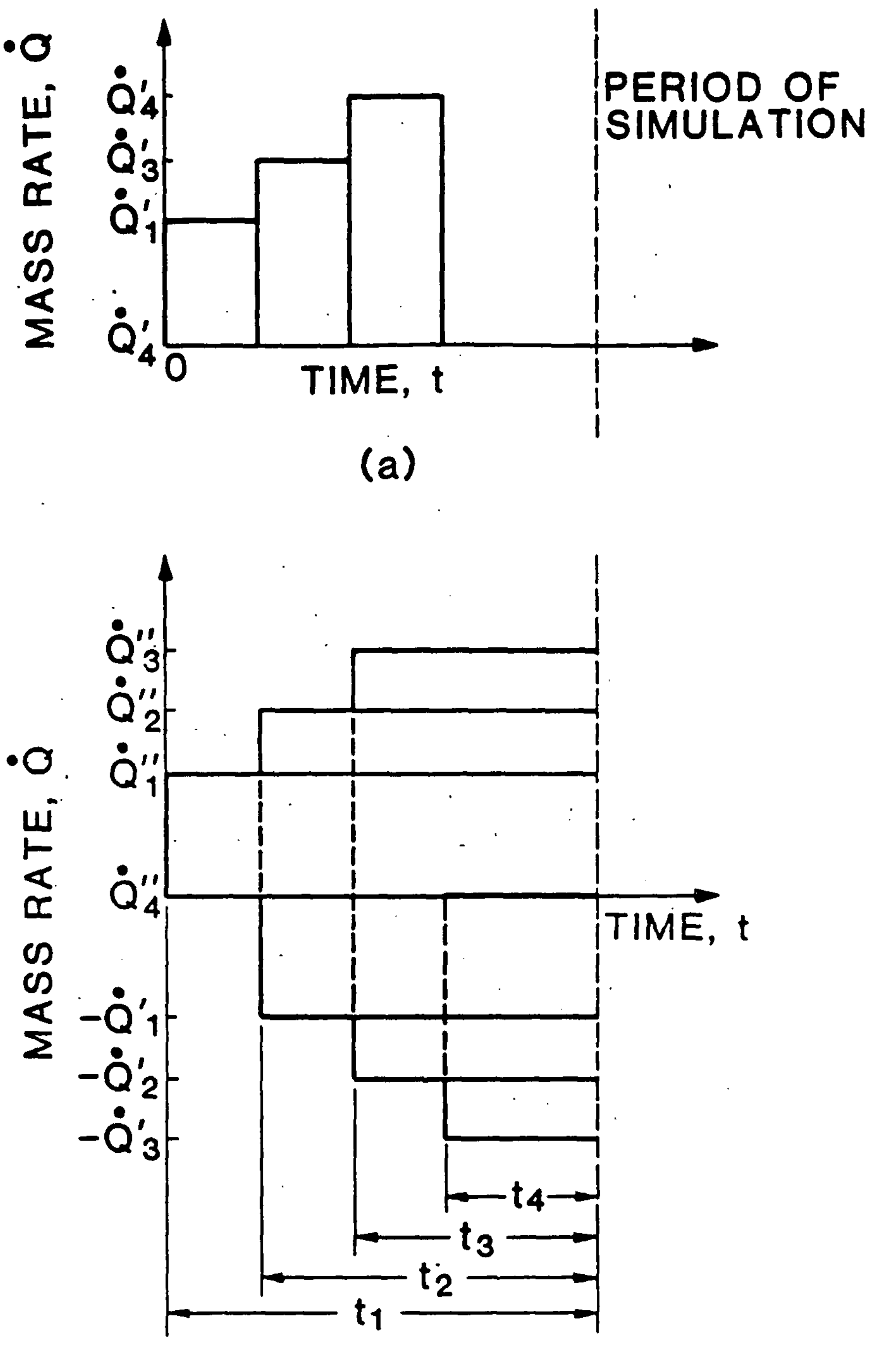


Figure 3. Decomposition of a variable source rate using superposition in time.

where t_{k-1} is the time corresponding to the end of mass rate Q_{k-1} or the beginning of rate Q_k with $Q_0 = 0$ and $t_0 = 0$.

A continuous non-uniform rate schedule may be approximated as closely as desired by increasing the number of discrete rates in the source rate schedule. In theory an infinite number of discrete rates would be required. An understanding of the physical problem and the assumptions incorporated in the mathematical model are the best guidelines for decomposing a continuous non-uniform source of contamination.

The influence of geohydrologic boundaries on the movement of a tracer is similar to the influence of these boundaries on the drawdown response of an aquifer to pumping. The applications of image well theory described by Walton (1962, 1970) can be extended to the horizontally-averaged solution to the solute transport problem considered in this report. The following discussion parallels Walton's examples of the use of image wells to account for barrier boundaries.

Consider the contaminant plume which would exist if the aquifer were of infinite depth as shown in Figure 4a. If the contaminant plume was to intersect an impermeable base of the aquifer as shown in Figure 4b, the vertical concentration gradient must change since there can be no transport of mass across the boundary as a result of dispersion. In mathematical terms

$$D_z \frac{\partial C}{\partial z} = 0$$

at z = B. Now, if an imaginary, or image, source were placed across the boundary at a distance equal to the depth of the aquifer, as shown in Figure 4c, this source would create a concentration gradient from the boundary to the image water table equal to the concentration gradient from the boundary

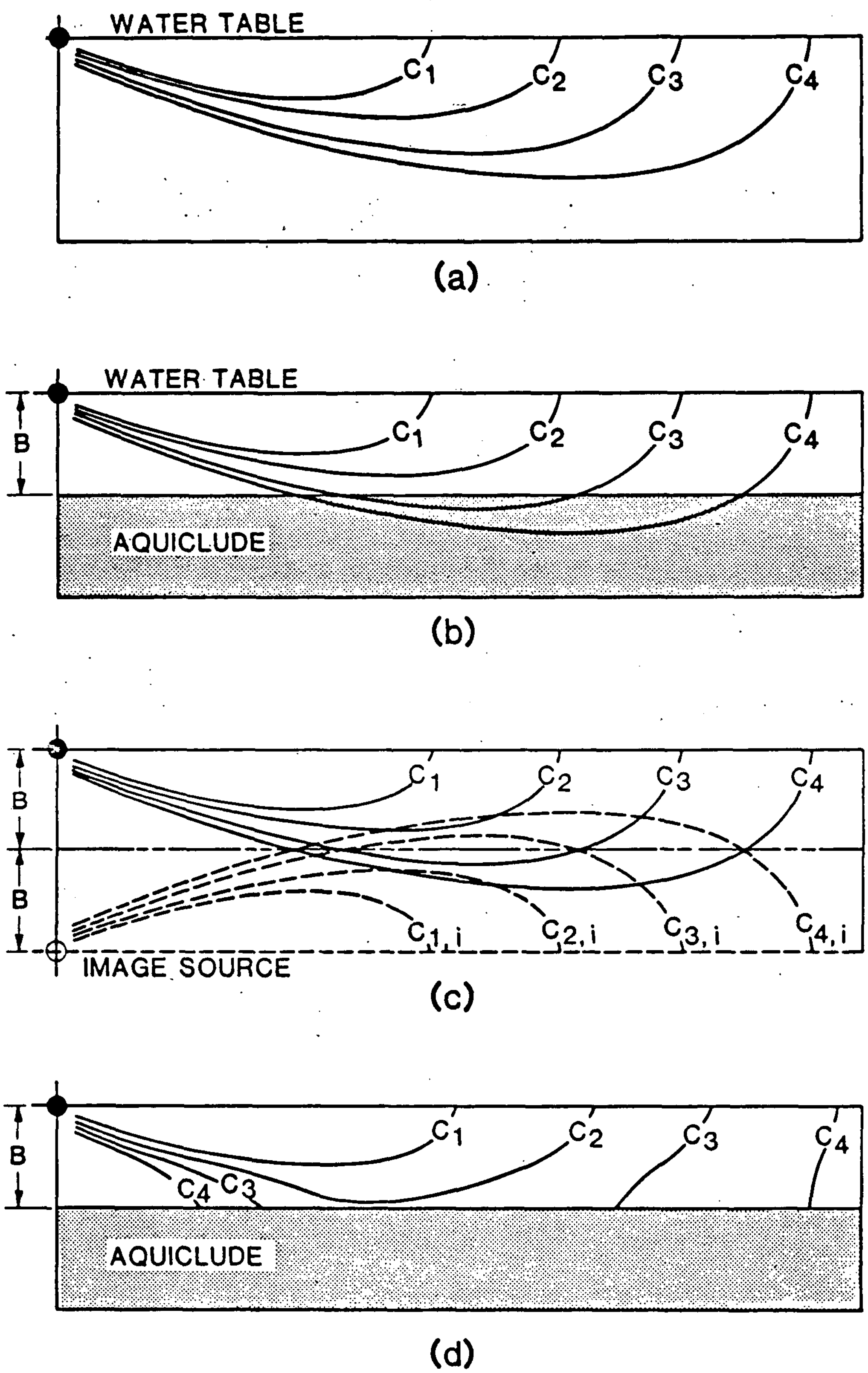
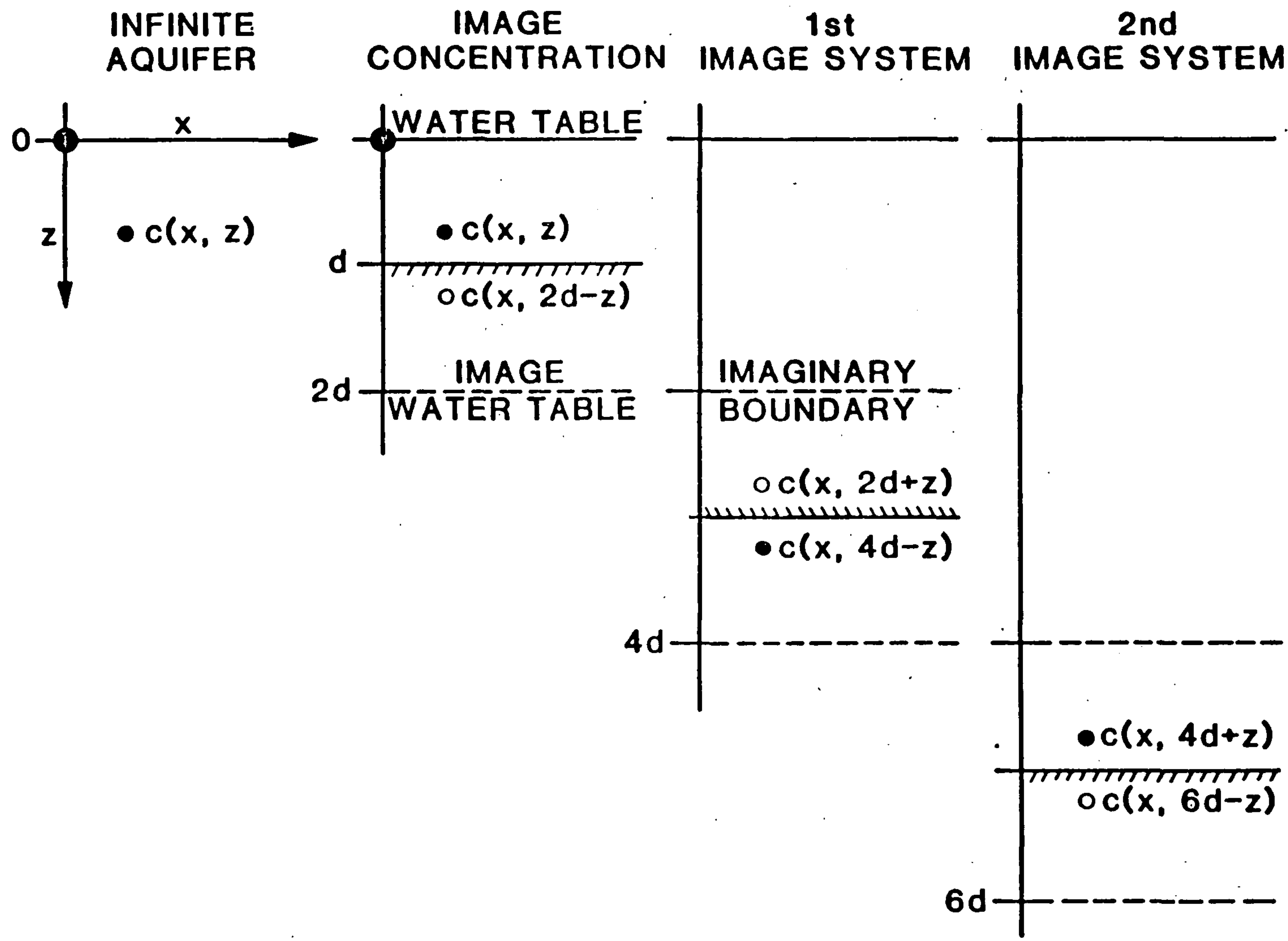


Figure 4. Use of image sources to account for aquifers of finite depth.

to the real water table. A "concentration divide" would be established at boundary, and the no-transport boundary condition ($\partial C/\partial z = 0$) would be satisfied.

The imaginary system of a contaminant source and its image in an aquifer of infinite depth satisfies the boundary conditions dictated by the finite depth system. The resultant concentration distribution is the sum of concentrations in both the real and image systems as shown in Figure 4d.

In theory an infinite number of image systems may be required. For example, if the plume in the infinite system intersects the water table in the image system, a second no-transport boundary is encountered as shown in Figure 5. This boundary can be handled by introducing another image system across the imaginary boundary and equidistant from the first image system. This process of adding image systems could be repeated indefinitely. In practice only a few image systems are required. The computer program automatically introduces an appropriate number of image systems.



$$c'(x, z) = c(x, z) + \sum_{n=1}^{\infty}\left[c(x, 2nd-z) + c(x, 2nd+z)\right]$$

Figure 5. Superposition in space to account for barrier boundaries.

SECTION II

COMPUTER PROGRAM

The computer program evaluates the analytical solutions of the differential equation describing concentration distributions in two-dimensional plumes with uniform ground-water flow. The program has been designed for interactive use and requires input data under two modes of operation - "Basic Input Data" and "Edit."

Basic Input Data

Basic input data are required to initiate a new problem using the PLUME2D program. The user is prompted for the required data through a series of input commands described below. Numeric data may be entered through the keyboard with or without decimal points and multiple data entries should be separated by comma(s). The first basic input command is:

```
ENTER TITLE
?
```

Any valid keyboard characters can be used. The first 60 characters will be retained for further problem identification.

The second input command is used to select the vertically-averaged solution or the horizontally-averaged solution. The command is:

```
ENTER COORDINATE SYSTEM
XY FOR VERTICALLY-AVERAGED SOLUTION
XZ FOR HORIZONTALLY-AVERAGED SOLUTION
?
```

Either of the indicated responses is valid.

The next three input commands define the units for all variables used in the calculations. Any consistent set of units may be used.

```
ENTER UNITS FOR LENGTH (2 CHARACTERS)
?
```

Any valid keyboard characters can be used. The first two characters will be retained for identifying the units of the length dimensions which may be required for other input data or output listings.

ENTER UNITS FOR TIME (2 CHARACTERS)
?

Any valid keyboard characters can be used. The first two characters will be retained for identifying the units of the time dimensions which may be required for other input data or output listings.

ENTER UNITS FOR CONCENTRATION (6 CHARACTERS)
?

The first six characters of any valid keyboard entries will be retained for identifying the concentration units for data input and output.

The remaining input commands are used to initialize all variables for a given problem. They include both aquifer and contaminant parameters. Input data errors which may interrupt the computational sequence are detected by the program and a command is issued to reenter the data for the appropriate variable.

ENTER SATURATED THICKNESS, (0 FOR INFINITE THICKNESS), L
?

If horizontally-averaged solution was selected (x-z coordinate system) this request is issued. The saturated thickness must be entered in the units requested with dimensions of L. If a zero or negative value is entered, the calculations will be carried out assuming an aquifer of infinite depth. The program automatically includes up to 20 image wells for aquifers of finite depth.

ENTER AQUIFER POROSITY
?

Enter the volume void fraction.

ENTER SEEPAGE VELOCITY, L/t
?

The seepage, or interstitial, velocity must be entered with dimensions of L/t in the units requested. Numerical values must be greater than zero.

ENTER RETARDATION COEFFICIENT
?

The retardation coefficient includes the effects of absorption of the tracer on the solid matrix (see Section I for discussion). The numerical value must be greater than 1.0, or equal to 1.0 if absorption is neglected.

ENTER X DISPERSION COEFFICIENT, SQ L/t
?

Dispersion coefficients have dimensions of L^2/t and must be entered in the units requested. Numerical values must be greater than zero.

If the X-Y coordinate system has been selected, the next command is:

ENTER Y DISPERSION COEFFICIENT, SQ L/t
?

If, instead, the X-Z coordinate system has been selected, a command for the Z dispersion coefficient will be issued.

ENTER Z DISPERSION COEFFICIENT, SQ L/t
?

The subsequent command will be:

ENTER DECAY CONSTANT, 1/t
?

The first order decay constant has dimensions of 1/t and must be entered in the units requested. The decay constant must be greater than, or equal to, zero.

SELECT TRANSIENT OR STEADY-STATE SOLUTION
TR FOR TRANSIENT SOLUTION
SS FOR STEADY-STATE SOLUTION
?

Selection of the transient solution also allows the approximation of a nonuniform rate schedule by a series of uniform rates (see Section I for discussion). Approximation is accomplished through superposition of a series of uniform rates. If steady-state solution is chosen, the steady state concentration will be evaluated.

ENTER THE NUMBER OF SOURCES (MAXIMUM OF N)
?

The number of sources of contaminant should be entered. The value entered must be greater than zero.

MASS RATES HAVE UNITS OF (M/L^3) (L^3/t)
TIME HAS UNITS OF t

This statement reminds the user of the units that will be used for mass rates and for time. All mass-rate and time values entered must be in these units.

The next series of commands will be repeated for each source.

ENTER X and Z COORDINATES OF SOURCE I (L)
?

The input units for the coordinates must be in the units requested. The Z-coordinate must be greater than or equal to zero. If, instead, the X-Y coordinate system has been selected, the following command is issued:

ENTER X AND Y COORDINATES OF SOURCE I (L)
?

If the transient solution was chosen the following two commands will be issued.

ENTER THE NUMBER OF RATES FOR SOURCE I (MAXIMUM OF N)
?

The number of uniform rates used to approximate a nonuniform rate schedule for this source is entered. The value must be greater than zero.

SOURCE I, RATE J STARTS AT TIME t
ENTER MASS RATE AND ENDING TIME
?

The source mass rate is entered in units of concentration times the volumetric rate. Note the actual source concentration and rate are not required, but the units must be consistent. The time units must also be consistent.

If the steady-state solution has been selected, the following command will be entered instead of the two previously listed commands.

ENTER STEADY-STATE MASS RATE I
?

The next two basic input commands are used to define the matrix of observation points, or coordinates at which concentration will be evaluated.

ENTER XFIRST, XLAST, DELTAX (L)
?,?,?

The input units for the coordinates must also be in the units requested. XFIRST and XLAST can be positive or negative values. A zero entry for DELTAX will result in a single X-coordinate observation. Results of calculations for multiple X-coordinates will be listed from XFIRST to XLAST.

ENTER YFIRST, YLAST, DELTAY (L)
?,?,?

Any of the numerical values used to define the Y-coordinates of observation points may be positive or negative. If the X-Z coordinate system has been selected, a command to enter the Z-coordinates, rather than the Y-coordinates, will be issued.

ENTER ZFIRST, ZLAST, DELTAZ (L)
?,?,?

ENTER TFIRST, TLAST, DELTAT (t)
?,?,?

The beginning value and ending value of the time interval of contaminant transport being modeled is entered. Both TFIRST and TLAST must be positive values in the units requested. A zero entry for DELTAT will result in model output at a single value of time.

Edit Commands

Once the basic input data have been entered, the problem as currently defined is listed and the program enters the "edit" mode. The edit commands are listed in Table 1 and are also listed the first time the program enters the edit mode. The request for information is

ENTER NEXT COMMAND?

One of the reponses from Table 2 should be given. If the response is incorrect or improperly formulated the statement

ERROR IN LAST COMMAND -- REENTER?

is issued. Error messages for invalid numerical data will be issued as described under the Basic Input Commands. The request for information will be repeated until one of the responses MU, LI, RN, NP, or DN is entered.

MU will list the table of edit commands.

LI will list the problem as currently defined.

RN will initiate the calculation of concentrations and print the results.

NP will request a complete new problem using the "Basic Input Data" dialog.

DN will terminate the program.

A listing of the dialog and the results for the example problem discussed in Section III are included in Appendix A.

Although many tests for valid input data and properly formulated edit commands have been embedded in the program, the user is encouraged to correct "keyboard errors" before the data are transmitted. These precautions will serve to minimize the frustration of program termination as a result of fatal errors during execution of the numerical computations.

Table 1

EDIT COMMANDS

Command	Variable changed/Execution
ST	Saturated Thickness
PO	Porosity
VX	New Seepage Velocity
RD	Retardation Coefficient
DE	Decay Constant
DX	X-Dispersion Coefficient
DY	Y-Dispersion Coefficient
DZ	Z-Dispersion Coefficient
RT	Source Rate Schedule
OB	Observation Points
XC	X-Coordinates
YC	Y-Coordinates
ZC	Z-Coordinates
TC	Observation Times
CS	Change Solution/Sources
MU	Menu of Edit Commands
LI	List input data
RN	Run
NP	New Problem
DN	Done

SECTION III

APPLICATIONS

The case history of ground-water contamination with hexavalent chromium in South Farmingdale, Nassau County, New York, (Perlmutter and Lieber, 1970), has been used as an example of the application of the two-dimensional plume model. The contaminant plume has been modeled numerically by Pinder (1973) and analytically by Wilson and Miller (1978). Details of the hydrologic system are described in the above references. A brief summary of the problem is presented in the following paragraphs.

The aquifer is assumed to have a saturated thickness of 33.52 m with a porosity of 0.35. Perlmutter and Lieber (1970) estimated the average seepage velocity to be approximately 0.366 m/dy. Using Pinder's (1973) values of dispersivity, α_x = 21.3 m and α_y = 4.27 m, and x and y dispersion coefficients are

$$D_x = (21.3 \text{ m}) (0.366 \text{ m/dy}) = 7.70 \text{ m}^2/\text{dy}$$

and

$$D_y = (4.27 \text{ m}) (0.366 \text{ m/dy}) = 1.56 \text{ m}^2/\text{dy}$$

The source of contamination consisted of three metal-plating-waste disposal ponds as shown in Figure 6. The mass rate of chromium entering the aquifer has been estimated at 23.6 kg/dy during the nine year period from 1941 through 1949 (Perlmutter and Lieber, 1970). Chromium is believed to be a conservative contaminant, thus absorption and degradation can be neglected. The vertically-averaged model parameters are summarized as follows:

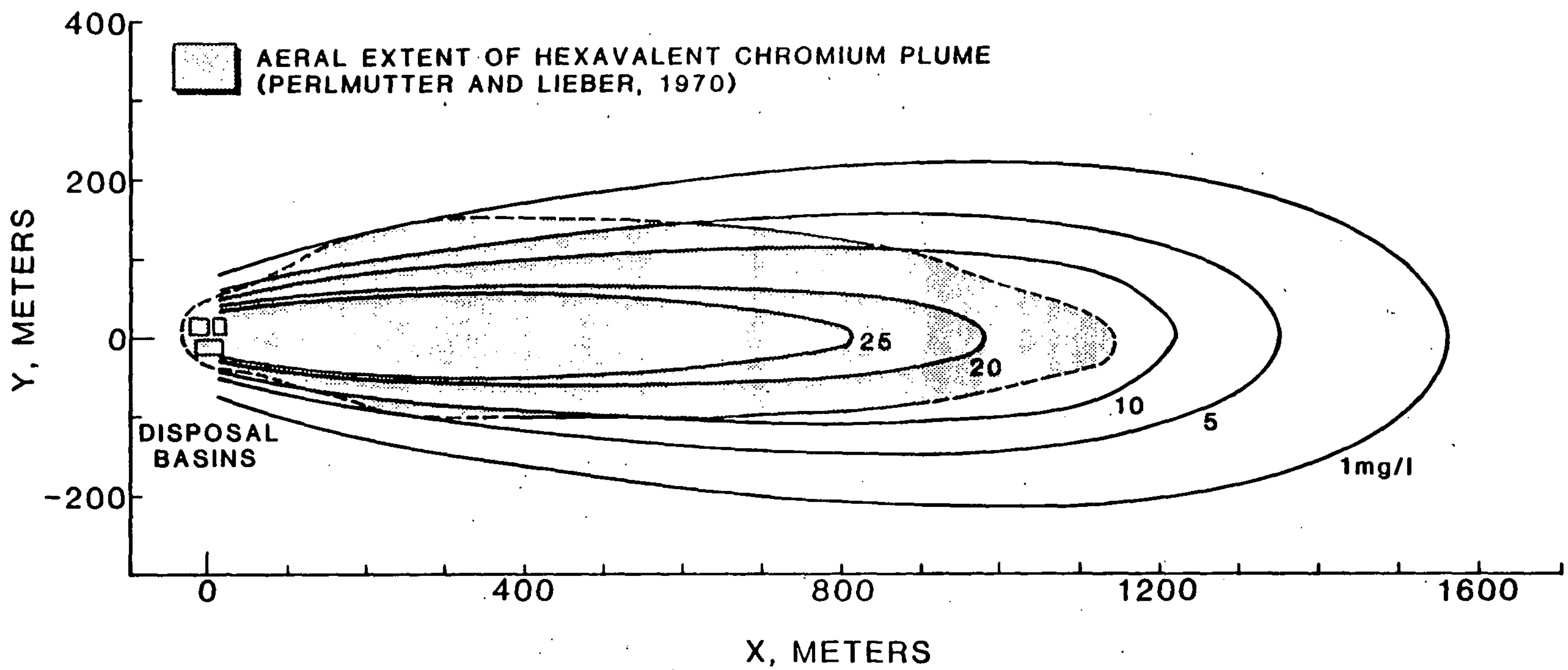


Figure 6. Results of hexavalent chromium plume simulation at 3280 days.

Aquifer porosity	0.35
Seepage velocity	0.366 m/dy
Retardation coefficient	1.0
x-Dispersion coefficient	7.79 m^2/dy
y-Dispersion coefficient	1.56 m^2/dy
Decay constant	0.0 1/dy

The contaminant source rate is assumed to be constant, and only one rate period is required. The mass rate can be converted to units of concentration times volume rate per unit depth as

$$\frac{23.6\ \text{kg}}{\text{dy}}\ \frac{10^6\ \text{mg}}{\text{kg}}\ \frac{\text{m}^3}{10^3\ \text{l}}\ \frac{1}{33.52\ \text{m}} = 704\ (\text{mg/l})\ (\text{m}^3/\text{dy})/\text{m}$$

for approximately nine years or 3280 days.

The numerical results for the vertically-averaged solution are summarized in Figure 6. The shape and general extent of the predicted plume are in fair agreement with the observed extent of contamination considering the availability of field data and the assumptions which have been made in characterizing the problem as two-dimensional uniform flow with a continuous line source.

Superposition in time will be illustrated using data for the aquifer contaminated with chromium described above. Rather than a continuous source of contamination from the disposal ponds, as "accidental spill" of high strength waste will be simulated. The contaminant source will be assumed to be an "instantaneous line source" of strength 704 (mg/l) (m^3/m). The source rate schedule for the vertically-averaged model is constructed as follows:

Rate 1: 704 (mg/1) $(m^3/dy)/m$ from 0 to 1 day

Rate 2: 0 (mg/1) $(m^3/dy)/m$ from 1 to 365 days

Other model parameters are identical to those used in the previous example. The results of the simulation are summarized in Figure 7, which shows the center of mass of the plume moving down-gradient at the seepage velocity and spreading longitudinally and transversely by diffusion.

The results of the simulation using superposition in time were compared with the concentrations calculated using

$$C = \frac{C_0 Q'}{4\pi\theta t (D_x D_y)^{0.5}} \text{EXP} \left(- \frac{(x - V^* t)^2}{4D_x t} - \frac{y^2}{4D_y t} - \lambda t\right) \quad (54)$$

which is the solution of Equation 14 for an instantaneous line source of strength C_0Q' (m/L^3) (L^3/L). The values of concentration and errors in approximating the instantaneous source through superposition in time are presented in Table 2. Note that the finite duration of the source results is slightly higher concentrations up-gradient from the center of mass than concentrations down-gradient. For an instantaneous source the concentration distribution should be symmetrical about a y-z plane through the center of mass located as x = Vt. A better approximation can be obtained by injecting the same total mass of contaminant over a shorter period of time; but for purposes of illustrating superposition in time, the errors in the example problem are not significant.

The example problems presented above are intended to illustrate the application of the two-dimensional plume models developed in this report. These models are tools which can aid in the analysis of ground-water contamination problems. The user must select the best tool for the problem at

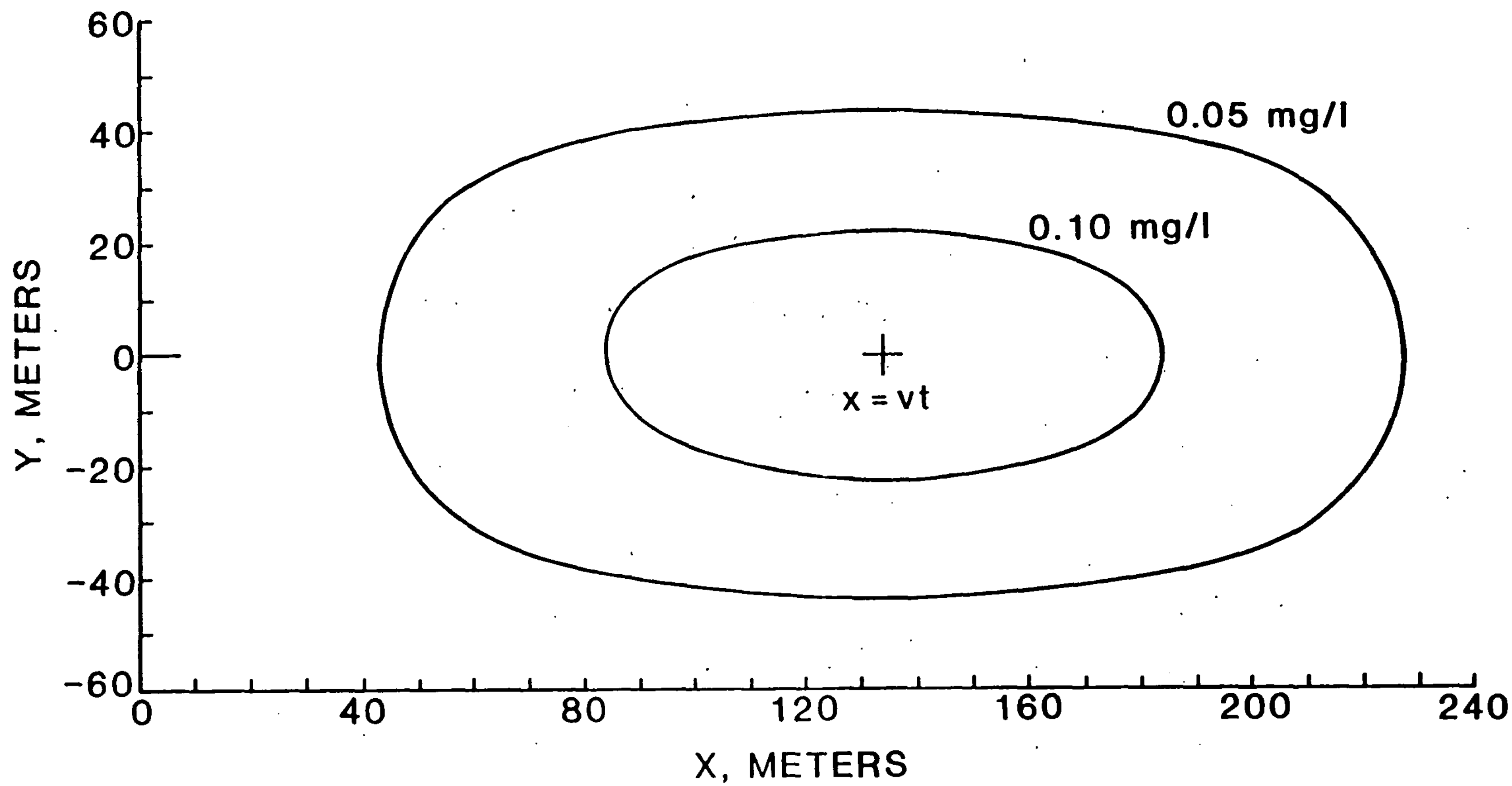


Figure 7. Results of hexavalent chromium spill simulation at 365 days.

hand, based on a sound understanding of the principles of ground-water hydrology, the physical problem, and the limitations of the mathematical model(s).

Perhaps the most difficult step in using any mathematical model is defining the problem to be solved. In addition to developing the physical boundaries of the problem domain, rock and fluid properties must also be quantified. Typical values of aquifer properties are listed in Table 3, but the user must accept the responsibility for developing the required model input data for the specific problem to be solved.

Table 2

COMPARISON OF CONCENTRATIONS CALCULATED USING SUPERPOSITION IN TIME AND AN ANALYTICAL SOLUTION FOR AN INSTANTANEOUS LINE SOURCE

MODEL PARAMETERS

Aquifer Porosity	0.35
Seepage Velocity	0.366 m/dy
Retardation Coefficient	1.0
x-Dispersion Coefficient	7.79 m^2/dy
y-Dispersion Coefficient	1.56 m^2/dy
First-Order Decay Constant	0.0 1/dy
Source Strength	704.0 (mg/1) (m^3/m)

Concentration at 365 days, mg/1

Superposition
(Equation 54)
% Error

y (meters)	x(meters) 73.59	103.59	133.59 (= Vt)	163.59	193.59
20.0	0.0919 (.0917) .22	0.1165 (.1162) .26	0.1259 (.1258) .08	0.1163 (.1162) .01	0.0916 (.0917) -.11
10.0	0.0878 (.0877) .11	0.1115 (.1112) .27	0.1206 (.1204) .17	0.1113 (.1112) .09	0.0876 (.0877) -.11
0.0	0.0771 (.0769) .26	0.0977 (.0975) .21	0.1057 (.1055) .19	0.0975 (.0975) .0	0.0768 (.0769) -.13

Table 3

Typical Values of Aquifer Properties

(after Yeh, 1981)

Parameter	Material: Clay	Silt	Sand
Bulk density, lb/ft^3	87.36 - 137.2	80.50 - 112.3	73.63 - 98.59
Effective porosity	0.03 - 0.05	0.05 - 0.10	0.10 - 0.30
Hydraulic Conductivity, $gal/day/ft^2$	0.01 - 0.1	1 - 10	100 - 100,000
Dispersivity, ft			
Longitudinal	0.1 - 1.0	1 - 10	10 - 100
Transverse	0.01 - 0.1	0.1 - 1.0	1.0 - 10
Vertical 0.01 - 0.1	0.1 - 1.0	1 - 10	

REFERENCES

Abramowitz, M. and I. A. Stegan, 1966, Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, National Bureau of Standards Applied Mathematics Series 55, U. S. Department of Commerce, 1046 pp.

Hantush, M. S., 1956, "Analysis of Data from Pumping Tests in Leaky Aquifers," Transitions, American Geophysical Union, Vol. 37, No. 6, pp. 702-714.

Hantush, M. S., 1964, "Hydraulics of Wells," Advances in Hydroscience, Vol. 1, pp. 281-432.

Hantush, M. S. and C. E. Jacob, 1955, "Non-Steady Radial Flow in an Infinite Leaky Aquifer," Transitions, American Geophysical Union, Vol. 26., No. 1, pp. 95-100.

Hunt, B., 1978, "Dispersive Sources in Uniform Ground-Water Flow," Journal of The Hydraulics Division, ASCE, Vol. 104, No. HY1, pp. 75-85.

Perlmutter, N. M. and M. Lieber, 1970, "Dispersal of Plating Wastes and Sewage Contaminants in Ground Water and Surface Water, South Farmingdale-Massapequa Area, Nassau County, New York," U. S. Geological Survey Water-Supply Paper 1879-G, pp. G1-G67.

Pinder, G. F., 1973, "A Galerkin Finite Element Simulation of Ground-water Contamination on Long Island," Water Resources Research, Vol. 9, No. 6, pp. 1657-1669.

Walton, W. C., 1962, "Selected Analytical Methods for Well and Aquifer Evaluation," Bulletin 49, Illinois State Water Survey, Urbana, Illinois, 81 pp.

Walton, W. C., 1970, Groundwater Resource Evaluation, Mc-Graw-Hill, New York, New York, 664 pp.

Wilson, J. L. and P. J. Miller, 1978, "Two-Dimensional Plume in Uniform Ground-Water Flow," Journal of the Hydraulics Division, ASCE, Vol. 104, No. HY4, pp. 503-514.

Wilson, J. L. and P. J. Miller, 1979, "Two-Dimensional Plume in Uniform Ground-Water Flow, Discussion," Journal of the Hydraulics Division, ASCE, Vol. 105, No. HY12, pp. 1567-1570.

Yeh, G. T., 1981, "AT123D: Analytical Transient One-, Two-, and Three-Dimensional Simulations of Waste Transport in the Aquifer System," Publication No. 1439, Environmental Sciences Division, Oak Ridge National Laboratory, Oak Ridge, Tennessee, 83 pp.

APPENDIX A

Example Problems

The two example problems presented in the following pages are discussed in Section III of this report. The first demonstrates the application of PLUME2D to a continuous source of contamination. The second example approximates an instantaneous source using the principle of superposition in time as discussed in Section I.

```
ENTER TITLE
?HEXAVALENT CHROMIUM PLUME

ENTER COORDINATE SYSTEM
   XY FOR VERTICALLY-AVERAGED SOLUTION
   XZ FOR HORIZONTALLY-AVERAGED SOLUTION
?XY

ENTER UNITS FOR LENGTH (2 CHARACTERS)
? M

ENTER UNITS FOR TIME (2 CHARACTERS)
?DY

ENTER UNITS FOR CONCENTRATION (6 CHARACTERS)
?MG/L

ENTER AQUIFER POROSITY
?Ø.35

ENTER SEEPAGE VELOCITY,  M/DY
?Ø.366

ENTER RETARDATION COEFFICIENT
?1.Ø

ENTER X DISPERSION COEFFICIENT, SQ  M/DY
?7.79

ENTER Y DISPERSION COEFFICIENT, SQ  M/DY
?1.56

ENTER DECAY CONSTANT, 1/DY
?Ø.Ø

SELECT TRANSIENT OR STEADY-STATE SOLUTION
   TR FOR TRANSIENT SOLUTION
   SS FOR STEADY-STATE SOLUTION
?TR
```

```
ENTER THE NUMBER OF SOURCES (MAXIMUM OF 10 )
?1

MASS RATES HAVE UNITS OF (MG/L  ) (CU  M/DY)
TIME HAS UNITS OF DY

ENTER X AND Y COORDINATES OF SOURCE 1 ( M)
?,?0.,0.

ENTER THE NUMBER RATES FOR SOURCE 1 (MAXIMUM OF 10)
?1

SOURCE  1, RATE  1 STARTS AT      0.0 DY
ENTER MASS RATE AND ENDING TIME
?,?704.,3280.

ENTER XFIRST, XLAST, DELTAX  ( M)
?,?,?200.,1200.,200.

ENTER YFIRST, YLAST, DELTAY  ( M)
?,?,?200.,-200.,50.

ENTER TFIRST, TLAST, DELTAT  (DY)
?,?,?3280.,0.,0.
```

PLUME2D
VERSION 2.01
PAGE 1

HEXAVALENT CHROMIUM PLUME

SEEPAGE VELOCITY, (M/DY)	.3660
X DISPERSION COEFFICIENT (M**2/DY)	7.7900
Y DISPERSION COEFFICIENT (M**2/DY)	1.5600
POROSITY	.3500
RETARDATION COEFFICIENT	1.0000
FIRST ORDER DECAY CONSTANT (1/DY)	0.0000

SOURCE/RATE SCHEDULE (MG/L)(CU M/DY)

	SOURCE		RATE	MASS	TIME (DY)	
NO	X (M)	Y (M)	NO	RATE	START	END
1	0.00	0:00				
			1	704.00	0.00	3280.00

OBSERVATION POINTS (M)

XFIRST =	200.00	XLAST =	1200.00	DELX =	200.0000
YFIRST =	200.00	YLAST =	-200.00	DELY =	50.0000

OBSERVATION TIMES (DY)

TFIRST =	3280.00	TLAST =	3280.00	DELT =	0.0000

MENU OF EDIT COMMANDS

RETARDATION COEFFICIENT	RD	OBSERVATION POINTS	OB
POROSITY	PO	X COORDINATES	XC
SEEPAGE VELOCITY	VX	Y COORDINATES	YC
X DISPERSION COEFFICIENT	DX	MENU OF COMMANDS	MU
Y DISPERSION COEFFICIENT	DY	LIST INPUT DATA	LI
DECAY CONSTANT	DE	RUN CALCULATIONS	RN
SOURCE RATE SCHEDULE	RT	DONE	DN
NEW PROBLEM	NP	SATURATED THICKNESS	ST
CHANGE SOLUTION/SOURCES	CS	OBSERVATION TIMES	TC

ENTER NEXT COMMAND
?RN

PLUME2D
VERSION 2.01
PAGE 2

HEXAVALENT CHROMIUM PLUME

CONCENTRATION DISTRIBUTION AT 3280.00 DY (MG/L)

Y(M) \ X(M)	200.00	400.00	600.00	800.00	1000.00	1200.00
200.00	.0372	.2773	.8210	1.4371	1.6352	1.1380
150.00	.4289	1.8560	3.6177	4.8444	4.7217	3.0238
100.00	4.0806	8.8387	11.3609	11.9818	10.2348	6.1201
50.00	24.5165	25.3968	23.5539	20.9946	16.4014	9.3721
0.00	51.8245	37.0664	30.2812	25.3930	19.2190	10.8087
-50.00	24.5165	25.3968	23.5539	20.9946	16.4014	9.3721
-100.00	4.0806	8.8387	11.3609	11.9818	10.2348	6.1201
-150.00	.4289	1.8560	3.6177	4.8444	4.7217	3.0233
-200.00	.0372	.2773	.8210	1.4371	1.6352	1.1380

ENTER NEXT COMMAND
?DN

STOP

```
ENTER TITLE
?ACCIDENTAL HEXAVALENT CHROMIUM SPILL

ENTER COORDINATE SYSTEM
   XY FOR VERTICALLY-AVERAGED SOLUTION
   XZ FOR HORIZONTALLY-AVERAGED SOLUTION
?XY

ENTER UNITS FOR LENGTH (2 CHARACTERS)
? M

ENTER UNITS FOR TIME (2 CHARACTERS)
?DY

ENTER UNITS FOR CONCENTRATION (6 CHARACTERS)
?MG/L

ENTER AQUIFER POROSITY
?Ø.35

ENTER SEEPAGE VELOCITY,  M/DY
?Ø.366

ENTER RETARDATION COEFFICIENT
?1.Ø

ENTER X DISPERSION COEFFICIENT, SQ  M/DY
?7.79

ENTER Y DISPERSION COEFFICIENT, SQ  M/DY
?1.56

ENTER DECAY CONSTANT, 1/DY
?Ø.Ø

SELECT TRANSIENT OR STEADY-STATE SOLUTION
   TR FOR TRANSIENT SOLUTION
   SS FOR STEADY-STATE SOLUTION
?TR
```

```
ENTER THE NUMBER OF SOURCES (MAXIMUM OF 1Ø )
?1

MASS RATES HAVE UNITS OF (MG/L  ) (CU  M/DY)
TIME HAS UNITS OF DY

ENTER X AND Y COORDINATES OF SOURCE 1 ( M)
?,?Ø.,Ø.

ENTER THE NUMBER RATES FOR SOURCE 1 (MAXIMUM ØF 1Ø)
?2

SOURCE  1, RATE  1 STARTS AT     Ø.Ø DY
ENTER MASS RATE AND ENDING TIME
?,?7Ø4.,1.

SOURCE  1, RATE  2 STARTS AT     1.Ø DY
ENTER MASS RATE AND ENDING TIME
?,?Ø.,365.

ENTER XFIRST, XLAST, DELTAX  ( M)
?,?,?73.59,193.59,3Ø.

ENTER YFIRST, YLAST, DELTAY  ( M)
?,?,?2Ø.,Ø.,1Ø.

ENTER TFIRST, TLAST, DELTAT  (DY)
?,?,?365.,Ø.,Ø.
```

PLUME2D
VERSION 2.01
PAGE 1

ACCIDENTAL HEXAVALENT CHROMIUM SPILL

SEEPAGE VELOCITY, (M/DY)	.3660
X DISPERSION COEFFICIENT (M**2/DY)	7.7900
Y DISPERSION COEFFICIENT (M**2/DY)	1.5600
POROSITY	.3500
RETARDATION COEFFICIENT	1.0000
FIRST ORDER DECAY CONSTANT (1/DY)	0.0000

SOURCE/RATE SCHEDULE (MG/L)(CU M/DY)

	SOURCE		RATE	MASS	TIME (DY)	
NO	X (M)	Y (M)	NO	RATE	START	END
1	0.00	0.00				
			1	704.00	0.00	1.00
			2	0.00	1.00	365.00

OBSERVATION POINTS (M)

XFIRST =	73.59	XLAST =	193.59	DELX =	30.0000
YFIRST =	20.00	YLAST =	0.00	DELY =	10.0000

OBSERVATION TIMES (DY)

TFIRST =	365.00	TLAST =	365.00	DELT =	0.0000

MENU OF EDIT COMMANDS

RETARDATION COEFFICIENT	RD	OBSERVATION POINTS	OB
POROSITY	PO	X COORDINATES	XC
SEEPAGE VELOCITY	VX	Y COORDINATES	YC
X DISPERSION COEFFICIENT	DX	MENU OF COMMANDS	MU
Y DISPERSION COEFFICIENT	DY	LIST INPUT DATA	LI
DECAY CONSTANT	DE	RUN CALCULATIONS	RN
SOURCE RATE SCHEDULE	RT	DONE	DN
NEW PROBLEM	NP	SATURATED THICKNESS	ST
CHANGE SOLUTION/SOURCES	CS	OBSERVATION TIMES	TC

ENTER NEXT COMMAND
?RN

PLUME2D
VERSION 2.Ø1
PAGE 2

ACCIDENTAL HEXAVALENT CHROMIUM SPILL

CONCENTRATION DISTRIBUTION AT 365.ØØ DY (MG/L)

```
*
  * X( M)
    *        73.59     1Ø3.59     133.59     163.59     193.59
Y( M) *
        *
   2Ø.ØØ     .Ø771      .Ø977      .1Ø56      .Ø975      .Ø768
   1Ø.ØØ     .Ø879      .1115      .12Ø4      .1113      .Ø876
    Ø.ØØ     .Ø919      .1165      .126Ø      .1163      .Ø916
```

ENTER NEXT COMMAND
?XC

ENTER XFIRST, XLAST, DELTAX (M)
?,?,?1Ø3.59,163.59,15.

ENTER NEXT COMMAND
?RN

```
PLUME2D
VERSION 2.01
PAGE   3

ACCIDENTAL HEXAVALENT CHROMIUM SPILL


          CONCENTRATION DISTRIBUTION AT     365.00 DY (MG/L  )


*
  * X( M)
    *        103.59     118.59     133.59     148.59     163.59
Y( M) *
       *
    20.00     .0977      .1036      .1056      .1035      .0975
    10.00     .1115      .1183      .1204      .1181      .1113
     0.00     .1165      .1236      .1260      .1234      .1163

ENTER NEXT COMMAND
?DN

STOP
```

APPENDIX B

Description of Program PLUME2D

Program PLUME2D has been written in an unextended Fortran computer code in an effort to make the program transportable between computer systems. The computer code consists of a main program and several function subroutines which are required to evaluate the Hantush well function. The program has been documented "internally" through the liberal use of comment statements.

The main program has been divided into three sections. A listing of the computer code is presented in Appendix D. Section I provides for the "Basic Input Data" as described in Section II of this report. The numerical evaluation of concentration at specified grid coordinates is accomplished in Section II of the main program which calls subroutine SOL2D, the code for the analytical solution of the governing differential equations. Section III provides for problem redefinition and control of execution under the "Edit" mode discussed in the body of this report.

Ten function subroutines are used to evaluate the Hantush well function using the numerical methods described in Appendix C. Listings of the computer codes are presented in Appendix E. FUNCTION W(U,B) evaluates the Hantush well function for $B \leqslant 20$. For $B > 20$, the term $EXP(Pe_x/2)$ W(U,B) in Equation 45 is evaluated using FUNCTION WELPRD(U,B,PEX). This procedure is used to avoid taking the direct product of very large numbers, $EXP(Pe_x/2)$, and very small numbers W(U,B), for large values of B.

FUNCTION GAUSS is a 24-point Gauss-Legendre quadrature numerical integration scheme which is used to evaluate the Hantush well function using either Equation C-6 or Equation C-7. FUNCTION FUNCTN evaluates the integrand of Equations C-6 and C-7.

The six remaining function subroutines are used to evaluate mathematical functions using rational approximations or polynomial approximations. They are:

FUNCTION BIO(Z) FUNCTION BIOLOG(Z)	Modified Bessel function of the first kind of order zero and the natural logarithm of the function.
FUNCTION BKO(Z) FUNCTION BKOLOG(Z)	Modified Bessel function of the second kind of order zero and the natural logarithm of the function.
FUNCTION E1LOG(Z)	Natural logrithm of the exponential integral.
FUNCTION ERFC(Z)	Complimentary error function.

These six function subroutines are used to support FUNCTION W(U,B) and/or FUNCTION WELPRD (U,B,PEX). If system subroutines are available for these functions they may be substituted for the function subroutines provided with Program PLUME2D.

APPENDIX C

Numerical Evaluation of the Hantush Well Function

The Hantush well function can be defined as

$$W(U,B) = \int_u^\infty \frac{1}{\xi} \text{EXP}(- \xi - \frac{B^2}{4\xi})\, d\xi \qquad \text{(C-1)}$$

or the reciprocal relation

$$W(U,B) = 2K_0(B) - \int_{B^2/4U}^\infty \frac{1}{\xi} \text{EXP}(- \xi - \frac{B^2}{4\xi})\, d\xi \qquad \text{(C-2)}$$

where ξ is a dummy integration variable (Hantush, 1964). Using the identity

$$\int_a^\infty f(\xi)\, d\xi = \int_0^\infty f(\xi)\, d\xi - \int_0^a f(\xi)\, d\xi \qquad \text{(C-3)}$$

Equation C-1 can be rewritten as

$$W(U,B) = \int_0^\infty \frac{1}{\xi} \text{EXP}(- \xi - \frac{B^2}{4\xi})\, d\xi - \int_0^u \frac{1}{\xi} (- \xi - \frac{B^2}{4\xi})\, d\xi \qquad \text{(C-4)}$$

Now

$$\int_0^\infty \frac{1}{\xi} \text{EXP}(- \xi - \frac{B^2}{4\xi})\, d\xi = 2K_0(B) \qquad \text{(C-5)}$$

where K_0 is the modified Bessel function of the second kind of order zero. Substituting Equation C-5 into Equation C-4, the well function becomes

$$W(U,B) = 2K_0(B) - \int_0^U \frac{1}{\xi} EXP(-\xi - \frac{B^2}{4\xi})\, d\xi \qquad (C\text{-}6)$$

The reciprocal relation, Equation C-2, can also be written in terms of finite limits. Using the relationship given by Equations C-4 and C-5, the reciprocal relation can be expressed as

$$W(U,B) = \int_0^{B^2/4U} \frac{1}{\xi} EXP(-\xi - \frac{B^2}{4\xi})\, d\xi \qquad (C\text{-}7)$$

For $0 < B \leq 20$, values of $W(U,B)$ for $0 < U \leq B/2$ are obtained from Equation C-6 by first evaluating the value of the integrand using a 24-point Gauss-Legendre numerical integration scheme. For $B/2 \leq U < \infty$, the reciprocal relation, Equation C-7, is evaluated using the same numerical integration scheme.

For $0 < B \leq 0.1$, values of $W(U,B)$ are obtained from the series expansions presented by Hantush and Jacob (1955). For $U \leq 1$

$$W(U,B) = 2K_0(B) - I_0(B)E_1(\frac{B^2}{4U})$$

$$+ EXP(-\frac{B^2}{4U}) \left[0.57721566 + \ln(U) + E_1(U) + \frac{U}{4}\frac{B^2}{4}(1 - \frac{U}{9})\right] \qquad (C\text{-}8)$$

and for $U > 1$

$$W(U,B) = I_0(B)E_1(U) - EXP(-U)\frac{B^2}{4}\left[(\frac{1}{U} - \frac{1}{36U^2}) + \frac{B^4}{16}(\frac{1}{4U} - \frac{1}{4U^2})\right] \qquad (C\text{-}9)$$

where I_0 is the modified Bessel function of the first kind of order zero, E_1 is the exponential integral, and 0.57721566 is Euler's constant.

For B > 20, the third order approximation for W(U,B) presented by Wilson and Miller (1979) is used to evaluate the well function. The approximation is

$$W(U,B) = \left(\frac{\pi}{2B}\right)^{1/2} EXP(-B) \left[\left(1 - \frac{1}{8B}\right) ERFC\ (-\beta) + \frac{\beta}{4B\pi^{1/2}} EXP(-\beta^2)\right] \tag{C-10}$$

where

$$\beta = \frac{B-2U}{(4U)^{1/2}}$$

and ERFC is the complimentary error function.

Now, for large positive values of β,

$$W(U,B) \simeq 2 \left(\frac{\pi}{2B}\right)^{1/2} EXP(-B) \left(1 - \frac{1}{8B}\right) \tag{C-11}$$

and an asympotic expansion for $K_0(B)$ can be written as

$$K_0(B) = \left(\frac{\pi}{2B}\right)^{1/2} EXP(-B) \left(1 - \frac{1}{8B} + \frac{9}{2\ (8B)^2} + \ldots\right) \tag{C-12}$$

Thus for B > 20 and β > 7.5 the well function is approximated as

$$W(U,B) = 2K_0(B) \tag{C-13}$$

Note that this approximation is equivalent to the relationship

$$W(0,B) = 2K_0(B) \tag{C-14}$$

Evaluations of the Hantush well function using the methods described in the previous paragraphs have been checked for both accuracy and continuity of the function between the various approximations. The Gauss-Legendre quadrature scheme was checked using up to 48 quadrature points. A maximum of 24 quadrature points yielded results accurate to four significant figures in the mantissa over the entire range of arguments which require numerical integration. The other approximations for W(U,B) are also accurate to four significant figures in the mantissa.

APPENDIX D

Listing of Program PLUME2D

```
C     PLUME2D
C     VERSION 2.02
C     TWO-DIMENSIONAL PLUMES IN UNIFORM GROUND-WATER FLOW
C        JAN WAGNER
C        SCHOOL OF CHEMICAL ENGINEERING
C        OKLAHOMA STATE UNIVERSITY
C        STILLWATER, OK  74078
C        PHONE (405) 624-5280
C        JULY, 1981
C
C     REVISIONS:   2.00       APR 84
C                  2.01    24 NOV 84
C                  2.02     9 DEC 84
C                  2.02A    3 MAY 85
C
      DIMENSION TITLE(30),IC(20),XS(10),YS(10),D(3),LBL(2,6),
     1 NR(10),IS(4),NP(2),DEL(2),XL(2),XF(2),CON(7),COL(7)
      REAL LAMBDA
      INTEGER TITLE
      COMMON/IO/NI,NO
      COMMON/RATE/Q(10,12),T(10,12),MT
      COMMON/PHYPRO/ALPHA,BETA,DX,LAMBDA,PE,RD,V
      DATA IC/'DE','VX','RD','DX','DY','DZ','PO','OB','XC','YC','ZC',
     1'RT','NP','RN','DN','LI','MU','ST','CS','TC'/
      DATA KPRO1/'XY'/, KPRO2/'XZ'/, KHARY/'Y'/, KHARZ/'Z'/
      DATA NPAGE/1/
      DATA KSOL1,KSOL2/'TR','SS'/
      DATA IS/'R','M','A','D'/
      DATA IY/'Y'/
      DATA LBL/'  ','(C','  ','ON','  ','TI','  ','NU','  ','ED',
     1 '  ',') '/
C
C     READ DEVICE: NI       WRITE DEVICE: NO
      NI=5
      NO=6
C
C     MAXIMUM NUMBER OF PRINTED COLUMNS PER PAGE IS SET TO MAXCOL
C        DIMENSION COL(MAXCOL),CON(MAXCOL)
      MAXCOL = 7
C
C     MAXIMUM NUMBER OF PRINTED ROWS PER PAGE IS SET TO MAXROW
      MAXROW = 40
C
C     MAXIMUM NUMBER OF SOURCES IS SET TO MAXSOR
C        DIMENSION XS(MAXSOR),YS(MAXSOR),NR(MAXSOR)
      MAXSOR = 10
C
C     MAXIMUM NUMBER OF SOURCE RATES FOR SUPERPOSITION IN TIME
C     IS SET TO MAXRT
C        COMMON/RATE/ Q(MAXSOR,MAXRT+2),T(MAXSOR,MAXRT+2)
      MAXRT = 10
C
C     MAXIMUM NUMBER OF IMAGE WELLS FOR SUPERPOSITION IN SPACE
C     IS SET TO MAXIMG
      MAXIMG = 20
C
C
C     INITIALIZE PROGRAM FLOW CONTROL VARIABLES
    1 IEDIT = 1
      KNTL = 1
C
C
C **** SECTION I -- BASIC INPUT DATA
C
C     READ TITLE
      WRITE(NO,3)
    3 FORMAT(1H1,2X,'ENTER TITLE',/'    ?')
      READ(NI,5) (TITLE(I), I=1,30)
    5 FORMAT(30A2)
C
```

```
C     SELECT VERTICALLY OR HORIZONTALLLY AVERAGED SOLUTION
      KFLOW = 3
      WRITE(NO,7)
    7 FORMAT(3X,'ENTER COORDINATE SYSTEM',/,
     16X,'XY FOR VERTICALLY-AVERAGED SOLUTION',/,
     26X,'XZ FOR HORIZONTALLY-AVERAGED SOLUTION',/,'    ?')
    8 READ(NI,9) KFLG
    9 FORMAT(A2)
      IF(KFLG.EQ.KPRO1) KFLOW=1
      IF(KFLG.EQ.KPRO2) KFLOW=2
      GO TO (12,12,10), KFLOW
   10 WRITE(NO,11)
   11 FORMAT(3X,'ERROR IN PROBLEM SELECTION -- REENTER',/,'    ?')
      GO TO 8
   12 CONTINUE
      KHAR = KHARY
      IF(KFLOW.EQ.2) KHAR=KHARZ
C
C     DEFINE UNITS
      WRITE(NO,15)
   15 FORMAT(3X,'ENTER UNITS FOR LENGTH (2 CHARACTERS)',/,'    ?')
      READ(NI,25) IL
   25 FORMAT(A2)
      WRITE(NO,35)
   35 FORMAT(3X,'ENTER UNITS FOR TIME (2 CHARACTERS)',/,'    ?')
      READ(NI,25) IT
      WRITE(NO,45)
   45 FORMAT(3X,'ENTER UNITS FOR CONCENTRATION (6 CHARACTERS)',/,'    ?')
      READ(NI,26) IM1,IM2,IM3
   26 FORMAT(3A2)
C
C     ENTER DATA FOR FIRST PROBLEM
C
C     SATURATED THICKNESS
      IF(KFLOW.EQ.1)GO TO 38
   30 IMAGE = MAXIMG/2
   44 WRITE(NO,46) IL
   46 FORMAT(3X,'ENTER SATURATED THICKNESS (0 FOR INFINITE THICKNESS), '
     1A2,/,'    ?')
   36 READ(NI,37,ERR=44) ST
   37 FORMAT(F10.0)
      IF(ST.GT.0.0) GO TO 39
   38 IMAGE = 1
      ST = 1.0E32
   39 CONTINUE
      GO TO (50,400),IEDIT
C
C     POROSITY
   50 WRITE(NO,55)
   55 FORMAT(3X,'ENTER AQUIFER POROSITY',/,'    ?')
      READ(NI,56,ERR=50) P
   56 FORMAT(F10.0)
   57 IF(P.GT.0.0.AND.P.LT.1.0) GO TO 59
   54 WRITE(NO,58)
   58 FORMAT(3X,'POROSITY MUST BE GREATER THAN ZERO',
     1' AND LESS THAN ONE -- REENTER',/,'    ?')
      READ(NI,56,ERR=54) P
      GO TO 57
   59 GO TO (60,400),IEDIT
C
C     SEEPAGE VELOCITY
   60 WRITE(NO,65) IL,IT
   65 FORMAT(3X,'ENTER SEEPAGE VELOCITY, 'A2,'/',A2,/,'    ?')
      READ(NI,56,ERR=60) V
   66 IF(V.GT.0.0) GO TO 69
   64 WRITE(NO,67)
   67 FORMAT(3X,'SEEPAGE VELOCITY MUST BE GREATER THAN ZERO',
     1' -- REENTER',/,'    ?')
      READ(NI,56,ERR=64) V
      GO TO 66
```

```
   69 GO TO (70,400),IEDIT
C
C      RETARDATION COEFFICIENT
   70 WRITE(NO,75)
   75 FORMAT(3X,'ENTER RETARDATION COEFFICIENT',/,'    ?')
      READ(NI,56,ERR=70) RD
   76 IF(RD.GE.1.0) GO TO 79
   74 WRITE(NO,77)
   77 FORMAT(3X,'RETARDATION COEFFICIENT MUST BE GREATER THAN OR',
     1' EQUAL TO ONE',/,' -- REENTER',/,'    ?')
      READ(NI,56,ERR=74) RD
      GO TO 76
   79 GO TO (80,400),IEDIT
C
C      X DISPERSION COEFFICIENT
   80 WRITE(NO,81) IL,IT
   81 FORMAT(3X,'ENTER X DISPERSION COEFFICIENT, SQ ',A2,
     1'/',A2,/,'    ?')
   82 READ(NI,56,ERR=80) DX
      IF(DX.GT.0.0) GO TO 85
      WRITE(NO,83)
   83 FORMAT(3X,'X DISPERSION COEFFICIENT MUST BE GREATER THAN ZERO',
     1' -- REENTER',/,'    ?')
      GO TO 82
   85 GO TO (86,400),IEDIT
C
C      Y OR Z DISPERSION COEFFICIENT
   86 WRITE(NO,87) KHAR,IL,IT
   87 FORMAT(3X,'ENTER ',A1,' DISPERSION COEFFICIENT, SQ ',A2,
     1'/',A2,/,'    ?')
   88 READ(NI,56,ERR=86) DY
      IF(DY.GT.0.0) GO TO 90
      WRITE(NO,89) KHAR
   89 FORMAT(3X,A1,' DISPERSION COEFFICIENT MUST BE GREATER THAN ZERO',
     1' -- REENTER',/,'    ?')
      GO TO 88
   90 GO TO (91,400),IEDIT
C
C      FIRST-ORDER DECAY CONSTANT
   91 WRITE(NO,95) IT
   95 FORMAT(3X,'ENTER DECAY CONSTANT, 1/',A2,/,'    ?')
      READ(NI,56,ERR=91) DECAY
      GO TO (1320,400),IEDIT
C
C      SOURCE RATE SCHEDULE.
C
C
C      DEFINE LOCATIONS AND RATES OF SOURCES
C      INITIALIZE SOURCE/RATE ARRAYS
 1320 MAXRT2 = MAXRT + 2
      DO 1330  I=1,MAXSOR
        DO 1330  J=1,MAXRT2
            Q(I,J) = 0.0
            T(I,J) = 0.0
 1330 CONTINUE
      JFLOW = 3
 1340 WRITE(NO,1345)
 1345 FORMAT(3X,'SELECT TRANSIENT OR STEADY-STATE SOLUTION',/,
     16X,'TR FOR TRANSIENT SOLUTION',/,
     26X,'SS FOR STEADY-STATE SOLUTION',/,'    ?')
 1350 READ(NI,25) KSOL
      IF(KSOL.EQ.KSOL1) JFLOW=1
      IF(KSOL.EQ.KSOL2) JFLOW=2
      GO TO (1370,1370,1360), JFLOW
 1360 WRITE(NO,1365)
 1365 FORMAT(3X,'ERROR IN SELECTION -- REENTER'/,'    ?')
      GO TO 1350
C
 1370 WRITE(NO,1375) MAXSOR
 1375 FORMAT(3X,'ENTER THE NUMBER OF SOURCES (MAXIMUM OF',I3,' )',/,
```

```
     1'    ?')
1380 READ(NI,1385,ERR=1370) FDUM
1385 FORMAT(F10.0)
     NS=FDUM
     IF(NS.GT.0.AND.NS.LE.MAXSOR) GO TO 1400
     WRITE(NO,1395) MAXSOR
1395 FORMAT(3X,'NUMBER OF SOURCES MUST BE GREATER THAN ZERO '
    1'AND LESS THAN',I3,' -- REENTER',/,'    ?')
     GO TO 1380
1400 WRITE (NO,1405) IM1,IM2,IM3,IL,IT,IL,IT
1405 FORMAT(3X,'MASS RATES HAVE UNITS OF (',3A2,') (CU ',A2,'/',A2,
    1 ')/',A2,/,3X,'TIME HAS UNITS OF ',A2,/)
     DO 1540  I=1,NS
     IF(KFLOW.EQ.2) GO TO 1414
1406 WRITE(NO,1410) I,IL
1410 FORMAT(3X,'ENTER X AND Y COORDINATES OF SOURCE',I2,
    1' (',A2,')',/,'    ?,?')
     READ(NI,1425,ERR=1406) XS(I),YS(I)
     GO TO 1440
1414 WRITE(NO,1415) I,IL
1415 FORMAT(3X,'ENTER X AND Z COORDINATES OF SOURCE',I2,
    1' (',A2,')',/,'    ?,?')
     READ(NI,1425,ERR=1414) XS(I),YS(I)
1425 FORMAT(2F10.0)
1430 IF(YS(I).GE.0.0.AND.YS(I).LE.ST) GO TO 1440
1434 WRITE(NO,1435) ST,IL
1435 FORMAT(3X,'Z-COORDINATE MUST BE GREATER THAN OR EQUAL TO ZERO',
    1' AND',/,3X,'LESS THAN OR EQUAL TO SATURATED THICKNESS (',
    2F10.4,A3,')',/,3X,' -- REENTER',/,'    ?')
     READ(NI,37,ERR=1434) YS(I)
     GO TO 1430
1440 IF(JFLOW.EQ.2) GO TO 1530
     Q(I,1) = 0.0
     T(I,1) = 0.0
1450 WRITE(NO,1455) I,MAXRT
1455 FORMAT(3X,'ENTER THE NUMBER RATES FOR SOURCE',I2,
    1' (MAXIMUM OF',I3,')',/,'    ?')
1460 READ(NI,1465,ERR=1450) FDUM
1465 FORMAT(F10.0)
     NR(I)=FDUM
     IF(NR(I).GT.0.AND.NR(I).LE.MAXRT) GO TO 1480
     WRITE(NO,1475) MAXRT
1475 FORMAT(3X,'NUMBER OF RATES MUST BE GREATER THAN ZERO AND ',
    1'LESS THAN',I3,'  -- REENTER',/,'    ?')
     GO TO 1460
1480 CONTINUE
     NRT = NR(I)
     DO 1520  J=1,NRT
        M = J + 1
1484    WRITE(NO,1485) I,J,T(I,M-1),IT
1485    FORMAT(3X,'SOURCE ',I2,', RATE ',I2,' STARTS AT',F8.1,A3,/,
    1   3X,'ENTER MASS RATE AND ENDING TIME ',/,'    ?,?')
        READ(NI,1495,ERR=1484) Q(I,M),T(I,M)
1495    FORMAT(2F10.0)
1500    IF(T(I,M).GT.T(I,M-1)) GO TO 1510
1504    WRITE(NO,1505)
1505    FORMAT(3X,'ENDING TIME MUST BE GREATER THAN STARTING TIME '
    1   ' -- REENTER',/,'    ?')
        READ(NI,37,ERR=1504) T(I,M)
        GO TO 1500
1510    CONTINUE
1520 CONTINUE
     GO TO 1540
1530 WRITE(NO,1535) I
1535 FORMAT(3X,'ENTER STEADY-STATE MASS RATE',I2,/,'    ?')
     READ(NI,37,ERR=1530) Q(I,1)
     NR(I) = 0
1540 CONTINUE
     IF(IEDIT.EQ.2.AND.JFLOW.EQ.1.AND.TF.LE.1.0E-06) GO TO 720
 123 GO TO (124,400),IEDIT
```

```
C
C     COORDINATES OF THE OBSERVATION POINTS
  124 WRITE(NO,125) IL
  125 FORMAT(3X,'ENTER XFIRST, XLAST, DELTAX  (',A2,')' ,/,'    ?,?,?')
      READ(NI,126,ERR=124) XF(1),XL(1),DEL(1)
  126 FORMAT(3F10.0)
      DEL(1) = ABS(DEL(1))
      IF(DEL(1).LE.1.0E-06) XL(1)=XF(1)
      IF(KNTL.LE.0) GO TO 400
  133 IF(KFLOW.EQ.2) GO TO 136
  134 WRITE(NO,135) IL
  135 FORMAT(3X,'ENTER YFIRST, YLAST, DELTAY  (',A2,')',/,'    ?,?,?')
      READ(NI,126,ERR=134) XF(2),XL(2),DEL(2)
      DEL(2) = ABS(DEL(2))
      IF(DEL(2).LE.1.0E-06) XL(2)=XF(2)
      GO TO 145
  136 WRITE(NO,137) IL
  137 FORMAT(3X,'ENTER ZFIRST, ZLAST, DELTAZ  (',A2,')',/,'    ?,?,?')
      READ(NI,126,ERR=136) XF(2),XL(2),DEL(2)
      DEL(2) = ABS(DEL(2))
  138 IF(XF(2).GE.0.0.AND.DEL(2).LE.1.0E-06) GO TO 144
      IF(XF(2).GE.0.0.AND.XF(2).LE.ST) GO TO 142
      WRITE(NO,139)
  139 FORMAT(3X,'ZFIRST MUST BE GREATER THAN OR EQUAL TO ZERO')
      IF(IMAGE.GT.1) WRITE(NO,140) ST,IL
  140 FORMAT(3X,' AND LESS THAN OR EQUAL TO SATURATED THICKNESS
     1 (',F10.4,A3,')')
  130 WRITE(NO,141)
  141 FORMAT(3X,' -- REENTER',/,'    ?')
      READ(NI,56,ERR=130) XF(2)
      GO TO 138
  142 IF(XL(2).GE.0.0.AND.XL(2).LE.ST) GO TO 145
      WRITE(NO,143)
  143 FORMAT(3X,'ZLAST MUST BE GREATER THAN OR EQUAL TO ZERO')
      IF(IMAGE.GT.1) WRITE(NO,140) ST,IL
  131 WRITE(NO,141)
      READ(NI,56,ERR=131) XL(2)
      GO TO 142
  144 XL(2) = XF(2)
  145 GO TO (720,400),IEDIT
C
C     OBSERVATION TIMES
  720 IF(JFLOW.EQ.2) GO TO 770
  724 WRITE(NO,725) IT
  725 FORMAT(3X,'ENTER TFIRST, TLAST, DELTAT  (',A2,')',/,'    ?,?,?')
  730 READ(NI,735,ERR=724) TF,TL,DELT
  735 FORMAT(3F10.0)
      DELT = ABS(DELT)
  740 IF(TF.GT.0.0.AND.DELT.LE.1.0E-06) GO TO 760
      IF(TF.GT.0.0) GO TO 750
  744 WRITE(NO,745)
  745 FORMAT(3X,'TFIRST MUST BE GREATER THAN ZERO -- REENTER',/,'    ?')
      READ(NI,37,ERR=744) TF
      GO TO 740
  750 IF(TL.GT.0.0) GO TO 770
  754 WRITE(NO,755)
  755 FORMAT(3X,'TLAST MUST BE GREATER THAN ZERO -- REENTER',/,'    ?')
      READ(NI,37,ERR=754) TL
      GO TO 750
  760 TL = TF
  770 GO TO (146,780),IEDIT
  780 IF(JFLOW.EQ.2) WRITE(NO,785)
  785 FORMAT(3X,'TIME IS NOT A PARAMETER IN STEADY-STATE SOLUTION')
      GO TO 400
C
C     LIST PROBLEM DEFINITION
  146 WRITE(NO,147) NPAGE,(TITLE(I),I=1,30)
  147 FORMAT(1H1,/,3X,'PLUME2D',/,3X,'VERSION 2.02',
     1/,3X,'PAGE ',I3,///,3X,30A2,///)
      NPAGE = NPAGE + 1
```

```
      IF(IMAGE.GT.1) WRITE(NO,148) IL,ST
  148 FORMAT(1HO,7X,'SATURATED THICKNESS, (',A2,')   ',24X,F10.4)
      WRITE(NO,149) IL,IT,V,IL,IT,DX,KHAR,IL,IT,DY,P
  149 FORMAT(8X,'SEEPAGE VELOCITY, (',A2,'/',A2,') ',25X,F10.4,/,
     18X,'X DISPERSION COEFFICIENT (',A2,'**2/',A2,')    ',13X,F10.4,/,
     28X,A1,' DISPERSION COEFFICIENT (',A2,'**2/',A2,')    ',13X,F10.4,/,
     38X,'POROSITY ',42X,F10.4)
      WRITE(NO,150) RD,IT,DECAY
  150 FORMAT(//,8X,'RETARDATION COEFFICIENT',28X,F10.4,/,
     18X,'FIRST ORDER DECAY CONSTANT (1/',A2,')',18X,F10.4)
      GO TO (159,151), JFLOW
  151 WRITE(NO,153) KHAR,IL,IL,IM1,IM2,IM3,IL,IT,IL
  153 FORMAT(//,3X,'STEADY-STATE SOURCE RATES',//,
     13X,'SOURCE',6X,'X',11X,A1,17X,'RATE',/,
     25X,'NO',6X,'(',A2,')',8X,'(',A2,')',6X,'(',3A2,
     3')(CU ',A2,'/',A2,')/',A2,/)
      DO 157 I=1,NS
      WRITE(NO,155) I,XS(I),YS(I),Q(I,1)
  155 FORMAT(5X,I2,F10.2,2X,F10.2,6X,F16.4)
  157 CONTINUE
      GO TO 171
  159 WRITE(NO,160) IM1,IM2,IM3,IL,IT,IL,IT,IL,KHAR,IL
  160 FORMAT(//,3X,'SOURCE/RATE SCHEDULE  (',3A2,')(CU ',A2,'/',A2,
     1')/',A2,//,15X,'SOURCE',13X,'RATE',4X,'MASS',8X,'TIME (',A2,')',
     2/,3X,'NO     X (',A2,')      ',A1,' (',A2,')',9X,' NO',5X,'RATE',
     35X,'START',7X,'END',/)
      DO 170  I=1,NS
      WRITE(NO,165) I,XS(I),YS(I)
  165 FORMAT(/,3X,I2,2F9.2)
      NRT = NR(I)
      DO 170  J=1,NRT
         M = J + 1
         WRITE(NO,167) J,Q(I,M),T(I,M-1),T(I,M)
  167    FORMAT(34X,I2,F12.2,2F9.2)
  170 CONTINUE
  171 WRITE(NO,175) IL,XF(1),XL(1),DEL(1),KHAR,XF(2),KHAR,XL(2),KHAR,
     1 DEL(2)
  175 FORMAT(//,8X,'OBSERVATION POINTS (',A2,')',//,
     112X,'XFIRST =',F10.2,3X,'XLAST =',F10.2,3X,'DELX =',F10.4,/,
     212X,A1,'FIRST =',F10.2,3X,A1,'LAST =',F10.2,3X,'DEL',A1,' =',
     1F10.4)
      IF(JFLOW.EQ.1) WRITE(NO,177) IT,TF,TL,DELT
  177 FORMAT(/,8X,'OBSERVATION TIMES (',A2,')',//,
     1 12X,'TFIRST =',F10.2,3X,'TLAST =',F10.2,3X,'DELT =',F10.4)
  180 CONTINUE
      GO TO 400
C
C
C
C ***** SECTION II -- NUMERICAL EVALUATION OF CONCENTRATION AT
C                     SPECIFIED GRID COORDINATES
C
C     NUMBER OF OBSERVATION POINTS IN EACH COORDINATE DIRECTION
 2000 CONTINUE
      DO 2020 L=1,2
         NP(L) = 1
         DEL(L) = ABS(DEL(L))
         IF(DEL(L).LE.1.0E-03) GO TO 2020
         DIF = XL(L) - XF(L)
         IF(ABS(DIF).LE.1.0E-03) GO TO 2020
         IF(DIF.LE.0.0) DEL(L)=-DEL(L)
         NPTS = ABS(DIF/DEL(L))
         REM = DIF - DEL(L)*FLOAT(NPTS)
         NPTS = NPTS + 1
         NP(L) = NPTS
         IF(ABS(REM).LT.1.0E-03) GO TO 2020
         NP(L) = NP(L) + 1
 2020 CONTINUE
      MAXRW = NP(2)
      MAXCL = NP(1)
```

```
C
C     TIME COORDINATES
      NTIME = 1
      IF(DELT.LE.1.0E-06) GO TO 2110
      NTIME = ABS(TL-TF)/DELT + 1.0
      IF(TF.GT.TL) DELT=-DELT
 2110 TSOL = TF
      MTIME = NTIME
C
C
      DAMK = DX*DECAY*RD/(V*V)
      ALPHA=SQRT(1.0+4.0*DAMK)
      PE=V/DX
      BETA = DX/DY
      LAMBDA = 1.0/(12.566731*P*SQRT(DX*DY))
C
C
      DO 2660  NT=1,NTIME
C
 2120 LPRT = 1
      LP = 1
      NCFLG = 1
 2140 NROW1 = 1
      NROW2 = MAXROW
 2160 IF(NROW2.GT.MAXRW) NROW2=MAXRW
      DO 2580  NROW=NROW1,NROW2
      GO TO (2180,2220,2200),NCFLG
 2180 NCOL1 = 1
      NCOL2 = MAXCOL
 2200 IF(NCOL2.GT.MAXCL) NCOL2=MAXCL
      NCOL = MAXCOL
      IF(NCOL2.EQ.MAXCL) NCOL=NCOL2-NCOL1+1
 2220 IX1 = NCOL1
      IX2 = NCOL2
C
      DO 2300  L=1,MAXCOL
         CON(L) = 0.0
 2300 CONTINUE
C
      DO 2440   N=1,NS
         D(1) = ST - YS(N)
         IF(ST.GE.0.9E32) D(1)=0.0
         D(2) = YS(N)
         D(3) = D(1)
         COEF = 1.0
         IF(D(1).LT.1.0E-03.OR.D(2).LT.1.0E-03) COEF=2.0
         DO 2440  I=IX1,IX2
            X = XF(1) + FLOAT(I-1)*DEL(1)
            IF(I.EQ.NP(1)) X=XL(1)
            XXS = X - XS(N)
            PEX = PE*XXS
            Y = XF(2) + FLOAT(NROW-1)*DEL(2)
            IF(NROW.EQ.NP(2)) Y=XL(2)
            YYS = Y - YS(N)
            PEY = PE*YYS
            L = I-IX1 + 1
            IF(CON(L).LT.0.0) GO TO 2430
            IF(ABS(XXS).LT.1.0.AND.ABS(YYS).LT.1.0)GO TO 2330
            CALL SOL2D(C,PEX,PEY,TSOL,N,NR(N))
            CXYT = COEF*C
            IF(IMAGE.EQ.1) GO TO 2325
C
            DO 2320  LM=1,2
               ZM = ((-1.0)**(LM+1))*YYS
               IF(D(LM).LT.1.0E-03) GO TO 2320
               ZIMAGE = 2.0*D(LM) - ZM
               PEZ = PE*ZIMAGE
               CALL SOL2D(C,PEX,PEZ,TSOL,N,NR(N))
               CXYT = CXYT + COEF*C
               DO 2310  IM = 1,IMAGE
```

```
                ZIMAGE = (2.0*D(LM)+ZM) + 2.0*FLOAT(IM)*D(LM+1)
     1                   + FLOAT(2*IM-2)*D(LM)
                PEZ = PE*ZIMAGE
                CALL SOL2D(C,PEX,PEZ,TSOL,N,NR(N))
                IF(C.LT.1.0E-06) GO TO 2312
                CXYT = CXYT + COEF*C
 2310         CONTINUE
 2312         CONTINUE
              DO 2314  IM=1,IMAGE
                ZIMAGE = (2.0*D(LM)-ZM) + 2.0*FLOAT(IM)*D(LM+1)
     1                   + FLOAT(2*IM)*D(LM)
                PEZ = PE*ZIMAGE
                CALL SOL2D(C,PEX,PEZ,TSOL,N,NR(N))
                IF(C.LT.1.0E-06) GO TO 2320
                CXYT = CXYT + COEF*C
 2314         CONTINUE
              WRITE(NO,2315) MAXIMG,X,Y
 2315         FORMAT(3X,'****** WARNING -- SOLUTION DID NOT',
     1        ' CONVERGE USING',/,9X,I2,' IMAGE WELLS AT X =',
     2        F10.4,'  Z =',F10.4)
 2320     CONTINUE
 2325     IF(KFLOW.EQ.1) CXYT=CXYT/2.0
          CON(L) = CON(L) + CXYT
          GO TO 2340
 2330     CON(L) = -9.9999
 2340     ROW = Y
          COL(L) = X
 2430     CONTINUE
 2440 CONTINUE
C
C
C     PRINT CONCENTRATION DISTRIBUTION
      GO TO (2460,2560), LPRT
 2460 WRITE(NO,147) NPAGE, (TITLE(I),I=1,30)
      NPAGE = NPAGE + 1
      IF(JFLOW.EQ.2) GO TO 2500
      WRITE(NO,2465) TSOL,IT,IM1,IM2,IM3,
     1(LBL(LP,L),L=1,6),IL
 2465 FORMAT(13X,'CONCENTRATION DISTRIBUTION AT ',F10.2,
     11X,A2,' (',3A2,') ',//,13X,3X,6A2,//,
     2' *',/,'    * X(',A2,')')
      GO TO 2520
 2500 WRITE(NO,2505) IM1,IM2,IM3,(LBL(LP,L),L=1,6),
     1IL
 2505 FORMAT(13X,'CONCENTRATION DISTRIBUTION AT STEADY STATE',
     1' (',3A2,') ',//13X,3X,6A2,//,
     2' *',/,'    * X(',A2,')')
 2520 CONTINUE
      WRITE(NO,2525) (COL(L),L=1,NCOL)
 2525 FORMAT('      *',4X,7F10.2)
      WRITE(NO,2545) KHAR,IL
 2545 FORMAT(1X,A1,'(',A2,') *',/,9X,'*')
C
 2560 WRITE(NO,2565) ROW,(CON(L),L=1,NCOL)
 2565 FORMAT(2X,F8.2,7F10.4)
      LPRT = 2
 2580 CONTINUE
      IF(NROW2.EQ.MAXRW) GO TO 2600
      NROW1 = NROW1 + MAXROW
      NROW2 = NROW2 + MAXROW
      LPRT = 1
      LP = 2
      NCFLG = 2
      GO TO 2160
 2600 IF(NCOL2.EQ.MAXCL) GO TO 2640
      NCOL1 = NCOL1 + MAXCOL
      NCOL2 = NCOL2 + MAXCOL
      LPRT = 1
      LP = 2
      NCFLG = 3
```

```
      GO TO 2140
 2640 CONTINUE
      TSOL = TSOL + DELT
      IF(NT.EQ.MTIME) TSOL=TL
 2660 CONTINUE
C ***** SECTION III -- PROBLEM REDEFINITION AND CONTROL OF EXECUTION
C
C
  400 CONTINUE
      IF(IEDIT.EQ.2)GO TO 401
      WRITE(NO,1001)KHAR,KHAR,KHAR,KHAR
      IEDIT = 2
  401 KNTL = 0
      WRITE(NO,405)
  405 FORMAT(//,3X,'ENTER NEXT COMMAND',/,'    ?')
  410 READ(NI,415) NEXT
  415 FORMAT(A2)
C
      DO 420  I=1,20
         IF(NEXT.EQ.IC(I)) GO TO 430
  420 CONTINUE
      WRITE(NO,425)
  425 FORMAT(3X,'ERROR IN LAST COMMAND -- REENTER',/,'    ?')
      GO TO 410
  430 GO TO (91,60,70,80,86,86,50,450,124,133,133,3060,1,
     12000,700,146,1000,602,1320,720),I
C
C     NEW SET OF X AND Y OBSERVATIONS
  450 KNTL = 1
      GO TO 124
C
C
C     NEW SOURCE/RATE SCHEDULE
 3060 WRITE(NO,3065)
 3065 FORMAT(3X,'ADD(A),DELETE(D),MODIFY(M) A SOURCE OR RETURN(R)'
     1' TO EDIT ?')
 3070 READ(NI,3075) ISK
 3075 FORMAT(A1)
      DO 3080  I=1,4
         IF(ISK.EQ.IS(I)) GO TO 3090
 3080 CONTINUE
      WRITE(NO,3085)
 3085 FORMAT(3X,'ERROR IN SELECTION -- REENTER ?')
      GO TO 3070
 3090 GO TO (400,3100,3450,3490),I
C
C     MODIFY SOURCE
C
 3100 WRITE(NO,3105) NS
 3105 FORMAT(3X,I2,' SOURCES IN CURRENT SCHEDULE',/,
     13X,'ENTER SOURCE TO MODIFY',/,'    ?')
      READ(NI,1465,ERR=3100) FDUM
      JS=FDUM
      IF(JS.GT.0.AND.JS.LE.NS) GO TO 3220
      WRITE(NO,3215) JS
 3215 FORMAT(3X,'SOURCE',I4,' NOT IN SCHEDULE')
      GO TO 3060
 3220 GO TO (3230,3260),JFLOW
 3230 WRITE(NO,3235) JS,XS(JS),IL,KHAR,YS(JS),IL,IT,
     1IM1,IM2,IM3,IL,IT,IL
 3235 FORMAT(3X,'SOURCE ',I2,':    X =',F8.2,A3,2X,',',A1,' =',
     1F8.2,A3,//,3X,'RATE',7X,'MASS RATE',14X,'TIME (',
     2A2,')',/,4X,'NO',3X,'(',3A2,')(CU ',A2,'/',A2,')/',A2,
     35X,'START',7X,'END',/)
      NRT = NR(JS)
      DO 3250  J=1,NRT
         M = J + 1
         WRITE(NO,3245) J,Q(JS,M),T(JS,M-1),T(JS,M)
 3245    FORMAT(4X,I2,5X,F14.2,7X,F8.3,3X,F8.2)
 3250 CONTINUE
```

```
      GO TO 3270
 3260 WRITE(NO,3265) JS,XS(JS),IL,KHAR,YS(JS),IL,Q(JS,1),
     1IM1,IM2,IM3,IL,IT,IL
 3265 FORMAT(3X,'SOURCE ',I2,':    X =',F8.2,A3,2X,',',A1,' =',
     1F8.2,A3,/,3X,'STEADY-STATE MASS RATE =',F16.4,
     2' (',3A3,')(CU ',A2,'/',A2,')/',A2,/)
 3270 WRITE(NO,3275)
 3275 FORMAT(3X,'CHANGE COORDINATES (Y/N)?')
      READ(NI,3075) JC
      IF(JC.NE.IY) GO TO 3290
      IF(KFLOW.EQ.2) GO TO 3277
 3276 WRITE(NO,1410) JS,IL
      READ(NI,1425,ERR=3276) XS(JS),YS(JS)
      GO TO 3290
 3277 WRITE(NO,1415) JS,IL
      READ(NI,1425,ERR=3277) XS(JS),YS(JS)
 3280 IF(YS(JS).GE.0.0.AND.YS(JS).LE.ST) GO TO 3290
 3281 WRITE(NO,1435) ST,IL
      READ(NI,37,ERR=3281) YS(JS)
      GO TO 3280
 3290 GO TO (3300,3430),JFLOW
C
C     TRANSIENT SOURCES
 3300 WRITE(NO,3305) JS
 3305 FORMAT(3X,'MODIFY RATE SCHEDULE FOR SOURCE',I3,' (Y/N) ?')
      READ(NI,3075) JY
      IF(JY.NE.IY) GO TO 3060
 3310 WRITE(NO,3315)
 3315 FORMAT(3X,'ENTER RATE TO BE CHANGED',/,
     13X,'(ENTER 0 TO CHANGE ALL RATES)',/,'    ?')
      READ(NI,1465,ERR=3310) FDUM
      JR=FDUM
      IF(JR.LE.0) GO TO 3350
      IF(JR.LE.NR(JS)) GO TO 3330
      WRITE(NO,3325) JR
 3325 FORMAT(3X,'RATE ',I2,' NOT IN CURRENT SCHEDULE')
      GO TO 3300
 3330 WRITE(NO,3335) JS,JR,T(JS,JR),T(JS,JR+1),IT
 3335 FORMAT(3X,'SOURCE ',I2,', RATE ',I2,' STARTS AT',F8.2,
     1' AND ENDS AT',F8.2,A3,/,3X,'ENTER NEW MASS RATE',/,'    ?')
      M = JR + 1
      READ(NI,3345,ERR=3330) Q(JS,M)
 3345 FORMAT(F10.0)
      GO TO 3300
C
 3350 NRT = NR(JS)
      DO 3360  J=1,NRT
         M = J + 1
         Q(JS,M) = 0.0
         T(JS,M) = 0.0
 3360 CONTINUE
 3370 WRITE(NO,1455) JS,MAXRT
 3380 READ(NI,1465,ERR=3370) FDUM
      NR(JS)=FDUM
      IF(NR(JS).GT.0.AND.NR(JS).LE.MAXRT) GO TO 3390
      WRITE(NO,1475) MAXRT
      GO TO 3380
 3390 CONTINUE
      NRT = NR(JS)
      DO  3420  J=1,NRT
         M = J + 1
 3394    WRITE(NO,1485) JS,J,T(JS,M-1),IT
         READ(NI,1495,ERR=3394) Q(JS,M),T(JS,M)
 3400    IF(T(JS,M).GT.T(JS,M-1)) GO TO 3410
 3404    WRITE(NO,1505)
         READ(NI,37,ERR=3404) T(JS,M)
         GO TO 3400
 3410    CONTINUE
 3420 CONTINUE
      GO TO 3060
```

```
C
C     STEADY-STATE SOURCES
 3430 WRITE(NO,3435) JS
 3435 FORMAT(3X,'CHANGE STEADY-STATE RATE FOR SOURCE ',I2,' (Y/N) ?')
      READ(NI,3075) JC
      IF(JC.NE.IY) GO TO 3060
 3444 WRITE(NO,3445) JS
 3445 FORMAT(3X,'ENTER NEW STEADY-STATE MASS RATE FOR SOURCE ',I2,/,
     1'    ?')
      READ(NI,3345,ERR=3444) Q(JS,1)
      GO TO 3060
C
C     ADD A NEW SOURCE
C
 3450 NS = NS + 1
      JS = NS
      IF(KFLOW.EQ.2) GO TO 3455
 3454 WRITE(NO,1410) JS,IL
      READ(NI,1425,ERR=3454) XS(JS),YS(JS)
      GO TO 3470
 3455 WRITE(NO,1415) JS,IL
      READ(NI,1425,ERR=3455) XS(JS),YS(JS)
 3460 IF(YS(JS).GE.0.0.AND.YS(JS).LE.ST) GO TO 3470
 3464 WRITE(NO,1435) ST,IL
      READ(NI,37,ERR=3464) YS(JS)
      GO TO 3460
 3470 GO TO (3370,3480),JFLOW
C
C     STEADY-STATE SOURCES
 3480 WRITE(NO,3485) JS
 3485 FORMAT(3X,'ENTER STEADY-STATE MASS RATE FOR SOURCE ',I2,
     1/,'    ?')
      READ(NI,3345,ERR=3480) Q(JS,1)
      NR(JS) = 0
      GO TO 3060
C
C     DELETE A SOURCE
C
 3490 IF(NS.GT.1) GO TO 3500
      WRITE(NO,3495)
 3495 FORMAT(3X,'ONLY ONE SOURCE IN SCHEDULE -- CAN NOT DELETE',/)
      GO TO 3060
 3500 WRITE(NO,3505) IL,IL,IL
 3505 FORMAT(3X,'SOURCE',6X,'X (',A2,')',3X,'Y (',A2,')',3X,
     1'Z (',A2,')',/)
      DO 3520  I=1,NS
         WRITE(NO,3515) I,XS(I),YS(I)
 3515    FORMAT(5X,I2,3X,F8.2,3X,F8.2)
 3520 CONTINUE
 3530 WRITE(NO,3535)
 3535 FORMAT(3X,'ENTER SOURCE TO DELETE',/,
     13X,'(ENTER 0 TO CANCEL)',/,'    ?')
      READ(NI,1465,ERR=3530) FDUM
      JS=FDUM
      IF(JS.LE.0) GO TO 3060
      IF(JS.LE.NS) GO TO 3550
      WRITE(NO,3545) JS
 3545 FORMAT(3X,'SOURCE ',I2,' NOT IN CURRENT SCHEDULE')
      GO TO 3530
 3550 WRITE(NO,3555) JS
 3555 FORMAT(3X,'DELETE SOURCE ',I2,' (Y/N)?')
      READ(NI,3075) JC
      IF(JC.NE.IY) GO TO 3530
      NSD = NS - 1
      GO TO (3560,3590),JFLOW
C
C     TRANSIENT SOURCES
 3560 IF(JS.EQ.NS) GO TO 3575
      DO  3570  J=JS,NSD
         XS(J) = XS(J+1)
```

```
            YS(J) = YS(J+1)
            NR(J) = NR(J+1)
            NRT = NR(J)
               DO 3570  K=1,NRT
               M = K + 1
               Q(J,M) = Q(J+1,M)
               T(J,M) = T(J+1,M)
 3570 CONTINUE
 3575 NRT = NR(NS)
      DO 3580  K=1,NRT
         M = K + 1
         Q(NS,M) = 0.0
         T(NS,M) = 0.0
 3580 CONTINUE
      NR(NS) = 0
      NS = NSD
      GO TO 3060
C
C     STEADY-STATE SOURCES
 3590 IF(JS.EQ.NS) GO TO 3605
      DO 3600   J=JS,NSD
         Q(J,1) = Q(J+1,1)
         XS(J) = XS(J+1)
         YS(J) = YS(J+1)
 3600 CONTINUE
 3605 Q(NS,1) = 0.0
      NS = NSD
      GO TO 3060
C
C     CONFIRM WHETHER SATURATED THICKNESS IS A VARIABLE
  602 IF(KFLOW.EQ.2)GO TO 30
      WRITE(NO,605)
  605 FORMAT(3X,'SATURATED THICKNESS IS NOT A VARIABLE IN',/,
     13X,'X-Y COORIDINATE SYSTEM (VERTICALLY AVERAGED SOLUTION)')
      GO TO 400
C
C     MENU OF EDIT COMMANDS FOR PLUMES VERSION 2.02
 1000 WRITE(NO,1001)KHAR,KHAR,KHAR,KHAR
 1001 FORMAT(1H1,/,3X,'MENU OF EDIT COMMANDS',//,
     1'  RETARDATION COEFFICIENT  RD      OBSERVATION POINTS  OB',/,
     2'  POROSITY                 PO      X COORDINATES       XC',/,
     3'  SEEPAGE VELOCITY         VX',6X,A1,' COORDINATES',7X,A1'C',/,
     4'  X DISPERSION COEFFICIENT DX      MENU OF COMMANDS    MU',/,
     52X,A1,' DISPERSION COEFFICIENT D',A1,6X,'LIST INPUT DATA      LI',
     6/,'  DECAY CONSTANT            DE      RUN CALCULATIONS    RN',/,
     7'  SOURCE RATE SCHEDULE     RT      DONE                DN',/,
     8'  NEW PROBLEM              NP      SATURATED THICKNESS ST',/,
     9'  CHANGE SOLUTION/SOURCES  CS      OBSERVATION TIMES   TC')
      GO TO 400
C
  700 STOP
      END
```

APPENDIX E

Listing of Utility Function Subroutines

```
      FUNCTION BIO(Z)
C     JAN WAGNER
C     SCHOOL OF CHEMICAL ENGINEERING
C     OKLAHOMA STATE UNIVERSITY
C     STILLWATER, OK  74078
C     TELEPHONE: (405) 624-5280
C
C     REVISED: 6 JANUARY 1983
C
C     EVALUATION OF MODIFIED BESSEL FUNCTION OF THE FIRST KIND
C     OF ORDER ZERO
C     POLYNOMIAL APPROXIMATIONS ARE USED FOR IO(Z)
C     SEE SECTION 9.8 OF ABRAMOWITZ AND STEGUN (1966)
      DIMENSION A(9)
      COMMON/IO/NI,NO
      DATA A/0.9189385,4.32105045,6.09540829,6.45308739,4.6926023,
     13.88357842,3.63608323,4.10583047,5.540702353/
C
      IF(Z.LE.0.0) GO TO 200
      T = Z/3.75
      IF(Z.GT.3.75) GO TO 100
      T2 = T*T
      T4 = T2*T2
      T6 = T2*T4
      T8 = T2*T6
      T10 = T2*T8
      T12 = T2*T10
      BIO = 1.0 + 3.5156229*T2 + 3.0899424*T4 + 1.2067492*T6
     1 + 0.2659732*T8 + 0.0360768*T10+ 0.0045813*T12
      RETURN
  100 CONTINUE
      SUM = 0.0
      DO 150  I=1,9
      SIGN = (-1.0)**(I+1)
      IF(I.EQ.2) SIGN=1.0
      ARG = -1.0*A(I) - 0.5*ALOG(Z) - FLOAT(I-1)*ALOG(T)
      SUM = SUM + SIGN*EXP(ARG)
  150 CONTINUE
      BIOLOG = ALOG(SUM) + Z
      BIO = EXP(BIOLOG)
      RETURN
  200 CONTINUE
      IF(Z.LT.0.0) GO TO 300
      BIO = 1.0
      RETURN
  300 WRITE(NO,305) Z
  305 FORMAT(6X,'ARGUMENT OF BESSEL FUNCTION IO(Z) IS NEGATIVE',/,
     16X,'Z = ',E12.6,'  -- PROGRAM TERMINATED')
      STOP
      END
```

```
      FUNCTION BIOLOG(Z)
C     JAN WAGNER
C     SCHOOL OF CHEMICAL ENGINEERING
C     OKLAHOMA STATE UNIVERSITY
C     STILLWATER, OK  74078
C     TELEPHONE: (405) 624-5280
C
C     REVISED: 6 JANUARY 1983
C
C     EVALUATION OF THE NATURAL LOG OF A MODIFIED BESSEL
C     FUNCTION OF THE FIRST KIND OF ORDER ZERO
C     POLYNOMIAL APPROXIMATIONS ARE USED FOR IO(Z)
C     SEE SECTION 9.8 OF ABRAMOWITZ AND STEGUN (1966)
      DIMENSION A(9)
      COMMON/IO/NI,NO
      DATA A/0.9189385,4.32105045,6.09540829,6.45308739,4.6926023,
     13.88357842,3.63608323,4.10583047,5.540702353/
C
      IF(Z.LE.0.0) GO TO 200
      T = Z/3.75
      IF(Z.GT.3.75) GO TO 100
      T2 = T*T
      T4 = T2*T2
      T6 = T2*T4
      T8 = T2*T6
      T10 = T2*T8
      T12 = T2*T10
      BIO = 1.0 + 3.5156229*T2 + 3.0899424*T4 + 1.2067492*T6
     1            + 0.2659732*T8 + 0.0360768*T10 + 0.0045813*T12
      BIOLOG = ALOG(BIO)
      RETURN
  100 CONTINUE
      SUM = 0.0
      DO 150  I=1,9
      SIGN = (-1.0)**(I+1)
      IF(I.EQ.2) SIGN=1.0
      ARG = -1.0*A(I) - 0.5*ALOG(Z) - FLOAT(I-1)*ALOG(T)
      SUM = SUM + SIGN*EXP(ARG)
  150 CONTINUE
      BIOLOG = ALOG(SUM) + Z
      RETURN
  200 CONTINUE
      IF(Z.LT.0.0) GO TO 300
      BIO = 1.0
      RETURN
  300 WRITE(NO,305) Z
  305 FORMAT(6X,'ARGUMENT OF BESSEL FUNCTION IO(Z) IS NEGATIVE',/,
     16X,'Z = ',E12.6,'  -- PROGRAM TERMINATED')
      STOP
      END
```

```
      FUNCTION BKO(Z)
C     JAN WAGNER
C     SCHOOL OF CHEMICAL ENGINEERING
C     OKLAHOMA STATE UNIVERSITY
C     STILLWATER, OK  74078
C     TELEPHONE: (405) 624-5280
C
C     REVISED: 6 JANUARY 1983
C
C     EVALUATION OF MODIFIED BESSEL FUNTION OF SECOND KIND
C     OF ORDER ZERO
C     POLYNOMIAL APPROXIMATIONS ARE USED FOR KO(Z)
C     SEE SECTION 9.8 OF ABRAMOWITZ AND STEGUN (1966)
      COMMON/IO/NI,NO
C
      IF(Z.LE.0.0) GO TO 200
      T = Z/2.0
      T2 = T*T
      T4 = T2*T2
      T6 = T2*T4
      IF(Z.GT.2.0) GO TO 100
      T8 = T2*T6
      T10 = T2*T8
      T12 = T2*T10
      BKO = -1.0*ALOG(T)*BIO(Z) - 0.57721566
     1      + 0.42278420*T2 + 0.23069756*T4 + 0.03488590*T6
     2      + 0.00262698*T8 + 0.00010750*T10+ 0.0740E-04*T12
      RETURN
  100 CONTINUE
      SUM = (1.25331414 - 0.07832358/T + 0.02189568/T2
     1      - 0.01062446/(T*T2) + 0.00587872/T4
     2      - 0.00251540/(T*T4) + 0.00053208/T6)
      BKOLOG = ALOG(SUM) - Z - 0.5*ALOG(Z)
      BKO = EXP(BKOLOG)
      RETURN
  200 CONTINUE
      WRITE(NO,205) Z
  205 FORMAT(6X,'ARGUMENT OF BESSEL FUNCTION KO(Z) IS LESS THAN',
     1' OR EQUAL TO ZERO',/,6X,'Z = ',E12.6,' -- PROGRAM TERMINATED')
      STOP
      END
```

```
      FUNCTION BKOLOG(Z)                                                BKOL001
C     JAN WAGNER                                                        BKOL002
C     SCHOOL OF CHEMICAL ENGINEERING                                    BKOL003
C     OKLAHOMA STATE UNIVERSITY                                         BKOL004
C     STILLWATER, OK  74078                                             BKOL005
C     TELEPHONE: (405) 624-5280                                         BKOL006
C                                                                       BKOL007
C     REVISED: 6 JANUARY 1983                                           BKOL008
C                                                                       BKOL009
C     NATURAL LOG OF MODIFIED BESSEL FUNTION OF SECOND KIND             BKOL010
C     OF ORDER ZERO                                                     BKOL011
C     POLYNOMIAL APPROXIMATIONS ARE USED FOR KO(Z)                      BKOL012
C     SEE SECTION 9.8 OF ABRAMOWITZ AND STEGUN (1966)                   BKOL013
      COMMON/IO/NI,NO                                                   BKOL014
C                                                                       BKOL015
      IF(Z.LE.0.0) GO TO 200                                            BKOL016
      T = Z/2.0                                                         BKOL017
      T2 = T*T                                                          BKOL018
      T4 = T2*T2                                                        BKOL019
      T6 = T2*T4                                                        BKOL020
      IF(Z.GT.2.0) GO TO 100                                            BKOL021
      T8 = T2*T6                                                        BKOL022
      T10 = T2*T8                                                       BKOL023
      T12 = T2*T10                                                      BKOL024
      BKO = -1.0*ALOG(T)*BIO(Z) - 0.57721566                            BKOL025
     1     + 0.42278420*T2 + 0.23069756*T4 + 0.03488590*T6              BKOL026
     2     + 0.00262698*T8 + 0.00010750*T10+ 0.0740E-04*T12             BKOL027
      BKOLOG = ALOG(BKO)                                                BKOL028
      RETURN                                                            BKOL029
  100 CONTINUE                                                          BKOL030
      SUM = (1.25331414 - 0.07832358/T + 0.02189568/T2                  BKOL031
     1     - 0.01062446/(T*T2) + 0.00587872/T4                          BKOL032
     2     - 0.00251540/(T*T4) + 0.00053208/T6)                         BKOL033
      BKOLOG = ALOG(SUM) - Z - 0.5*ALOG(Z)                              BKOL034
      RETURN                                                            BKOL035
  200 CONTINUE                                                          BKOL036
      WRITE(NO,205) Z                                                   BKOL037
  205 FORMAT(6X,'ARGUMENT OF BESSEL FUNCTION KO(Z) IS LESS THAN',       BKOL038
     1' OR EQUAL TO ZERO',/,6X,'Z = ',E12.6,'  -- PROGRAM TERMINATED')  BKOL039
      STOP                                                              BKOL040
      END                                                               BKOL041
```

```
      FUNCTION ERFC(Z)
C     JAN WAGNER
C     SCHOOL OF CHEMICAL ENGINEERING
C     OKLAHOMA STATE UNIVERSITY
C     STILLWATER, OK  74078
C     TELEPHONE: (405) 624-5280
C
C     REVISED: 6 JANUARY 1983
C
C     RATIONAL APPROXIMATION OF THE COMPLIMENTARY ERROR FUNCTION
C     SEE SECTION 7.1 OF ABRAMOWITZ AND STEGUN (1966)
C     THE FOLLOWING IDENTITIES ARE USED TO HANDLE NEGATIVE ARGUMENTS
C      ERFC( Z) = 1 - ERF(Z)
C      ERF (-Z) = -ERF (Z)
C
      REAL*8 COEFLG,DERFC,DI,FX,TERMI,TERMO,SUM,X
      COMMON/IO/NI,NO
C
      X = ABS(Z)
      IF (X.GT.3.0D00) GO TO 50
C
C     FOR X<3 A RATIONAL APPROXIMATION OF THE COMPLIMENTARY ERROR
C     FUNCTION IS USED.
C
      DERFC = 1.0D00/((1.0D00 + 7.05230784D-02*X + 4.22820123D-02*(X**2)
     1   + 9.2705272D-03*(X**3) + 1.520143D-04*(X**4)
     2   + 2.76572D-04*(X**5) + 4.30638D-05*(X**6))**16)
      GO TO 100
C
C     FOR X>3 AN ASYMPTOTIC EXPANSION OF THE COMPLIMENTARY ERROR
C     FUNCTION IS USED.
   50 COEFLG = X*X + DLOG(X) + 0.57236494D00
      FX = 2.0D00*X*X
      SUM = 1.0D00
      TERMO = 1.0D00
      DO 60 I=2,50
      DI = I
      TERMI = -TERMO*(2.0D00*DI - 3.0D00)/FX
      IF(DABS(TERMI).GT.DABS(TERMO)) GO TO 70
      SUM = SUM + TERMI
      TEST = TERMI/SUM
      IF(ABS(TEST).LT.1.0E-16) GO TO 70
      TERMO = TERMI
   60 CONTINUE
      WRITE(NO,65)
   65 FORMAT(6X,'*** WARNING -- ASYMPTOTIC EXPANSION FOR ERFC DID NOT',
     1'                        CONVERGE WITH 50 TERMS IN THE SUMMATION')
   70 SUM = DLOG(SUM) - COEFLG
      IF(SUM.LT.-72.0D00) SUM=-72.0D00
      DERFC = DEXP(SUM)
  100 CONTINUE
C
C     FOR Z<0, ERFC(-Z) = 2-ERFC(Z)
      ERFC = DERFC
      IF(Z.LT.0.0) ERFC=2.0D00-DERFC
      RETURN
      END
```

```
      FUNCTION E1LOG(Z)
C     JAN WAGNER
C     SCHOOL OF CHEMICAL ENGINEERING
C     OKLAHOMA STATE UNIVERSITY
C     STILLWATER, OK  74078
C     TELEPHONE: (405) 624-5280
C
C     REVISED: 6 JANUARY 1983
C
C     EVALUATION OF THE NATURAL LOG OF THE EXPONENTIAL INTEGRAL
C     POLYNOMIAL APPROXIMATIONS ARE USED FOR E1(Z)
C     SEE SECTION 5.1 OF ABRAMOWITZ AND STEGUN (1966)
      COMMON/IO/NI,NO
C
      IF(Z.LE.0.0) GO TO 200
      Z2 = Z*Z
      Z3 = Z*Z2
      IF(Z.GT.1.0) GO TO 100
C     ARGUMENTS LESS THAN UNITY
      E1 = - 0.57721566    + 0.99999193*Z     - 0.24991055*Z2
     1    + 0.05519968*Z3 - 0.00976004*Z2*Z2 + 0.00107857*Z2*Z3
     2    - ALOG(Z)
      E1LOG = ALOG(E1)
      RETURN
  100 CONTINUE
C     ARGUMENTS GREATER THAN UNITY
      E1LOG = ALOG(Z2*Z2 +  8.5733287401*Z3 + 18.0590169730*Z2
     1     +  8.6347608925*Z  +  0.2677737343 )
     2       - ALOG(Z2*Z2 +  9.5733223454*Z3 + 25.6329561486*Z2
     3    + 21.0996530827*Z  +  3.9584969228 )
     4       - ALOG(Z) - Z
      RETURN
  200 WRITE(NO,205) Z
  205 FORMAT(6X,'ARGUMENT OF EXPONENTIAL INTEGRAL IS LESS THAN',
     1' OR EQUAL TO ZERO',/,6X,'Z = ',E12.6,3X,' -- PROGRAM',
     2' TERMINATED')
      STOP
      END
```

```
      FUNCTION FUNCTN(Z)                                              FUNC001
C     INTEGRAND OF HANTUSH WELL FUNCTION                              FUNC002
      REAL*8 DB,Z,ARG,FUNCTN                                          FUNC003
      COMMON BF                                                       FUNC004
      DB=BF                                                           FUNC005
      ARG = DLOG(Z) + Z + DB*DB/(4.0D00*Z)                            FUNC006
      FUNCTN = DEXP(-ARG)                                             FUNC007
      RETURN                                                          FUNC008
      END                                                             FUNC009
```

```
      FUNCTION GAUSS(A,B,FUNCTN)
C
C     NUMERICAL INTEGRATION  BY 24 POINT GAUSS-LEGENDRE QUADRATURE
C     ZEROS AND WEIGHTING FACTORS ARE FROM TABLE 25.4, P916, OF
C     ABRAMOWITZ AND STENGUN(1966)
C
      REAL*8 A,B,C,D,FUNCTN,GAUSS,SUM,W,Z
      DIMENSION Z(12),W(12)
C
C
C
      DATA Z/0.064056892862065,0.191118867473616,0.315042679696163,
     1        0.433793507626045,0.545421471388839,0.648093651936975,
     2        0.740124191578554,0.820001985973902,0.886415527004401,
     3        0.938274552002732,0.974728555971309,0.995187219997021/
C
      DATA W/0.127938195346752,0.125837456346828,0.121670472927803,
     1        0.115505668053725,0.107444270115965,0.097618652104113,
     2        0.086190161531953,0.073346481411080,0.059298584915436,
     3        0.044277438817419,0.028531388628933,0.012341229799987/
C
C
C.....SET UP INITIAL PARAMETERS
      C = (B-A)/2.0D00
      D = (B+A)/2.0D00
C
C.....ACCUMULATE THE SUM IN THE 24-POINT FORMULA
      SUM = 0.0
      DO 5  J = 1, 12
    IF(Z(J).EQ.0.0) SUM = SUM + W(J)*FUNCTN(D)
    IF(Z(J).NE.0.0) SUM = SUM + W(J)*(FUNCTN(Z(J)*C + D)
     1          + FUNCTN(-Z(J)*C + D))
    5 CONTINUE
C
C.....MAKE INTERVAL CORRECTION AND RETURN
      GAUSS = C*SUM
      RETURN
      END
```

```
      SUBROUTINE SOL2D(C,PEX,PEY,TSOL,N,NR)
C     NUMERICAL EVALUATION OF ANALYTICAL SOLUTION
C
C     REVISED: 18 APRIL 1984
C
      REAL LAMBDA
      COMMON/RATE/Q(10,12),T(10,12)
      COMMON/PHYPRO/ALPHA,BETA,DX,LAMBDA,PE,RD,V
C
      PEXY = SQRT(PEX**2 + BETA*(PEY**2))
      MT = NR + 1
      IF(MT.GT.1) GO TO 10
C
C     STEADY-STATE SOLUTION
C
      S = Q(N,1)
      B = 0.5*PEXY*ALPHA
      C = 2.0*LAMBDA*S*EXP(PEX/2.0 + BKOLOG(B))
      GO TO 50
C
C     TRANSIENT SOLUTION
   10 C = 0.0
      IF(T(N,MT).LT.TSOL) MT=MT+1
      DO 40  K=2,MT
         IF(T(N,K-1).GT.TSOL) GO TO 50
         S = Q(N,K) - Q(N,K-1)
         PINJ = V*V*(TSOL-T(N,K-1))/(DX*RD)
         TAU = PINJ/(PEXY*PEXY)
         U = 0.25/TAU
         B = 0.5*PEXY*ALPHA
         IF (B.GT.20.0) GO TO 20
         WF =   W(U,B)
         SUMLOG = PEX/2.0 + ALOG(WF)
         TERM = EXP(SUMLOG)
         GO TO 30
C
C
C       FOR LARGE VALUES OF B USE THE THIRD ORDER APPROXIMATION
C       OF WILSON AND MILLER (1979) JOUR. HYDRAULICS DIV., ASCE,
C       VOL 105, NO HY12, P 1565.
C
C       NOTE:  THE TERM EXP(PEX/2) IS INCLUDED IN THE NUMERICAL
C       APPROXIMATION FOR W(U,B) TO AVOID COMPUTATIONAL DIFFICULTIES
C       IN TAKING THE PRODUCT  EXP(PEX/2)*W(U,B)
C
C
   20          TERM = WELPRD(U,B,PEX)
   30          IF(TERM.LE.1.0E-32) TERM=0.0
               C = C + LAMBDA*S*TERM
   40 CONTINUE
   50 RETURN
      END
```

```
      FUNCTION W(U,B)
C
C     JAN WAGNER
C     SCHOOL OF CHEMICAL ENGINEERING
C     OKLAHOMA STATE UNIVERSITY
C     STILLWATER, OK  74078
C     TELEPHONE: (405) 624-5280
C
C     REVISED: 6 JANUARY 1983
C
C     EVALUATION OF THE WELL FUNCTION FOR LEAKY ARTESIAN AQUIFERS
C     THIS VERSION HANDLES ARGUMENTS OVER THE ENTIRE RANGE
      REAL*8 A1,A2,FUNCTN,GAUSS,DZ
      EXTERNAL FUNCTN
      COMMON BF
      BF=B
      IF(B.GT.0.1) GO TO 200
      IF(U.GT.1.0) GO TO 100
C
C     FOR B < 0.1  USE APRROXIMATIONS PRESENTED BY
C     HANTUSH, M.S. AND C.E. JACOB (1955)
C     TRANSACTIONS AMERICAN GEOPHYSICAL UNION,
C     VOL 36, NO. 1 PP. 95 - 100.
C
C
      IF(U.LE.1.0E-10) GO TO 50
C
C
C     EQUATION 12, FOR U < 1.0
C
C
      CON = B*B/(4.0*U)
      TERM1 = 2.0*BKO(B)
      TERM2 = EXP(BIOLOG(B) + E1LOG(CON))
      E1U = EXP(E1LOG(U))
      SUM = 0.57721566 + ALOG(U) + E1U + (U*B*B/16.0)*(1.0 - U/9.0)
      SUMLOG = ALOG(SUM)
      TERM3 = EXP(SUMLOG - CON)
      W = TERM1 - TERM2 + TERM3
      RETURN
   50 W = 2.0*BKO(B)
      RETURN
C
C     RECIPROCAL RELATION, EQUATION 17, FOR U > 1.0
C
C
  100 TERM1 = EXP(BIOLOG(B) + E1LOG(U))
      SUM = (B*B/4.0)*(1.0/U - 1.0/(36.0*U*U))
     1        + ((B*B/4.0)**2)*(1.0/(4.0*U) - 1.0/(4.0*U*U))
      SUMLOG = ALOG(SUM)
      TERM2 = EXP(SUMLOG - U)
      W = TERM1 - TERM2
      RETURN
C
C
  200 CONTINUE
C     FOR 0.1 < B < 20.0  USE NUMERICAL INTEGRATION
C     FOR U < B/2, W(U,B) = 2KO(B)-INT(0--U) FUNCTION
C     FOR U > B/2  W(U,B) = INT(0--B**2/4U) FUNCTION
C
C
      A1 = 0.0
      A2 = U
      B2 = B/2.0D00
      IF(U.GE.B2) A2=B2*B2/U
      DZ = GAUSS(A1,A2,FUNCTN)
      Z = DZ
      W = 2.0*BKO(B) - Z
      IF(U.GE.B2) W=Z
      RETURN
```

END WELLO71

```
      FUNCTION WELPRD(U,B,PEX)
C     JAN WAGNER
C     SCHOOL OF CHEMICAL ENGINEERING
C     OKLAHOMA STATE UNIVERSITY
C     STILLWATER, OK  74078
C     TELEPHONE: (405) 624-5280
C
C     REVISED: 6 JANUARY 1983
C
C     THIS FUNCTION SUBROUTINE EVALUATES EXP(PEX/2)*W(U,B)
C     USING THE THIRD-ORDER APPROXIMATION FOR W(U,B) PRESENTED
C     BY WILSON AND MILLER (1979) JOUR. HYDRAULICS DIV., ASCE,
C     VOL 105,NO HY12, P 1565.
C
      REAL*8 DI, FZ, TERMI, TERMO, SUM, Z
      COMMON/IO/NI,NO
C
      PAR = (B-2.0*U)/((4.0*U)**0.5)
      IF(ABS(PAR).GT.3.0) GO TO 50
      TERM1 = (1.0 - 1.0/(8.0*B))*ERFC(-PAR)
      TERM2 = (PAR/(7.0898154*B))/EXP(PAR**2)
C     W = ((1.5707963/B)**0.5)*EXP(-B)*(TERM1+TERM2)
C     WELPRD = EXP(PEX/2)*W(U,B)
C      = ((1.5707963/B)**0.5)*EXP(PEX/2 - B)*(TERM1+TERM2)
      SUMLOG = ALOG(TERM1 + TERM2)
      WELOG = 0.5*(0.45158271 - ALOG(B)) + (PEX/2.0 - B) + SUMLOG
      IF(WELOG.LT.-72.0) GO TO 20
      WELPRD = EXP(WELOG)
      RETURN
   20 CONTINUE
      WELPRD = 1.0E-32
      RETURN
   50 CONTINUE
      IF(PAR.LT.0.0) GO TO 100
C     FOR B>20 AND PAR>3.0 W(U,B)=2K0(B)
      WELOG = PEX/2.0 + BKOLOG(B) + 0.69314718
      WELPRD = EXP(WELOG)
      RETURN
  100 CONTINUE
C     FOR PAR<-3.0 AN ASYMPTOTIC EXPANSION FOR ERFC(-PAR) IS UTILIZED
C     SEE SECTION 7.1 OF ABRAMOWITZ AND STEGUN (1966)
      COEFLG = PEX/2.0 - B - PAR*PAR - ALOG(-PAR) - 0.5*ALOG(2.0*B)
     1         + ALOG(1.0 - 0.1250/B)
      IF(COEFLG.LT.-72.0) GO TO 200
      Z = -PAR
      FZ = 2.0D00*Z*Z
      SUM = 1.0D00
      TERMO = 1.0D00
      DO 120  I=2,50
          DI = I
          TERMI = -TERMO*(2.0D00*DI-3.0D00)/FZ
          IF(DABS(TERMI).GT.DABS(TERMO)) GO TO 150
          SUM = SUM + TERMI
          TEST = TERMI/SUM
          IF(ABS(TEST).LT.1.0E-16) GO TO 150
          TERMO = TERMI
  120 CONTINUE
      WRITE(NO,500)
  500 FORMAT(6X,'*** WARNING -- ASYMPTOTIC APPROXIMATION FOR ERFC IN',/,
     1       6X,'                FUNCTION WELPRD DID NOT CONVERGE WITH ',/
     2       6X,'                50 TERMS IN THE SUMMATION')
  150 SUMLOG = DLOG(SUM)
      WELOG = COEFLG + SUMLOG
      WELPRD = EXP(WELOG)
      RETURN
  200 CONTINUE
C     FOR LARGE NEGATIVE VALUES OF PAR, ERFC(-PAR) -> 2
      WELPRD = 0.0
      RETURN
      END
```