THEORY AND IMPLEMENTATION FOR SWIFT FOR WINDOWS THE SANDIA WASTE-ISOLATION FLOW AND TRANSPORT MODEL FOR FRACTURED MEDIA

Prepared by:

HSI GeoTrans
46050 Manekin Plaza, Suite 100
Sterling, Virginia 20166

ABSTRACT

This report is one of two that describe the three-dimensional finite-difference code SWIFT for Windows. This code is used to simulate flow and transport processes in geologic media that may be fractured and was developed for use in the analysis of deep geologic nuclear waste-disposal facilities. However. it may be used in other areas such as waste injection into saline aquifers and heat storage in aquifers. Both dual-porosity and discrete-fracture conceptualizations may be considered for the fractured zones. In order to maximize computer efficiency, only one-dimensional migration is permitted within the rock matrix. Otherwise, the treatment of the matrix is general and entirely consistent with the treatment of the fracture and single-porosity zones. A variable density is included throughout, and a variety of options are available to facilitate the various uses of the code. This document presents the theory and formulation of the SWIFT for Windows code. It is intended to be a comprehensive statement of the mathematical, numerical, and computer implementation of the code. A companion document, Data Input Guide for SWIFT for Windows., gives the input data guide.

TABLE OF CONTENTS

Page

TABLE OF CONTENTS
(continued)

Page

TABLE OF CONTENTS
(continued)

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

1 INTRODUCTION

1.1 BRIEF DESCRIPTION OF THE SWIFT FOR WINDOWS MODEL

SWIFT for Windows (Sandia Waste Isolation, Flow and Transport Code) is a fully transient, three-dimensional code that solves the coupled equations for transport in geologic media. The processes considered are:

- fluid flow
- heat transport
- dominant-species miscible displacement (referred to as "brine")
- trace-species miscible displacement (referred to as "radionuclides")

The first three processes are coupled via fluid density and viscosity. Together they provide the velocity field required in the third and fourth processes. This document describes the extensions of the capabilities of SWIFT (Reeves and Cranwell, 1981) and SWIFT II (Reeves, et al., 1984) to include fractured media.

1.2 APPLICATIONS OF SWIFT FOR WINDOWS

Because the SWIFT for Windows code is general, it has many possible applications. They include, but are not limited to, the following:

- nuclear waste isolation in both fractured and porous geologic media
- injection of industrial wastes into saline aquifers
- heat storage in aquifers
- in-situ solution mining
- migration of contaminants from landfills
- disposal of municipal wastes
- salt-water intrusion in coastal regions

- brine disposal from petroleum-storage facilities
- brine disposal as a byproduct of methane production from geopressured aquifers
- determination of aquifer transport parameters from well-test data

1.3 PURPOSE OF THIS DOCUMENT

Having evolved from the U.S. Geological Survey Waste Injection Program, SWIP, the SWIFT code has been continuously developed and maintained since 1975. The original documentation for SWIP was presented in INTERCOMP (1976). This was followed by Dillon, Lantz, and Pahwa (1978), by Reeves and Cranwell (1981) and Reeves, et al., 1984.

The objective for writing this report is twofold. First, the facility for treating fractured media requires elaboration. This feature of SWIFT for Windows is interwoven into the entire treatment. Second, a comprehensive statement of the theory and implementation of SWIFT for Windows is necessary. Thus, a complete statement of the finite-difference implementation is included here for the first time since the INTERCOMP document. Other items, such as the wellbore sub-model, aquifer-influence functions, and the matrix solution algorithms are given here. These were either treated superficially or excluded from earlier reports. The intent here is to produce a stand-alone description of the theoretical, numerical, and computer implementations which may be used without reference to SWIFT or SWIP documentation issued previously.

Two other documents for SWIFT II have been published, namely an Verification and Field Comparison for SWIFT (Ward, et al., 1984) and a Self-Teaching Curriculum (Reeves, et al., 1986b).

2 MATHEMATICAL IMPLEMENTATION

The SWIFT for Windows code is designed to simulate flow and transport processes in both fractured and porous media. The analyst designates the fractured regions of the system to which the dual-porosity approach is to be applied. In those particular regions, two sets of equations are solved, one for processes in the fractures and the other for processes in the matrix. The fracture-porosity equations describing flow and transport for the fractured. regions are identical to the equations for the porous zone, except for sink/source terms representing exchange processes with the matrix. Consequently, one general set of equations that applies to both zones is presented, which will be called the "global" set of equations. The matrix-porosity equations for the fractured zone differ somewhat from their global counterparts. Therefore, a separate set of equations is presented that will be called the "local" set of equations. As was mentioned in the introduction, a variable-density formulation is used throughout. Density, viscosity, porosity, and enthalpy are treated as functions of pressure, temperature, and brine concentration, but not radionuclide concentrations. For this reason, the flow, heat, and brine equations are termed the primary equations.

A steady-state solution option is provided for the global primary equations with two qualifications. First, it is assumed that heat-transport is basically a transient process. Certainly, this is true for high-level nuclear waste repositories, a dominant application for the code. Thus, heat transport, like radionuclide transport, is not included in the steady-state option. Secondly, it is assumed that matrix processes are negligible at steady state. Consequently, the state equations for the matrix porosity are not solved for the steady-state option. Of course, the code will permit transient solution of radionuclide transport (with or without dual porosity) in conjunction with steady-state solution of the primary equations since this is perceived as a very useful simulation capability.

In this chapter, the order of presentation is that of global transient equations followed by global steady-state equations followed by local transient equations.

2.1 THE GLOBAL TRANSIENT-STATE EQUATIONS FOR FLOW, HEAT, BRINE, AND RADIONUCLIDE TRANSPORT

The transport equations are obtained by combining the appropriate continuity and constitutive relations and have been presented by several authors, including Cooper (1966), Reddell and Sunada (1970), Bear (1979), and Aziz and Settari (1979). Sink/source terms Γ are included for fractured zones in which exchange with the rock matrix are significant. The resulting relations may be stated as follows:[1]

Fluid:

$$\underbrace{-\Delta\cdot(\rho\underline{u})}_{\text{convection}} - \underbrace{q}_{\text{production}} - \underbrace{q_W}_{\text{Sink / source}^2} + \underbrace{R'_C}_{\text{salt dissolution}} - \underbrace{\Gamma_w}_{\text{exchange with matrix}} = \underbrace{\frac{\partial}{\partial t}(\phi\rho)}_{\text{accumulation}} \qquad (2\text{-}1)$$

Heat:

$$\underbrace{-\Delta\cdot(\rho H\underline{u})}_{\text{convection}} + \underbrace{\Delta\cdot(\underline{\underline{E}}_H\cdot\Delta T)}_{\text{conduction / dispersion}} - \underbrace{H_I q}_{\text{injected enthalpy}^3} - \underbrace{Hq}_{\text{produced enthalpy}} - \underbrace{q_H}_{\text{sink / source}} - \underbrace{(H\Gamma_W+\Gamma_H)}_{\text{exchange with matrix}^4} = \underbrace{\frac{\partial}{\partial t}\left[\phi\rho U+(1-\phi)\rho_R U_R\right]}_{\text{accumulation in fluid and rock}} \qquad (2\text{-}2)$$

---

[1]All terms are defined in the notation section.

[2]This term refers to a sink/source other than a well. A positive sign denotes a sink, and a negative sign denotes a source.

[3]This is a source term since, by the adopted sign convention. The rate of fluid injection is inherently negative.

[4]Exchange incorporates losses to matrix (+) or gains from matrix (-).

Brine:

$$\underset{\text{convection}}{-\nabla\cdot\left(\rho\hat{C}u\right)} + \underset{\substack{\text{dispersion /}\\ \text{diffusion}}}{\nabla\cdot\left(\rho E_C\cdot\nabla\hat{C}\right)} - \underset{\substack{\text{injected}\\ \text{brine}}}{\hat{C}_I q} - \underset{\substack{\text{produced}\\ \text{brine}}}{\hat{C}q}$$

$$+ \underset{\substack{\text{salt}\\ \text{dissolution}}}{R_C} - \underset{\substack{\text{exchange with}\\ \text{matrix}}}{\left(\hat{C}\Gamma_W + \Gamma_C\right)} = \underset{\text{accumulation}}{\frac{\partial}{\partial t}\left(\phi\rho\hat{C}\right)} \qquad (2\text{-}3)$$

Radionuclide r:

$$\underset{\text{convection}}{-\nabla\cdot\left(\rho C_r u\right)} + \underset{\substack{\text{dispersion /}\\ \text{diffusion}}}{\nabla\cdot\left(\rho E_C\cdot\nabla C_r\right)} - \underset{\substack{\text{produced}\\ \text{component}}}{C_r q} - \underset{\substack{\text{sink /}\\ \text{source}}}{q_r} + \underset{\substack{\text{waste}\\ \text{leach}}}{q_{wr}}$$

$$\underset{\substack{\text{exchange with}\\ \text{matrix}}}{-\left(C_r\Gamma_W + \Gamma_r\right)} + \underset{\substack{\text{generation of component}\\ \text{r by decay of s}}}{\sum_{s=1}^{N} k_{rs}\lambda_s\left[\phi\rho C_s + \left(1-\phi\right)\rho_R W_s\right]}$$

$$\underset{\substack{\text{decay of}\\ \text{component}}}{-\lambda_r\left[\phi\rho C_r + \left(1-\phi\right)\rho_R W_r\right]} = \underset{\text{accumulation}}{\frac{\partial}{\partial t}\left[\phi\rho C_r + \left(1-\phi\right)\rho_R W_r\right]} \qquad (2\text{-}4)$$

Several quantities in Equations (2-1) through (2-4) require further definition in terms of the basic parameters. The tensors in Equations (2-2), (2-3), and (2-4) are defined as sums of dispersion and molecular terms:

$$\underline{\underline{E}}_C = \underline{\underline{D}} + D_m\underline{\underline{I}} \qquad (2\text{-}5)$$

and

$$\underline{\underline{E}}_H = \underline{\underline{D}}\rho c_p + \underline{\underline{K}}_m \qquad (2\text{-}6)$$

where

$$D_{ij} = \alpha_T u\delta_{ij} + \left(\alpha_L - \alpha_T\right)u_i u_j / u \qquad (2\text{-}7)$$

in a Cartesian system. Also, sorption of radionuclides is assumed to be given by a nonlinear Freundlich equilibrium isotherm:

$$W_r = \kappa_r \left(\rho C_r\right)^{\eta_r} \tag{2-8}$$

Equations (2-1) through (2-4) are coupled by auxiliary relations for:

Darcy flux:

$$\underline{u} = -\left(\underline{\underline{k}} / \mu\right) \cdot \left(\nabla p \; - \; \frac{\rho q}{g_c} \nabla z \right) \tag{2-9}$$

porosity:

$$\phi = \phi_o \left[1 + c_R \left(p - p_o\right)\right] \tag{2-10}$$

fluid density:

$$\rho = \rho_o \left[1 + c_W \left(p - p_o\right) - c_T \left(T - T_o\right) + c_C \hat{C}\right] \tag{2-11}$$

fluid viscosity:

$$\mu = \mu_R \left(\hat{C}\right) \exp\left[B\left(\hat{C}\right)\left(T^{-1} - T_R^{-1}\right)\right] \tag{2-12}$$

fluid enthalpy:

$$H = U_o + U + p / \rho \tag{2-13}$$

fluid internal energy:

$$U = c_p \left(T - T_o\right) \tag{2-14}$$

rock internal energy:

$$U_R = c_{pR} \left(T - T_o\right) \tag{2-15}$$

where parameter c_C in Equation (2-11) is defined in terms of an input density range (ρ_I-ρ_N) and the reference density ρ_o:

$$c_C = (\rho_I - \rho_N) / \rho_o \tag{2-16}$$

Furthermore, an internal energy U_o is included in Equation (2-13) to account for the difference in reference conditions as specified by the analyst and the reference conditions specified internally for the enthalpy.

2.2 THE GLOBAL STEADY-STATE EQUATIONS FOR FLOW AND BRINE TRANSPORT

In safety evaluations for nuclear-waste repositories, quite often the time frame of interest may extend over many thousands of years. Typically, the assumption of time invariant flow and brine conditions is justified in such cases due to the lack of specific data for such a long period of time. For the fluid flow, the overall effect of transient rainfall boundary conditions may have a minor effect on radionuclide transport. Duguid and Reeves (1976) have shown this to be the case for a combined saturated-unsaturated simulation of tritium transport averaged over a period of only one month. For the brine transport, transient effects in deep formations will be negligible also. For the heat transport, however, the radionuclides stored within a repository will provide a transient heat source for thousands of years. Thus, heat transport is considered here to be a transient process and is not included in the steady-state option.

Two steady-state options have been included. The first option permits solution of the time-independent flow equation:

Fluid (steady-state):

$$\underset{\text{convection}}{-\nabla\cdot(\rho\underline{u})} \; \underset{\text{production}}{-\; q} \; \underset{\text{source}}{-\; q_W} \; \underset{\text{dissolution}}{+\; R'_C} \; = 0 \tag{2-17}$$

In both options the accumulation and the matrix-exchange term are set to zero, as shown in Equation (2-17). The latter is neglected because the local set of equations for the matrix are not solved at a steady state. For the steady-state fluid-flow option. however, the salt

dissolution term is also set to zero, and the effects of brine and heat are included through the variable density and the variable viscosity.

The second option permits a coupled time-independent solution for both fluid flow, Equation (2-17), and brine transport:

Brine (steady-state):

$$\underset{\text{convection}}{-\nabla\cdot\left(\rho\hat{C}\underline{u}\right)} + \underset{\substack{\text{dispersion /}\\ \text{diffusion}}}{\nabla\cdot\left(\rho\underline{\underline{E}}_C\cdot\nabla\hat{C}\right)} - \underset{\substack{\text{injected}\\ \text{brine}}}{\hat{C}_I q} - \underset{\substack{\text{produced}\\ \text{brine}}}{\hat{C}q} + \underset{\substack{\text{salt}\\ \text{dissolution}}}{R_C} = 0 \qquad (2\text{-}18)$$

In this case, in addition to a variable density and a variable viscosity, the salt-dissolution term is generally nonzero.

2.3 THE LOCAL TRANSIENT-STATE EQUATIONS FOR FLOW, HEAT, BRINE, AND RADIONUCLIDE TRANSPORT WITHIN THE ROCK MATRIX

The flow and transport processes occurring within the rock matrix are conceptualized as being one-dimensional in a lateral direction relative to the movement in the fractures. Thus, it is assumed that the fractures provide the only means for large-scale movements through the entire system while the matrix provides most of the storage of the system. The approach used here to treat the fracture-matrix system is similar to that used by Bear and Braester (1972), Huyakorn et al. (1983), Pruess and Narasimhan (1982), Tang et al. (1981), Grisak and Pickens (1980), Streltsova-Adams (1978), and Rasmuson et al. (1982).

Conservation equations used here for the matrix are very similar to those presented in Section 2.1. They are as follows:

Fluid (matrix):

$$\underset{\text{convection}}{-\nabla\cdot\left(\rho'\underline{u}'\right)} + \underset{\substack{\text{gain from}\\ \text{fracture}}}{\Gamma'_W} = \underset{\text{accumulation}}{\frac{\partial}{\partial t}\left(\phi'\rho'\right)} \tag{2-19}$$

Heat (matrix):

$$\underset{\text{convection}}{-\nabla\cdot\left(\rho' H'\underline{u}\right)} + \underset{\substack{\text{conduction /}\\ \text{dispersion}}}{\nabla\cdot\left(E'_H \nabla T'\right)} + \underset{\substack{\text{exchange with}\\ \text{fracture}}}{\left(H'\Gamma'_W+\Gamma'_H\right)} = \underset{\substack{\text{accumulation in fluid}\\ \text{and rock}}}{\frac{\partial}{\partial t}\left[\phi'\rho' U'+(1-\phi)\rho_R U'_R\right]} \tag{2-20}$$

Brine (matrix):

$$\underset{\text{convection}}{-\nabla\cdot\left(\rho'\hat{C}'\underline{u}\right)} + \underset{\substack{\text{dispersion /}\\ \text{diffusion}}}{\nabla\cdot\left(\rho' E'_C \nabla\hat{C}'\right)} + \underset{\substack{\text{exchange with}\\ \text{fracture}}}{\left(\hat{C}'\Gamma'_W+\Gamma'_C\right)} = \underset{\text{accumulation}}{\frac{\partial}{\partial t}\left(\phi'\rho'\hat{C}'\right)} \tag{2-21}$$

Radionuclide r (matrix):

$$\underset{\text{convection}}{-\nabla\cdot\left(\rho' C'_r\,\underline{u}'\right)} + \underset{\substack{\text{dispersion /}\\ \text{diffusion}}}{\nabla\cdot\left(\rho' E'_C \nabla C'_r\right)} + \underset{\substack{\text{generation of}\\ \text{component}\\ \text{r by decay of s}}}{\sum_s k_{rs}\lambda_s K'_S \phi'\rho' C'_s}$$

$$\underset{\substack{\text{decay of}\\ \text{component r}}}{-\lambda_r K'_r \phi'\rho' C'_r} + \underset{\substack{\text{exchange with}\\ \text{fracture}}}{\left(C'_r\Gamma'_W+\Gamma'_r\right)} = \underset{\text{accumulation}}{\frac{\partial}{\partial t}\left(K'_r \phi'\rho' C'_r\right)} \tag{2-22}$$

Both convection and dispersion terms are retained in Equations (2-19) through (2-22). These terms arise only through fluid-density changes and likely will be negligible except for highly pressurized and/or highly heated regions. It is anticipated that either parallel fractures (idealized in Figure 2-1a) or intersecting sets of parallel fractures (idealized in Figure 2-1b) will be treated. A prismatic block is invoked in the numerical solution, and for the latter, either prismatic or spherical blocks may be used to approximate the actual matrix geometry. Thus, either one-dimensional Cartesian or spherical geometry may be used for the local matrix equations. In either case, the interior boundary is assumed to be a reflective no-flow boundary. The fracture/matrix interface provides a source Γ', which is identical to the fracture loss Γ to within a geometrical scaling factor.

Many of the coefficients of Equations (2-19) to (2-22) require further specification. The coefficients of the second-order transport terms are defined as follows:

$$E'_C = D' + D'_m \tag{2-23}$$

$$E'_H = D'\rho' c_p + K'_m \tag{2-24}$$

$$D' = \alpha'_L \, u' \tag{2-25}$$

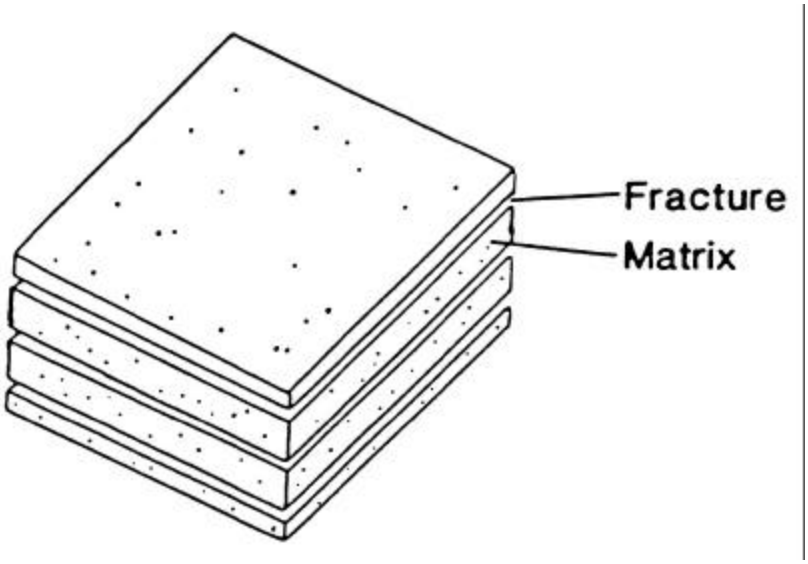
For the rock matrix, diffusion is expected to dominate the dispersion E'_C, in contrast to the dispersion E_C for the global simulation. Consequently, the dependence of diffusion upon temperature is expected to be much more significant and is included through the linear relation:

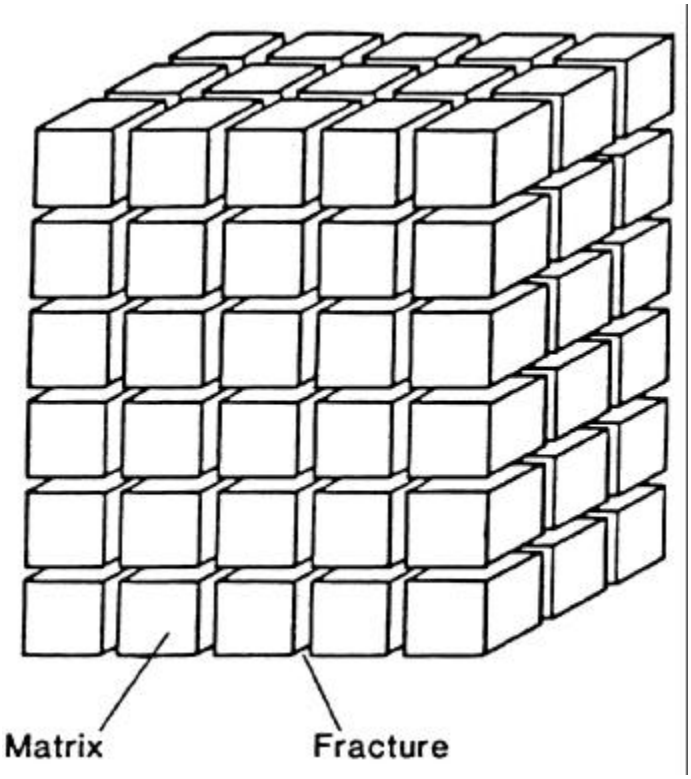
$$D'_m = D'_{mo}\left[1 + \delta'\left(T' - T_o\right)\right] \tag{2-26}$$

Sorption of radionuclides within the matrix is included via the assumption of a linear equilibrium isotherm:

$$K'_r = 1 + \rho_R k'_{dr}\left(1 - \phi'\right)/\phi' \tag{2-27}$$

Figure 2.1. Idealization of Fracture Sets.





Equations (2-19) through (2-22) are coupled by four auxiliary relations for:

Darcy flux (matrix):

$$\underline{u}' = -\left(k'/\mu'\right)\nabla p' \tag{2-28}$$

Porosity (matrix):

$$\phi' = \phi'_o\left[1 + c'_R\left(p' - p_o\right)\right] \tag{2-29}$$

Fluid density (matrix):

$$\rho' = \rho_o\left[1 + c_W\left(p' - p_o\right) - c_T\left(T' - T_o\right) + c_C\hat{C}'\right] \tag{2-30}$$

Fluid viscosity (matrix):

$$\mu = \mu_R\left(\hat{C}'\right)\exp\left[B\left(\hat{C}'\right)\left(T'^{-1} - T_R^{-1}\right)\right] \tag{2-31}$$

Fluid enthalpy (matrix):

$$H' = U_o + U' + p'/\rho' \tag{2-32}$$

Fluid internal energy (matrix):

$$U' = c_p\left(T' - T_o\right) \tag{2-33}$$

Rock internal energy (matrix):

$$U'_R = c'_{pr}\left(T' - T_o\right) \tag{2-34}$$

Parameter c_C is defined by Equation (2-16), and it is assumed in Equation (2-28) that gradients of the elevation head are of negligible importance in determining Darcy velocities within the matrix.

3 SYSTEM DESCRIPTION

Several equations, or submodels, are given in Chapter 2 with which hydrological, thermal, and geochemical properties of the equation may be specified. For example, fracture porosity and matrix porosity are related to compressibility in Equations (2-10) and (2-29), respectively. In addition, the nonlinear Freundlich isotherm is presented in Equation (2-8). In this chapter additional discussion is given for four such submodels. In two cases, namely, density and viscosity, the purpose is to provide additional insight regarding the choice of parameter values. In the other two cases, salt-dissolution and waste leach, the intent is to present the model structure more precisely.

3.1 DENSITY

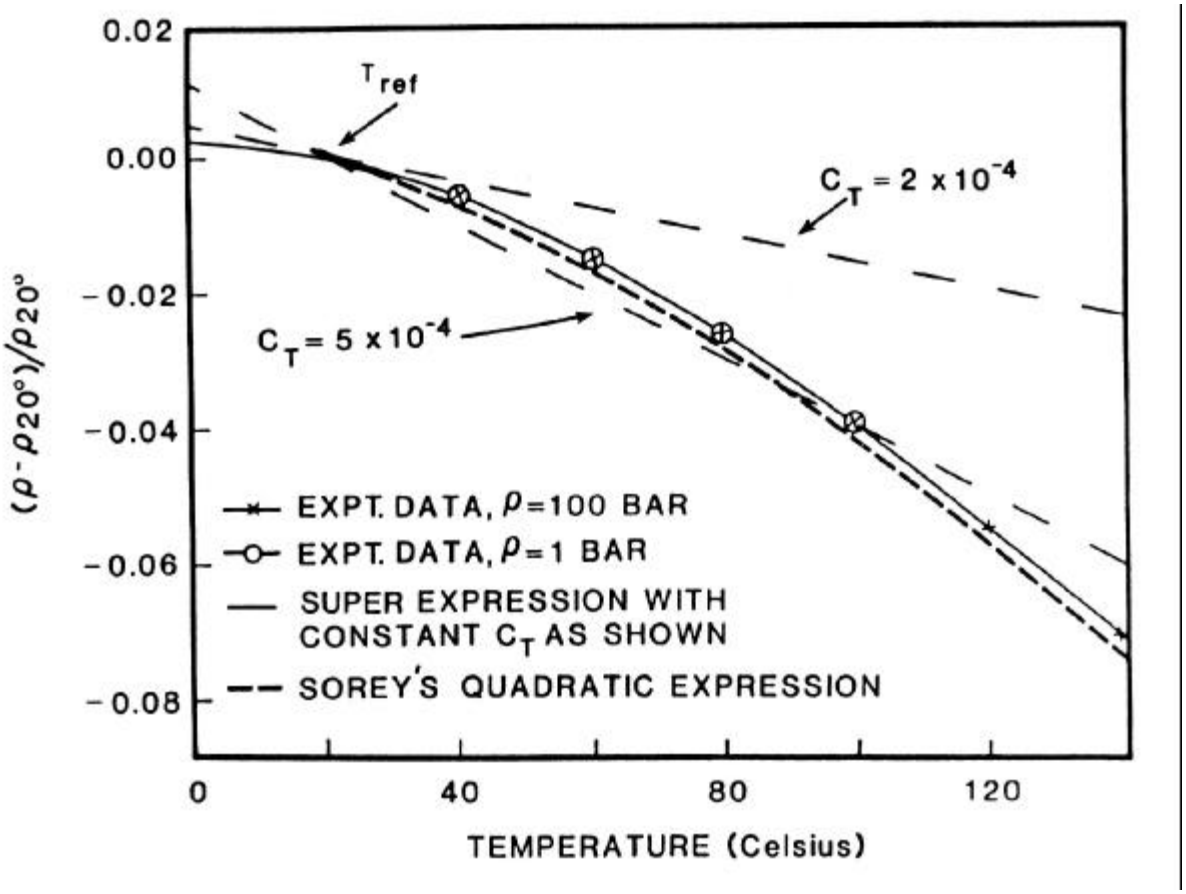
One of the unique features of the SWIFT for Windows code is that it permits variable densities and viscosities. The submodel for density is given in Equation (2-11):

$$\rho = \rho_o\left[1 + c_W\left(p - p_o\right) - c_T\left(T - T_o\right) + c_C\hat{C}\right] \tag{3-1}$$

This relation has been investigated by Muller, Finley, and Pearson (1981). They conclude that the use of constant values for c_W and c_C is adequate for most simulations but that the constant c_T must be carefully chosen for the expected temperature range. Since the variation of c_W is only 0.5 percent for a pressure change of 100 bars, the use of a constant value of c_W is adequate for most hydrological simulations. The experimentally observed variation of density with concentration for the brine coefficient c_C is not linear for four different salts. However, using a linear relation introduces a maximum error of two percent in the density. Such an error would likely be acceptable relative to other uncertainties in most simulations.

Parameter c_T is somewhat more variable than c_W and c_C. To show this, Muller et al. (1981) compare experimental data from Kennedy and Holser (1966) with the linear model of SWIFT for Windows and with a density model of Sorey (1978), which contains a quadratic term in temperature. Figure 3-1 shows density variations from a reference value at 20°C. As shown, Sorey's quadratic model compares quite favorably with the data. The linear model

Figure 3.1. Graph of Density Variations as a Function of Temperature. The SWIFT for Windows linear model may be compared with Sorey's quadratic model and with experimental data for pure water form Kennedy and Holser (1966).



shows a sensitive dependence to the value chosen for c_T. If this value is taken from the slope of the data at 20°C, the predicted densities can vary by as much as eight percent at elevated temperatures. However, if the value for c_T is chosen at 60°C, the variation is one percent or less.

INTERCOMP (1976) examines parameters c_C and c_T as coupled functions of brine concentration and temperature with similar results. The conclusion is that a linear model is adequate for parameters c_W and c_C, but that c_T must be based on the expected temperature range.

3.2 VISCOSITY

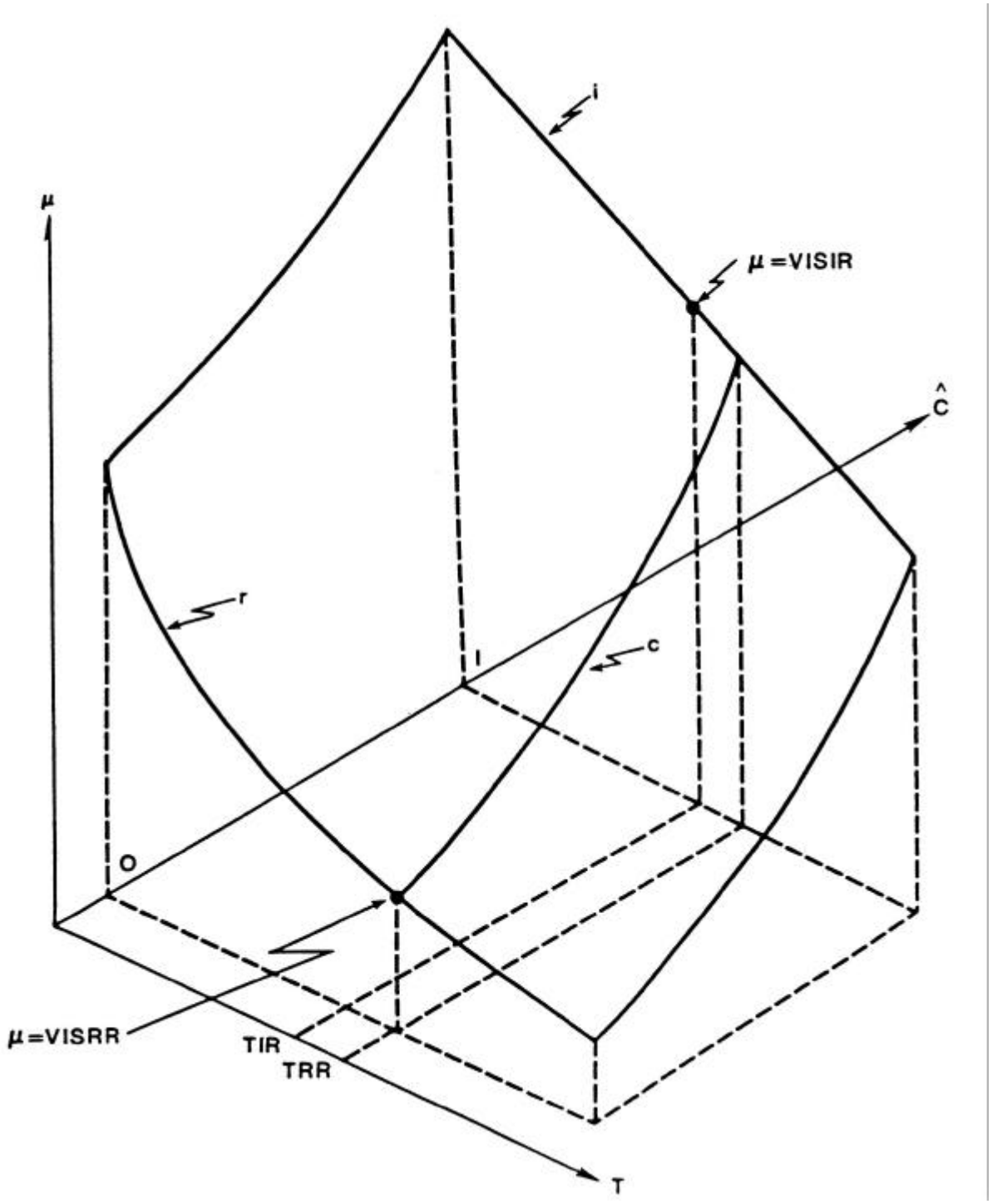
The SWIFT for Windows code uses Equation (2-12) to express the dependence of viscosity on concentration and temperature, namely

$$\mu = \mu_R\left(\hat{C}\right)\exp\left[B\left(\hat{C}\right)\left(T^{-1} - T_R^{-1}\right)\right] \tag{3-2}$$

Quantities μ_R and B are empirical functions which are determined from data. The relation in Equation (3-2) is shown schematically in Figure 3-2. Data of varying amounts, depending on availability, are specified along Curves r, i, and c and are used to determine the empirical relations $\mu_R(C)$ and B(C). Curves r and i are given by the extreme concentration values and are given as $\mu(0,T)$ and $\mu(1,T)$, respectively. Curve c, on the other hand, is defined by an intermediate value of temperature called the reference temperature T_R and is given by $\mu(C,T_R)$. From the data given for Curves r and i, the following four values of the empirical constants are determined: μ_R (0), μ_R (1), $B_r = B(\phi)$, and $B_i = B(l)$. From the latter two, function B(C) is determined by linear interpolation. From the former two, and from such additional data given along Curve c as may be available, function $\mu_R(C)$ is determined either by linear interpolation or by power-law interpolation, depending on the amount of additional data. In summary, Equation (3-2) and the above-mentioned techniques for determining empirical functions $\mu_R(C)$ and B(C) constitute the viscosity submodel of the SWIFT for Windows code.

The minimum data needed are the two points $\mu(\phi,TRR)$ = VISRR and $\mu(1,TIR)$ = VISIR (Reeves. et al., 1986a, p. 25). Additional data from the general curve of Lewis and

Figure 3.2. Fluid Viscosity as a Function of Temperature and Concentration.



Squires (1934) is supplied internally by the code. Additionally, one may specify other values for C = 0 and C = 1, (i.e., for Curves r and i in Figure 3-2). Or, in the best case, one may also supply mixture data for reference temperature T = TRR.

Muller et al. (1981) compare SWIFT for Windows viscosity model calculations in two of the above cases with experimental viscosity-temperature data for pure water. Some of their results are shown graphically in Figure 3-3, where the experimental data of various researchers may be compared with results from the SWIFT for Windows viscosity model. For the curve labeled "general curve," only one data point is supplied. The SWIFT for Windows viscosity model supplemented this single data point with the general curve of Lewis and Squires (1934) and performed a least-squares fit to obtain the parameters $\mu_R(0)$ and B(0) of Equation (3-2). SWIFT for Windows viscosity model results compare poorly with the experimental data at the two extremes in temperature. There is a 25 percent deviation of the SWIFT for Windows viscosity model results from experimental data.

For the curve labeled "exact fit" in Figure 3-3, only two data points are supplied: viscosities at the reference temperature of 20°C and at 60°C. The SWIFT for Windows viscosity model then performs an exact fit to obtain $\mu_R(0)$ and B(0). The comparison with experimental data here is quite satisfactory for the temperature range 0 to 120°C. Of course, additional viscosity-temperature values could be used to improve the fit further.

In INTERCOMP (1976), this viscosity model is compared with experimental viscosity-concentration data. Varying amounts of viscosity- temperature and viscosity-concentration (at 59°F) data are specified for the determination of the empirical functions $\mu_R(C)$ and B(C). In each case, the viscosity model is then used to calculate a viscosity concentration curve at 150°F. Results are displayed in Figure 3-4; these observations are similar to those of Muller et al. (1981). In Case III, only two data points are provided, and the viscosity model uses the data of Lewis and Squires (1934). The agreement between model and experimental results is relatively poor with a maximum deviation of about 18 percent. On the other hand, in Cases I and II, where additional data are specified, calculated results are acceptable, showing a maximum deviation of 5 percent from the experimental data.

Figure 3.3. Graph Comparing Viscosities Measured at Various Temperatures With Those Calculated by the SWIFT for Windows Viscosity Model.

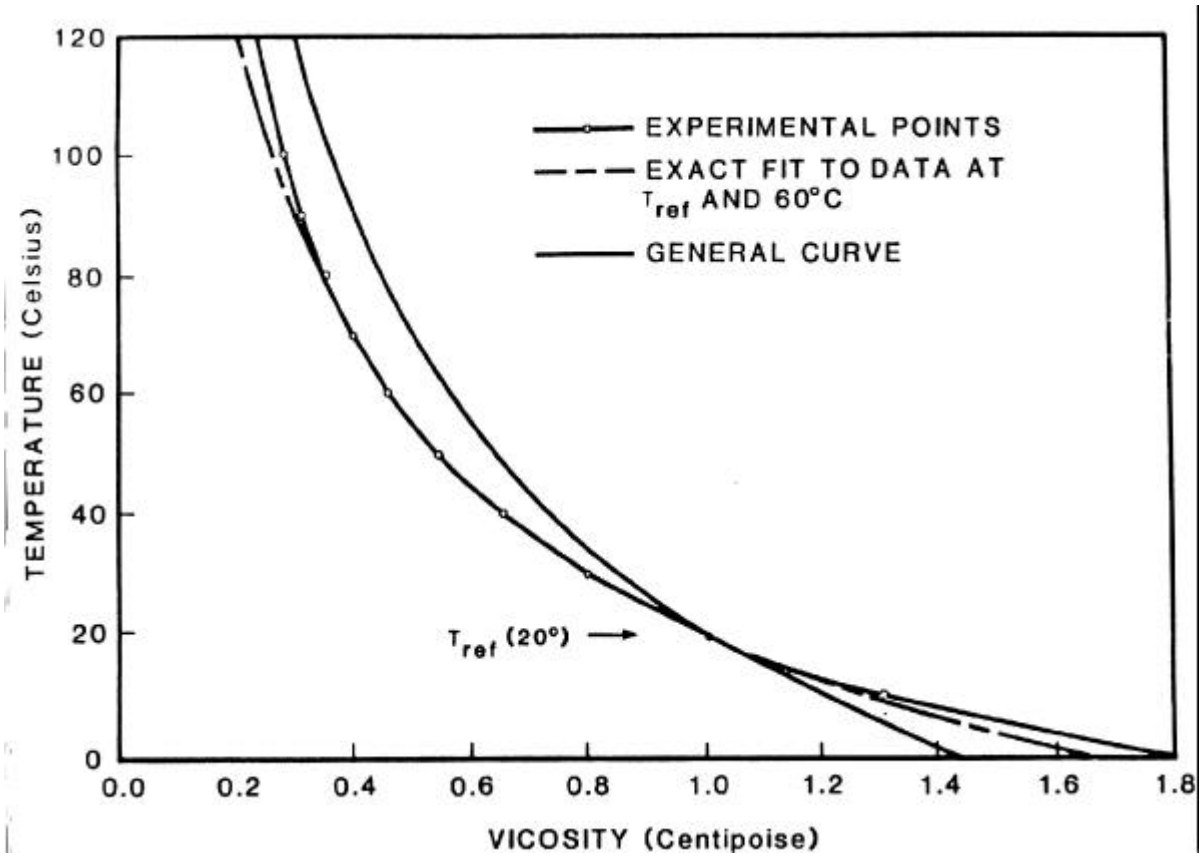
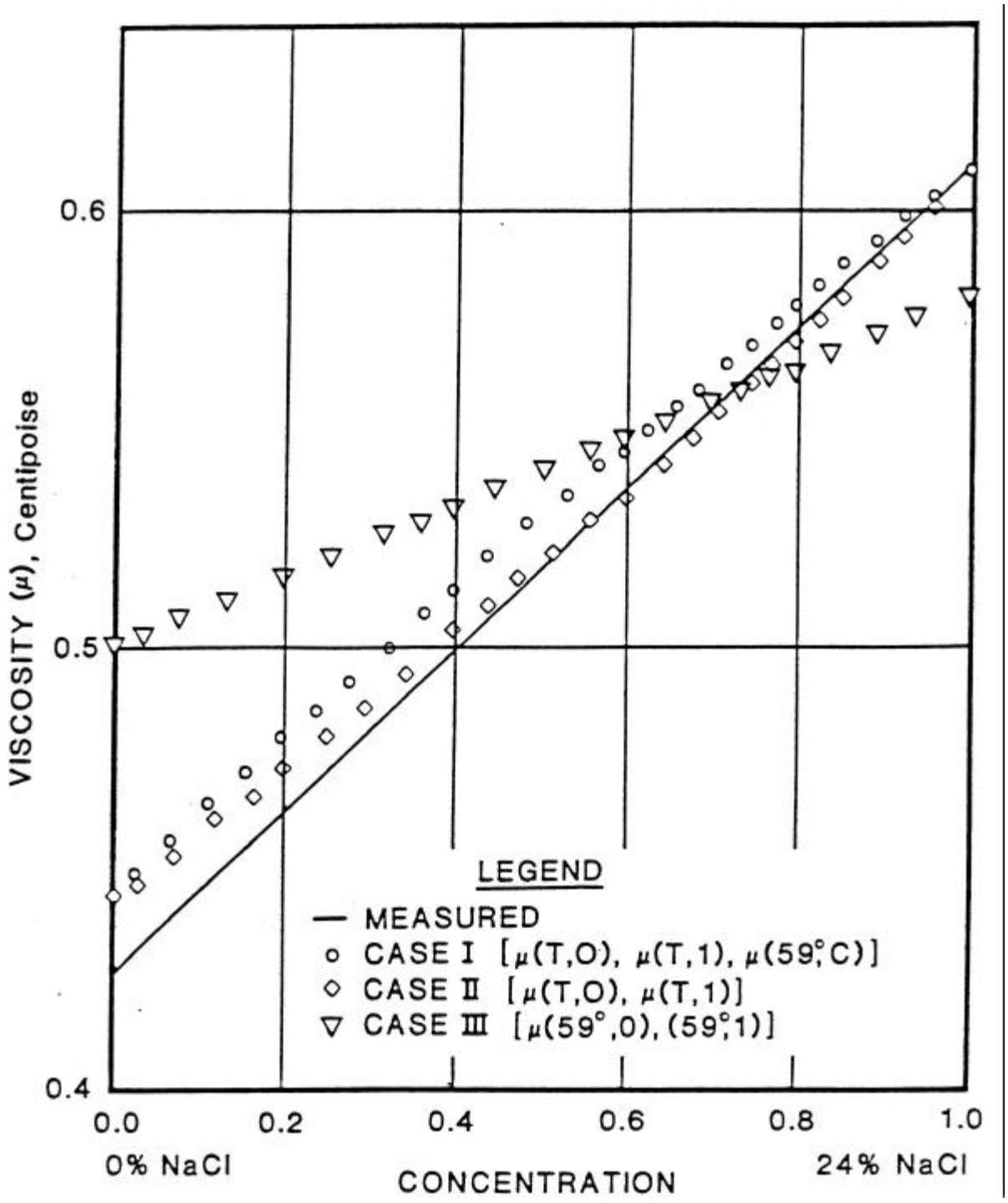


Figure 3.4. Predicted Brine Viscosity at 150°F with Three Levels of Available Data (T,C).



3.3 SALT DISSOLUTION

The salt dissolution mechanism appears in the fluid flow Equations (2-1) and (2-17) through the source term R'_C and in the brine transport, Equations (2-3) and (2-18), through the source term R_C. These quantities are defined by

$$R_C = \phi \rho k_C f_C \left(1 - \hat{C}\right) \tag{3-3}$$

and

$$R'_C = c_C R_C / \left(1 + c_C\right) \tag{3-4}$$

where coefficient c_C is defined by Equation (2-16). It should be noted that two parameters are used to characterize the dissolution process, namely, a rate constant k_C and a mass fraction f_C of solubles to total solid mass. This formulation of dissolution is similar to that of Nolen et al. (1974), where salt-cavern formation for storage of crude oil was considered.

3.4 WASTE LEACH

Another submodel which is present is that for waste leach. The purpose of this model is to determine the source rate q_{wr} (see Equation (2-4)) at which a radionuclide r from a repository is dissolved into solution. More specifically, this model considers each radioactive component to be in one of the following distinct phases:

1. unleached from the waste matrix,
2. leached but undissolved, or
3. dissolved.

Phases (1) and (2) are coupled by the leach rate. There is, of course, considerable uncertainty in the time dependence of this rate. However, in accord with arguments given by Campbell et al. (1978), we have chosen a constant leach rate for the implementation. Phases (2) and (3) above are coupled by the solubilities. The source rate q_{wr} for dissolving Phase (2) is kept small enough that the solubility of any given nuclide will not be exceeded. Radioactive decay and production processes are considered throughout the analysis.

As mentioned above, there are three phases to be tracked; specifically, the unleached radionuclides within the waste matrix m_r, the undissolved nuclides no longer in the waste matrix S_r, and the dissolved leachate C_r. The appropriate conservation equations may be written in the following manner. For the dissolved phase of component r:

$$-\nabla\cdot\left(\rho C_r \underline{u}\right) - q_{wr} + \sum_s k_{rs}\lambda_s K_s \phi \rho C_s - \lambda_r K_r \phi \rho C_r = \frac{\partial}{\partial t}\left(\phi \rho K_r C_r\right) \tag{3-5}$$

for the undissolved, but leached, component r:

$$-R_{wr} + q_{wr} + \sum_s k_{rs}\lambda_s S_s - \lambda_r S_r = \frac{dS_r}{dt} \tag{3-6}$$

and for the unleached component r:

$$\sum_S k_{rs}\lambda_w m_s - \lambda_s m_r = \frac{dm_r}{dt} \tag{3-7}$$

The constant fractional leach rate is taken to be $1/t_a$, which yields the leach rate:

$$R_{wr} = \begin{cases} -m_r(t)\rho_w / t_a & , \; t_b \le t \le t_b + t_a \\ 0 & , \; \text{otherwise} \end{cases} \tag{3-8}$$

it is understood that $s \neq r$ in each summation in the radioactive decay and production terms of Equations (3-5) through (3-7). Equations (3-5) and (3-6) are always solved numerically by the SWIFT for Windows code. Furthermore, they are solved in such a manner that the fluid concentrations never exceed the solubility limit.

Two comments are appropriate for the waste-leach equations, Equations (3-5) through (3-8). First, for the unleached component, Equation (3-7) either may be solved internally and numerically by the SWIFT for Windows code, or it may be solved externally by a code such as the ORIGEN code (Bell, 1973). In the latter case, power-law interpolation from tabular input data is used by SWIFT for Windows. Second, Equation (3-5) is a simplified version of Equation (2-4), the general transport equation for a radionuclide. Dropping the source/sink terms $C_r q$ and q_r is immaterial since there would be no wells completed into the repository. Neglecting dispersion, the E term, however, deserves some explanation. Equations (3-5)

through (3-8) are solved only for repository blocks and not for the entire system. The purpose is only to determine q_{wr}, and in order to make the algorithm as efficient as possible, physical dispersion is neglected. Furthermore, in order to calculate q_{wr} most easily, an explicit algorithm is used for the convective-transport term in Equation (3-5). This results in numerical dispersion which compensates, to some extent, for the loss of the physical dispersion term. As a final word of explanation, it should be noted that the accuracy of the solution of Equation (3-5) is important only for the case in which solubility is restrictive. Our present experience indicates that the algorithm performs quite satisfactorily, even in that case, in that it achieves the solubility to within a few percent.

4 SOURCE/SINK CONSIDERATIONS

The term "well," as used here, denotes either a source or a sink for a system. Mathematically, it is denoted by q in Equations (2-1) through (2-4). Physically, a well may be used to characterize a variety of mechanisms. Originally, when the SWIFT code was strictly a waste-injection model (see INTERCOMP [1976], Part 1, App. B), this facility was used to simulate injection and production wells. When the scope of the code was enlarged to include waste isolation, application of the well submodel was likewise enlarged. Thus, wells are now also used to simulate both aquifer recharge from upland areas and aquifer discharge into rivers and streams. The model option permitting switching from rate to pressure control is frequently useful for such simulations. In some applications, wells are used simply to establish flow boundary conditions. The purpose of this chapter is to present the well model and several related topics. The latter include the wellbore model, the injection/production of enthalpy and mass via wells and the separate sink/source terms q_X of Equations (2-1) through (2-4). The subscript X is a generic symbol denoting either fluid, heat, brine, or radionuclide.

4.1 WELL SUBMODEL

Much of the terminology presented here derives from petroleum reservoir engineering and would appear to be appropriate only for injection and production wells. However, the concepts apply equally well to any type of source/sink mechanism. The following discussion about well index, mobility, and rate allocation illustrates the general application of the terms.

There are three purposes for defining the terms well index, mobility, and rate allocation: (1) to relate the pressure of source or sink at a sub-grid scale to the average grid-block pressures; (2) to distribute fluid between different permeability layers to meet a specified net source or sink rate; and (3) to define a boundary condition of constant rate (injection or production), constant pressure, or a switch-over between the two. The second item is more applicable to the case of an actual well, whereas the first and last items are generally applicable. The following discussion provides a background about sources and sinks.

4.1.1 WELL INDEX

The region surrounding a well is called the skin (see Figure 4-1). The ability of this region to transmit fluid may be either degraded or enhanced relative to that of the undisturbed formation, depending on well completion. This transmitting capability of the skin is characterized by the well index WI, which is generally defined by the relation

$$q = \left(WI / \mu\right)\Delta p \tag{4-1a}$$

where q is the flow rate in ft^3/d (m^3/s) and Δp is the pressure drop across the skin region in psi (Pa). For specific values of fluid properties μ_o and ρ_o, the well index may be defined in terms of head drop rather than pressure drop:

$$q = WI_o \Delta H \tag{4-1b}$$

where WI_o is measured in ft^2/day or m^2/s and is defined by

$$WI_o = \rho_o \left(g / g_c\right) WI / \mu_o \tag{4-2}$$

The SWIFT for Windows code requires WI_o as input, where ρ_o and μ_o are defined in terms of reference values of pressure, temperature, and concentration.

For injection or production wells, the well index may be estimated by a one-dimensional, steady-state solution of Equation (2-1) which yields

$$WI_o = 2\pi K_s \sum_k \Delta z_k \,/\, \ell n\left(r_1 / r_w\right) \tag{4-3}$$

where K_s is the hydraulic conductivity of the skin and where index k ranges over all layers in which the well is completed. This equation is directly applicable for radial coordinates since radius r_1 is defined as the position of the first nodal point in that case.

For Cartesian coordinates, radius r_1 is not defined directly, but may be specified in terms of the radius

$$\bar{r} = \left(\Delta x \Delta y / \pi\right)^{1/2} \tag{4-4a}$$

Schematically, the assumed relation between the skin radius and this average block radius is shown in Figure 4-2. Mathematically, this relation is given by

Figure 4.1. Characterization of the Skin Region Surrounding a Well.

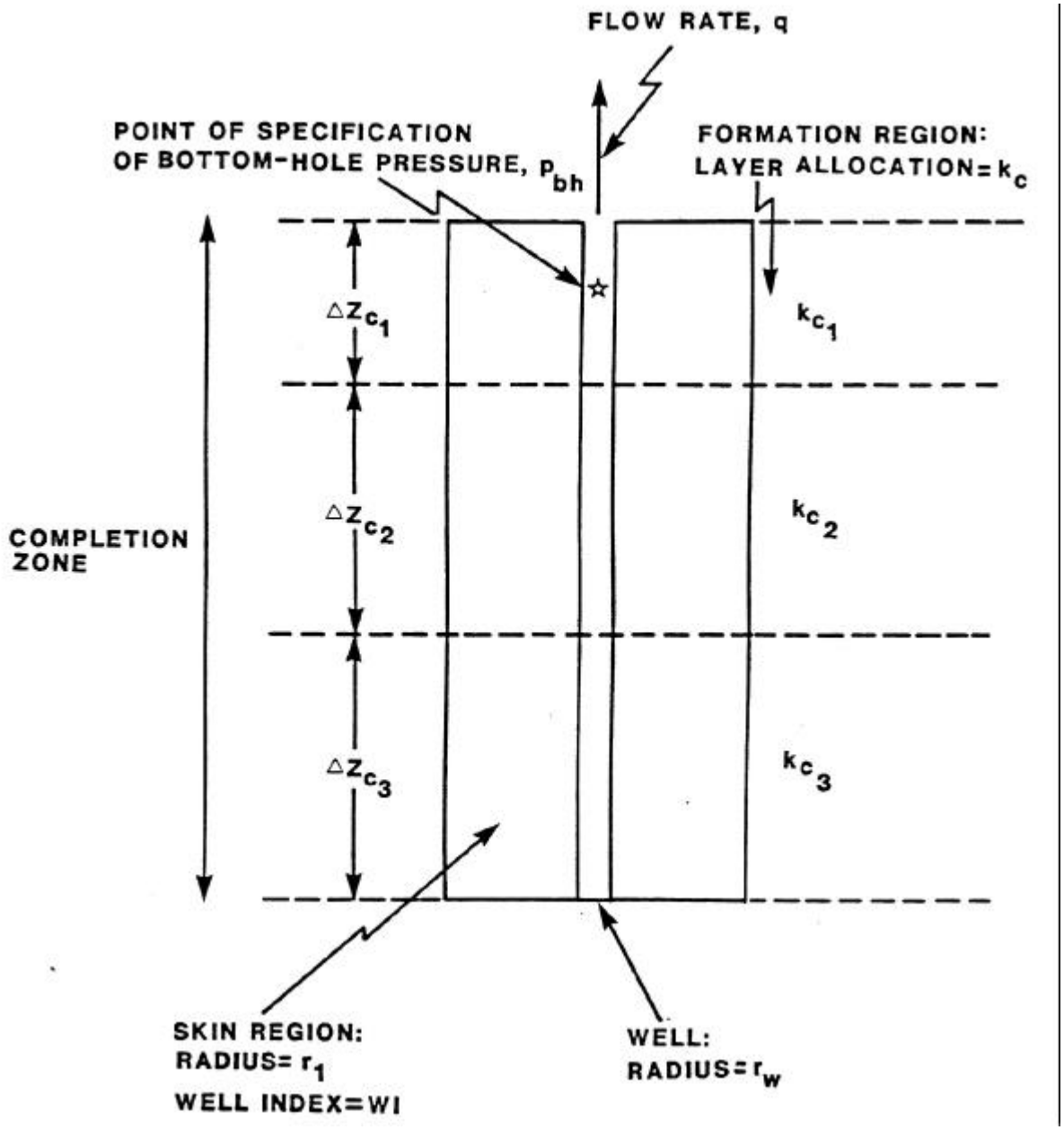
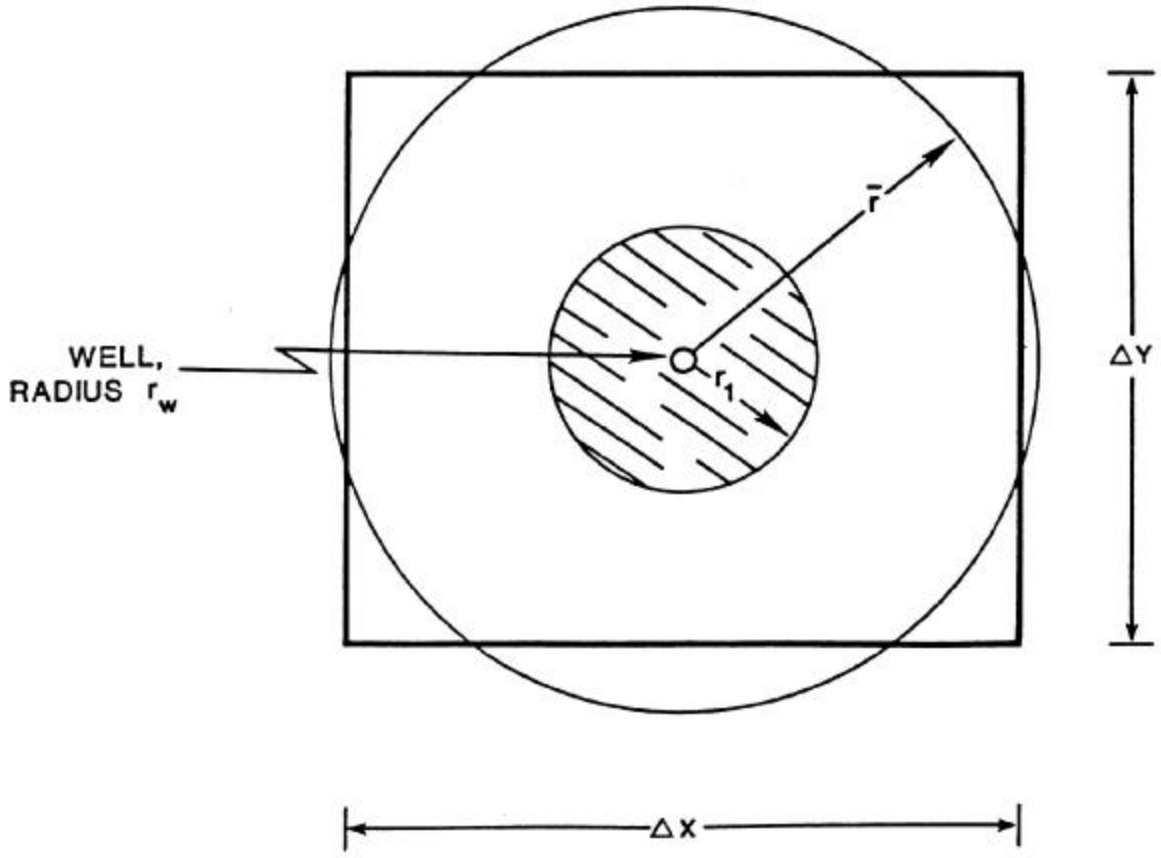


Figure 4.2. Concept of Skin Thickness for a Cartesian Grid Block. Here r is the radius of an equivalent circle and r1 denotes the distance to the average pressure of the cone of influence between rw and r.



$$\ell n\left(r_1 / r_w\right) = r_w\left\{1 + \left(\bar{r} / r_w\right)\left[1n\left(\bar{r} / r_w\right) - 1\right]\right\} \Big/ \left(\bar{r} - r_w\right) \qquad (4\text{-}4b)$$

In this case, the pressure drop Δp of Equation (4-1) is the difference in pressures between the well and the grid block pressure, and radius r_1 is taken to be the location of the radially averaged pressure of the cone of influence between radii r_w and r. An alternate approach to Equation (4-4b) for determining the effective radius r_1 within a Cartesian system is given by Pritchett and Garg (1980).

For simulation of aquifer recharge or discharge, one might choose the skin to be identical to the block itself. In this case, the well index may be related to the transmissibility, as depicted in Figure 4-3 for a block bounded on one side by a river. The well index in that case would be

$$WI_o = K_o \Delta y \Delta z / \left(\Delta x / 2\right) \qquad (4\text{-}5)$$

where K_o is the block conductivity rather than the skin conductivity.

4.1.2 MOBILITY

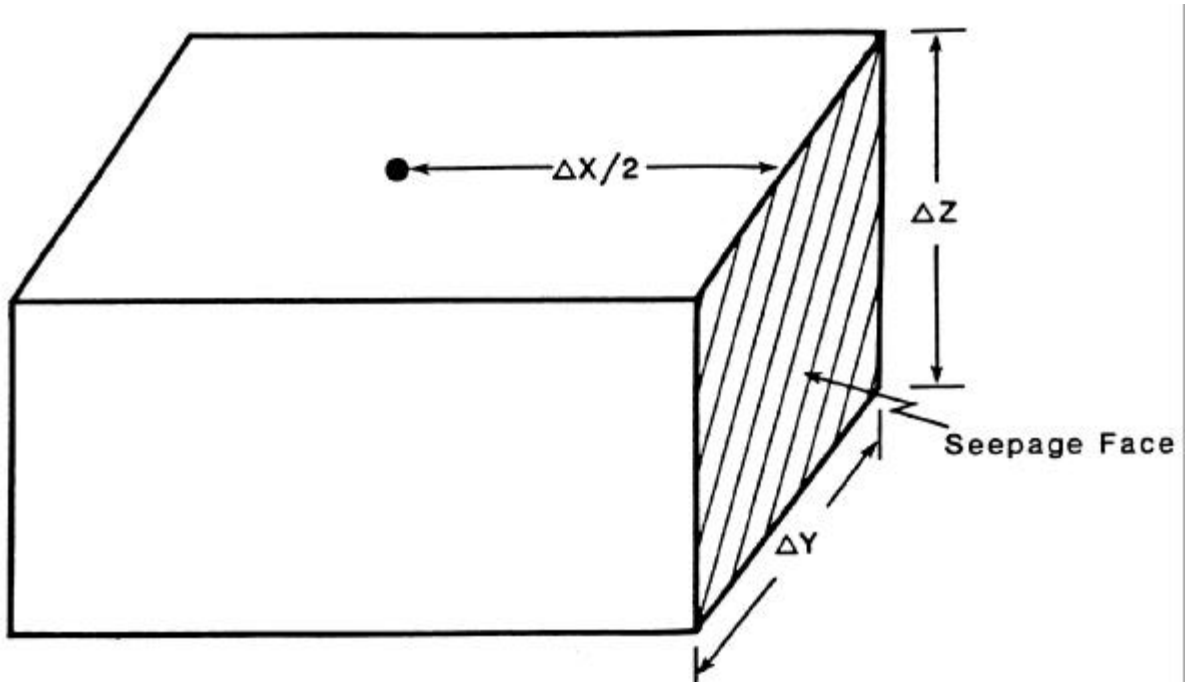
In the case of an actual well, we consider another concept which is quite similar to that of well index, namely mobility. Both terms relate to the transmissive properties of the skin. For the case of a well completed into more than one layer, however, mobility is a layer-dependent term which, to some extent, partitions flow between layers. A fractional allocation factor k_ℓ is assigned to each layer. Typically, each factor is taken to be proportional to the thickness-permeability product for that layer. Mobility for layer k is then defined to be

$$M_k = \left(k_\ell / \mu\right) WI \qquad (4\text{-}6)$$

4.1.3 RATE ALLOCATION

There are two ways in which the partitioning of flows takes place within the code, depending on the option chosen. The appropriate control parameter IINDW1 is specified in

Figure 4.3. Use of a Well to Simulate Seepage to a River.



the data input guide (Reeves et al., 1986a, p. 70). The first option (IINDW1 = 1) is that of rate allocation on the basis of mobilities alone, i.e.,

$$q_k = M_k q / \sum_k M_k \tag{4-7}$$

A second option ($|\text{IINDW1}|$ = 2 or 3) is rate allocation on the basis of mobilities and pressure drops. Here the flow rate allocated to layer k is given by

$$q_k = -\left[p_{bh} + (\rho g / g_c)\left(h_k - h_{c_1}\right) - p_k\right] M_k \tag{4-8a}$$

where the bottom-hole pressure p_{bh} is determined in terms of the grid-block pressures by the condition

$$q = \sum_k q_k \tag{4-8b}$$

Subscript c_1 denotes the first layer in which the well is completed (see Figure 4-1). The bottom-hole pressure is defined to be the well pressure at h_{c_1}, the depth of the center of this layer.

The SWIFT for Windows code applies Equations (4-8) for the case of a rate limitation. It does so in either an explicit (IINDW1 = 2) or a semi-implicit (IINDWI = -2) manner. In the former case, the evaluation of the right-hand side of Equation (4-8a) is lagged by one time step relative to the most current calculation. In the latter semi-implicit case, each q_k for the current time step n+1 is expanded about the previous time step by the relation

$$q_k^{it+1} = q_{ko}^{it} + M_k \delta p_k^{it+1} \tag{4-9a}$$

where q_o^{it} is evaluated iteratively as follows:

$$q_{ko}^1 = -\left\{\left[p_{bh}^n + (\rho g / g_c) h_{c_1}\right] - p_k^n\right\} M_k, \quad it = 1 \tag{4-9b}$$

and

$$q_{ko}^{it} = q_o^{it-1} q \Big/ \sum_k q_k^{it}, \quad it > 1 \tag{4-9c}$$

4.1.4 VARIABLE RATE/PRESSURE LIMITATION FOR TRANSIENT APPLICATION

This useful well option ($|IINWD1| = 3$), in addition to rate allocation on the basis of mobilities and pressure drops, has another feature. It permits switching between rate and pressure limitations. Thus, for injection, both a maximum bottom-hole pressure p_{bh}^{sp} and a maximum pump rate q^{sp} may be specified so that the specified rate is maintained until the maximum pressure is attained. At that time, the pressure is controlled at its maximum value, and the rate falls below the specified rate. Such cases are illustrated in Figure 4-4 for cases of both injection and production. In addition to its usefulness in simulating wells, this facility is also useful in simulating recharge, when the "bottom-hole" pressure, i.e., the surface pressure, cannot exceed atmospheric pressure. It has also been used for simulating a river by forcing a pressure limitation.

Variable rate/pressure limitation is implemented by a more flexible usage of the same equations which were used in the previous section. For example, consider the explicit (IINDW1 = 3) case for an injection well (see Figure 4.4). Here, Equations (4-8) are applied in the following manner starting from a rate-controlled situation. At each time step, the following algorithm is used:

1. Pressure P_{bh} is determined from a combination of Equations (4-8), using $q = q^{sp}$.

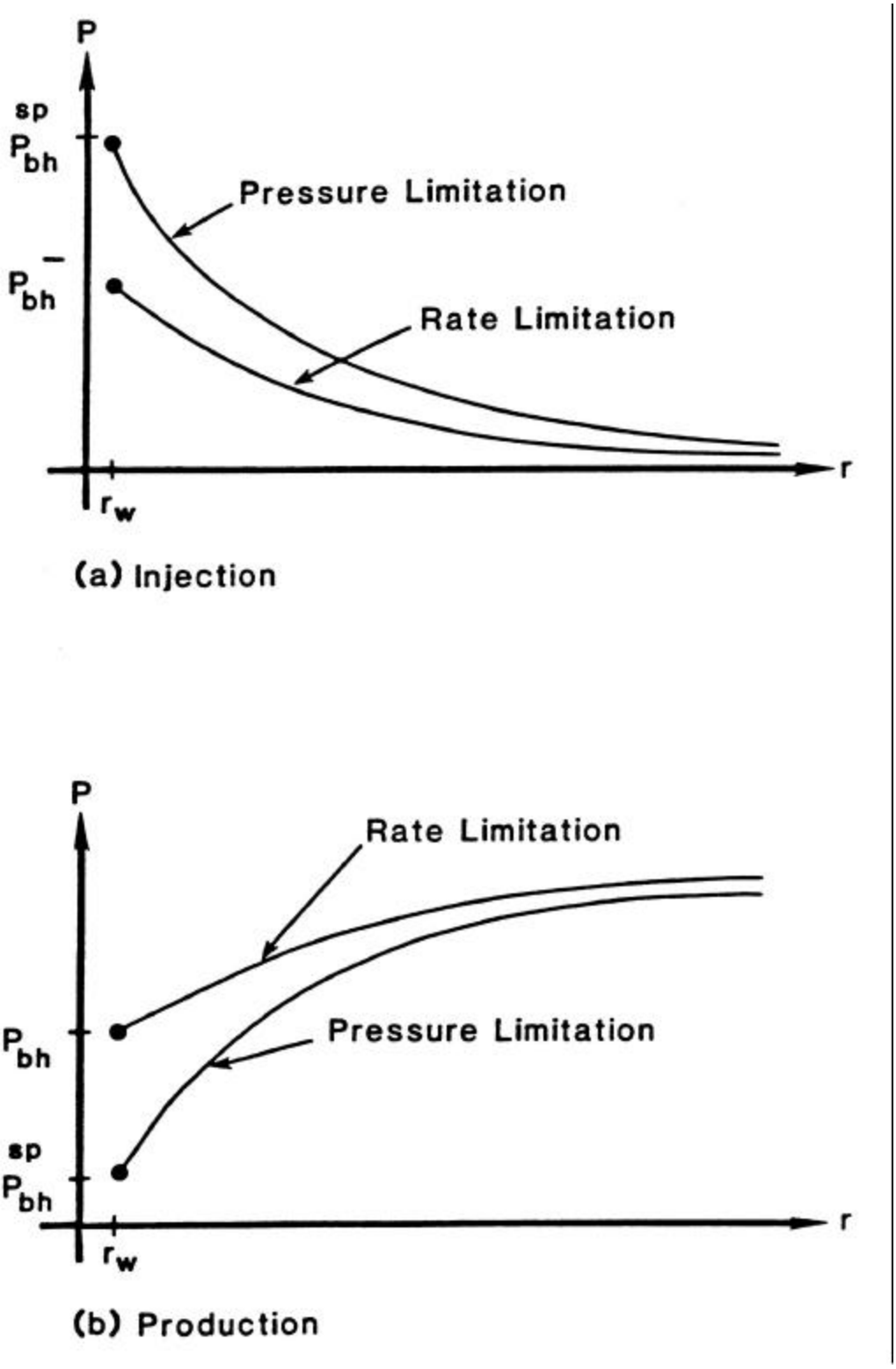
2. If $p_{bh} < p_{bh}^{sp}$, then rate control is used, and source terms q_k are calculated from Equation (4-8a), using p_{bh} as determined in Step 1 above.

3. If $p_{bh} > p_{bh}^{sp}$, then pressure control is used in that source terms q_k are calculated from Equation (4-8a), using $p_{bh} = p_{bh}^{sp}$. The total fluid injection will, in this case, be less than the specified amount, i.e., $q < q^{sp}$.

4. Pressure control, i.e., $p_{bh} = p_{bh}^{sp}$, is maintained until the pressure gradient between the wellbore and grid block becomes sufficiently great that the fluid injection rate exceeds the specified maximum, i.e., $q > q^{sp}$. At that point, the code changes back to rate limitation (Step 2 above).

The corresponding algorithm for the case of a production well is easily obtained by analogy.

Figure 4.4. Pressure Profiles Illustrating the Use of Variable Rate/Pressure Limitation.



In contrast, the implicit (IINDW1 = -3) case is implemented by using a more flexible implementation of Equations (4-8) and (4-9) than for the case IINDW1 = -2. Starting from rate control, the following procedure is used for an injection well:

1. Pressure p_{bh} is determined explicitly from a combination of Equations (4-8) using $q = q^{sp}$.

2. If $p_{bh} < p_{bh}^{sp}$, then rate control is used, and source terms q_k are calculated from an iterative implementation of Equations (4-8) and (4-9). In this case, at each iteration the coefficient of δp_k in Equation (4-9) is included in the diagonal of the propagation matrix, which carries the solution from one time step to the next, and q_o^{it} is included in the load vector.

3. If $p_{bh} > p_{bh}^{sp}$, then pressure control is used in that, from Equation (4-9b)
$$q_o = -\left\{\left[p_{bh}^{sp} + (\rho g / g_c)\left(h_k - h_{c_1}\right)\right] - p_k^n\right\} M_k \qquad (4\text{-}10)$$
Iteration is not necessary in this case.

4. Pressure control, i.e., $p_{bh} = p_{bh}^{sp}$, is maintained unless the fluid injection rate exceeds the specified maximum, i.e., $q > q^{sp}$. At that point, the code changes back to rate limitation (Step 2 above).

The corresponding algorithm for the case of a production well is easily obtained by analogy.

4.1.5 PRESSURE LIMITATION FOR STEADY-STATE APPLICATIONS

In a steady-state fluid-flow application, it is anticipated that the analyst will not need the facility for changing between rate and pressure limitations during the course of a simulation. Thus, for this case, the analyst may specify either a. flow limitation ($|\text{IINDW1}| = 2$) or a pressure limitation ($|\text{IINDW1}| = 3$). The flow limitation is implemented in the same manner as discussed above for the fully transient simulator with only one exception. Time stepping as indicated by the superscript n in Equations (4-9) is replaced by an iteration on the nonlinear flow equation.

The implementation of the pressure-limitation option is similar to that of the rate-limitation option. In the explicit case (IINDW1 > 1). Equation (4-8a) is the working

equation. The only difference is the determination of the bottom-hole pressure. For a pressure limitation, this quantity is known, i.e., $p_{bh} = p_{bh}^{sp}$. However, for a rate limitation, the bottom-hole pressure is not known and must be calculated using Equation (4-8b). In either case, the entire right side of Equation (4-8a) is determined explicitly using pressure values determined by the last iterate.

For the implicit case (IINDW1 < 1), the working equations become Equations (4-9). The mobility term is applied implicitly under both rate and pressure limitations. Again, the difference is in the bottom-hole pressure term. For the pressure limitation, the term q_o is known directly from Equation (4-10) and no further iterations are required. For the rate limitation, however, this term is not known and must be inferred via iteration using Equations (4-8) and (4-9). Hence, the application of Equations (4-9) is semi-implicit for the rate-limitation case and fully implicit for the pressure-limitation case.

4.1.6 SUMMARY

In summary, all sources of fluid and sinks of fluid are called wells regardless of whether they are actual injection or production wells. Furthermore, all such wells are characterized by rate, bottom-hole pressure, position, completion layer, well indices WI_o, layer allocation factors k_ℓ, and a specification option IINDW1. The specification option determines whether rate allocation will be via mobilities alone (IINDW1 = 1) or via mobilities and pressure drops ($|$IINDW1$|$ = 2 or 3). In addition, this specification option determines whether pressure control may also be applied ($|$IINDW1$|$ = 3) and also whether the source-sink term will be applied explicitly (IINDW1 > 1) or implicitly (IINDW1 < -1). Finally, the meaning of the option ($|$IINDW1$|$ = 3) is somewhat different in steady-state than in transient applications. For the former, this option means variable pressure-rate limitation. For the latter, however, this option means pressure limitation only.

4.2 WELLBORE SUBMODEL

For some applications, it is more convenient to specify top-hole conditions than bottom-hole conditions. To accommodate this situation, the SWIFT for Windows code

provides a wellbore submodel to account for both energy losses to the formation surrounding the wellbore and frictional pressure drops within the wellbore. Two simultaneous ordinary differential equations are solved. One is an expression of conservation of energy and is given by

$$dH = -dQ_H + d(p/\rho) \tag{4-11a}$$

This equation focuses upon the region within increment Δz and within the inner wellbore radius r_T (see Figure 4-5(b)). It states that the net enthalpy gained by the fluid in transversing this region arises from a loss of heat dQ_H to the rock formation and the work $d(p/\rho)$ performed on the fluid. Here dH and dQ_H are mass specific, i.e., Btu/1bm or J/kg.

The second equation solved by the wellbore submodel may be viewed as a definition of the mechanical-energy term in Equation (4-11a). It is

$$dp = \rho g \phi dz / g_c + \tau_f dz \tag{4-11b}$$

The first term on the right side gives the gravitational component of the pressure change, and the second gives the frictional component. Inherent in the shear-stress term τ_f is a sign to indicate that the pressure drop occurs in the direction of the flow. Actually, the code slightly generalizes Equation (4-11b) for which the wellbore is inclined by an angle θ (input variable THETA [Reeves et al., 1986a, p. 71]) to the vertical.

4.2.1 BOUNDARY CONDITIONS AND OUTER ITERATIONS

This section will address boundary conditions and then discuss heat loss and formation and frictional pressure losses in Equations (4-11). Table 4-1 shows the four different cases which arise. The conditions given there are specified in terms of the variable sets (p, T) and may, of course, be transformed via Equations (2-13) and (2-14) into the variable sets (p, H) appropriate for Equations (4-11).

Figure 4.5. Conceptualization of the Wellbore Submodel.

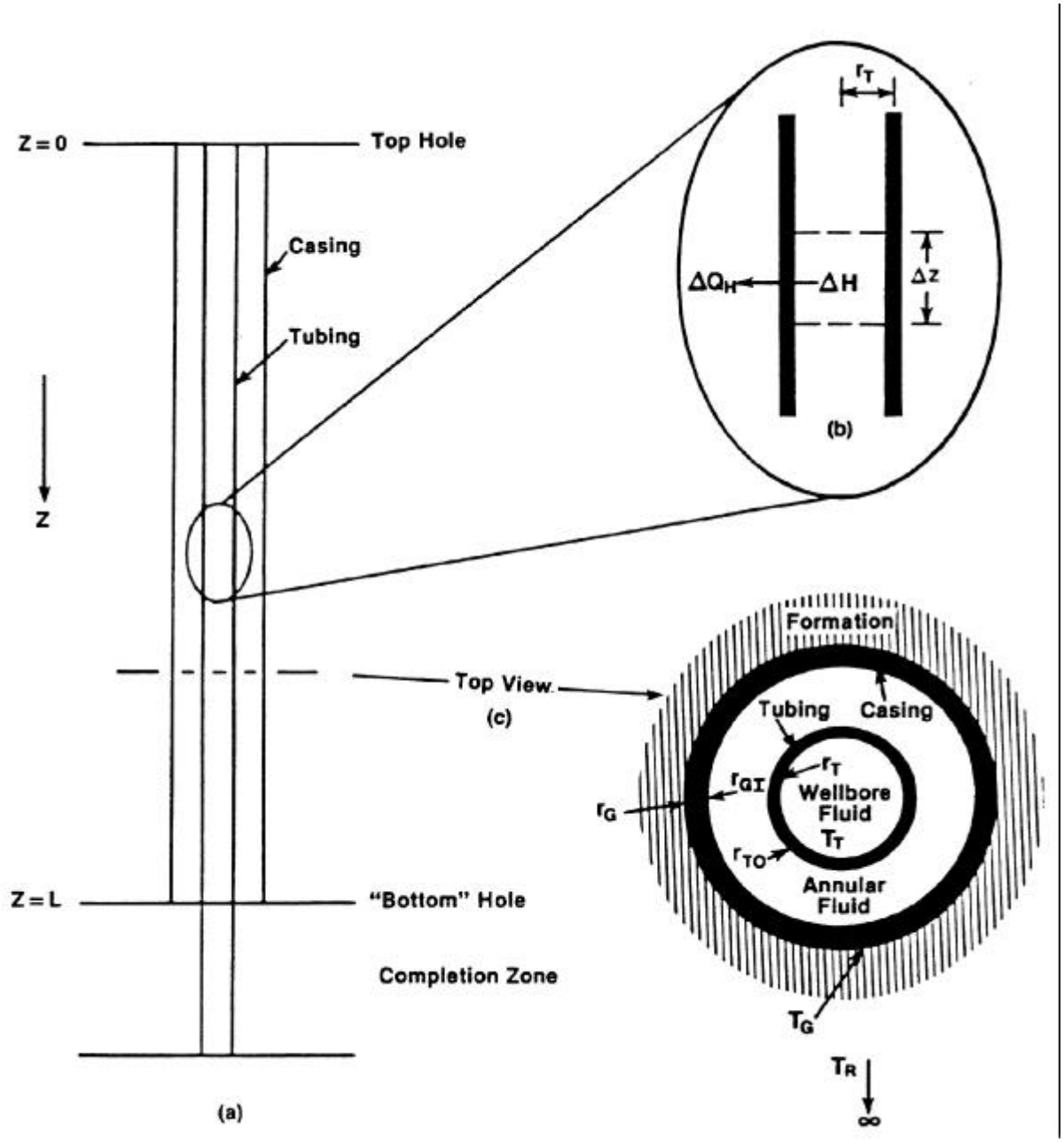


Table 4.1. Boundary Conditions for the Wellbore Integration.

Control	Well Type	
	Injection	Production
Rate	p(L), T(0)	p(L), T(L)
Pressure	p(0), T(0)	p(0), T(L)

Most of the conditions shown in Table 4.1 are specified in a natural manner. Temperatures are always given at the surface (via program input parameter TINJ [Reeves et al., 1986a, p. 71]) for injection and at bottom-hole (from current formation conditions) for production. The bottom-hole is denoted there by the parameter L. Pressures for rate-controlled production and pressure controlled injection (input variable THP [Reeves et al., 1986a. p. 73]) are specified in an analogous manner. There are, however, two cases in which the pressure condition requires explanation. First, for rate-controlled injection, the bottom-hole pressure required to inject the specified rate is computed, and this value, imposed as a boundary condition, is then translated to the surface by means of the wellbore model. Secondly, for pressure-controlled production, a minimum value of the applied top-hole pressure is specified as a limit. Whenever the specified rate of production would require a lower pressure, the limiting pressure is used. The point here is that this limiting pressure is applied at the surface in this case as a boundary condition to the wellbore model.

These two cases, i.e., rate-controlled injection and pressure-controlled production, present a problem for the numerical integration since pressure and temperature conditions are specified at opposite ends of the wellbore. Because of the strong coupling through both pressure and temperature variables, integration must proceed in the same direction for both equations. The wellbore submodel solves the problem through iteration. To be specific, take the case of rate-controlled injection. A pressure condition $p^{\gamma}(0)$ is estimated, where γ is the iteration index. and the enthalpy condition $H^{\gamma}(0)$ is computed via the relation

$$H^{\gamma}(0) = U_o + c_p\left[T(0) - T_o\right] + p^{\gamma}(0)/\rho^{\gamma}(0) \qquad (4\text{-}12)$$

Equations (4-11) are integrated in a downward direction to determine enthalpy and pressure profiles, in particular the pressure $p^{\gamma}(L)$. By comparing with the specified pressure p(L), the boundary estimate $p^{\gamma+1}(0)$ may be refined.

Iteration within the outer loop proceeds until either the tolerances (TOLX for relative change or TOLDP for absolute change [Reeves et al., 1986a, p. 68]) or the maximum number of iterations is exceeded. For relatively large frictional losses, oscillations may develop in this outer iteration loop. Therefore, a damping factor (DAMPX [Reeves et al., 1986a, p. 68]) is included. This factor has the range $1 \leq \text{DAMPX} \leq 2$ and has the effect of including more of the previous iterate ($p^{\gamma-1}(0)$ in the above case) in the extrapolation to the current iterate ($p^{\gamma+1}(0)$) as the damping factor increases. The iteration loop on boundary conditions is termed an "outer" loop since two inner iterations are required to solve the enthalpy and pressure equations for a given pass through the outer iteration loop, as will be demonstrated below.

4.2.2 HEAT LOSS TO THE FORMATION

A conceptualization of the heat-loss process is shown in Figures 4-5(a) and 4-5(c). As a fluid passes through the wellbore, a temperature drop (T_T-T_R) between the fluid within the tubing and the geologic formation (at large radial distances) drives the flow of heat outward from the wellbore to formation. A constant geothermal gradient is assumed for the formation such that

$$T_R(z) = T_R(0) + (z/L)\left[T_R(L) - T_R(0)\right] \tag{4-13}$$

Insofar as the SWIFT for Windows code is concerned, the heat flow occurs in two stages, namely, from wellbore fluid to the outer casing surface and from outer casing surface to the geologic formation. The first transport process is assumed to be a steady-state process described by heat-transfer coefficient U_G,

$$\frac{dq_H}{d\sigma} = \overline{U}_G\left(T_T - T_G\right) \tag{4-14}$$

where dq_H is the flow rate and $d\sigma$ is the differential area (a r_T) whose normal is parallel to the flow. Coefficient U_G is an input to the code (input parameter UCOEFF [Reeves et al., 1986a, p. 71]).

To determine U_G, the analyst may consider its various components to constitute a group of resistors in series with the result being

$$1/\overline{U}_G = 1/h_T + r_T\left[1n\left(r_{TO}/r_T\right)/K_T + 1n\left(r_{GI}/r_{TO}\right)/Ka + 1n\left(r_G/r_{GI}\right)/K_G\right] \quad (4\text{-}15)$$

The group of logarithmic terms represents thermal resistances for the tubing, the annular fluid and the casing walls. The logarithms appear because of the choice of a radial coordinate system. The overall multiplier r_T appears because the per-unit-area flow rate is referenced to the inside of the tubing.

The term h_T is the heat-transport coefficient appropriate for the fluid film which lines the interior of the tubing. It is given by McAdams (1954) as

$$2h_T r_T / K_W = 0.023\, R_e^{0.8} P_r^{1/8} \quad (4\text{-}16a)$$

where K_W is the thermal conductivity of the fluid. As would be expected, both a flow parameter (R_e) and a heat transport parameter (P_r) are needed to define this quantity. The former, the Reynolds number, is

$$R_e = 2\rho u r_T / \mu \quad (4\text{-}16b)$$

where u is the average wellbore velocity. The latter is the Prandtl number

$$P_r = c_p \mu / K_W \quad (4\text{-}16c)$$

Heat transport from the wellbore fluid to the outer surface of the casing is one process to be considered. The other is that of conduction from the outer radius of the casing into the rock formation. To solve this problem rigorously would most likely involve an r-z geometry with a time-dependent boundary at the inner radius r_G. However, to simplify the analysis, vertical transport is neglected, as in the work of Ramey (1962).

Further, following Ramey, it is observed that for large times, the various solutions, corresponding to different boundary conditions, all converge to the same profile. Large times, of course, are to be determined relative to the inherent time constant of the system, i.e.,

$$t >> r_G^2 / \kappa_R \quad (4\text{-}17)$$

where κ_R is the thermal diffusivity of the formation.

Carslaw and Jaeger (1959) solve the heat-transport equation for a line source subject to a constant rate of release (page 262). They also solve this equation for the case of the region bounded internally by a finite-circular cylinder. For a constant boundary temperature, the results are reported on page 336 and for a constant rate of release, the results are given on page 339. In each case, the solution for large times is the following (in our notation):

$$dq_H = 2\pi K_R \left(T_G - T_R\right) dz / F(t) \tag{4-18a}$$

where K_R is the thermal conductivity of the rock, T_G is the temperature at the casing-rock interface and T_R is the initial temperature of the formation (at depth z). The function F is given by

$$F(t) = -1n\left[r_G / \sqrt{\left(2\kappa_R t\right)}\right] - \gamma^* / 2 \tag{4-18b}$$

where γ^* is the Euler constant

$$\gamma^* = 0.57722... \tag{4-18c}$$

Ramey (1962) further asserts that a radiation boundary at $r = r_G$ will also yield the same solution for large times. Thus, Equation (4-18a) is chosen to describe the heat flow rate into the formation.

However, before this equation can be used for the wellbore model, it must be modified in two respects. First of all, it must be changed to a mass-specific basis, i.e.,

$$dQ_H = dq_H / q \tag{4-19}$$

where q is the mass flow rate of the fluid. Second of all, the temperature T_G should be eliminated using Equation (4-14) with $d\sigma = 2\pi r_T dz$. The result is

$$dQ_H = 2\pi r_G \overline{U}\left(T_T - T_R\right) dz / qF(t) \tag{4-20a}$$

where the new heat-transfer coefficient is defined as

$$1/\overline{U} = 1/\left[\overline{U}_G F(t)\right] + r_T / K_R \tag{4-20b}$$

Equations (4-20 a & b) are the working equations for the heat loss to the formation.

4.2.3 FRICTIONAL PRESSURE DROPS IN THE WELLBORE

The frictional pressure drop is represented in Equation (4-11) by the term τ_f, where $\tau_f r_T/2$ is the shear stress (McCabe and Smith, 1976). This term is related to the friction factor f via

$$\tau_f = f\rho u^2 / 2r_T \tag{4-21}$$

For laminar flow, the friction factor is taken to be

$$f = 64 / R_e \quad , \quad \overline{R}_e \le 3.3 \tag{4-22a}$$

where the Reynolds number is given in Equation (4-16b) and where

$$\overline{R}_e = \log_{10}\left(R_e\right) \tag{4-22b}$$

For the transition zone, the following is used

$$f = 10^x \quad , \quad 3.3 < \overline{R}_e \le 3.6 \tag{4-23a}$$

with the exponent given by the polynomial

$$x = 260.67 - 228.62\ \overline{R}_e + 66.307\ \overline{R}_e^2 + 6.3944\ \overline{R}_e^3 \tag{4-23b}$$

For the turbulent-flow region, the surface roughness f_r of the wellbore becomes important so that

$$f = f\left(R_e, f_r\right) \tag{4-24}$$

Here, analytical fits to the curve of L.F. Moody are used (see Schlichting, 1955).

4.2.4 NUMERICAL SOLUTIONS AND INNER ITERATIONS

Within the wellbore model, the enthalpy and pressure equations of Equations (4-11) are integrated. As discussed in Section 4.2.1, the direction of integration of these ordinary differential equations is downward for an injection well and upward for a production well. Outer iterations on the boundary conditions are required for a rate-controlled injection well and for a pressure-controlled production well because conditions are specified on opposite sides of the wellbore. Outer iterations are indicated by the superscript γ in the notation below.

For each incremental step Δz, there are two iterations. The innermost iteration (superscript α) is for the enthalpy equation. Equation (4-11a) in incremental form is

$$\Delta H_k^{\alpha\beta\gamma} = -\Delta Q_H^{\alpha\beta\gamma} + \Delta\left(p_k^{\gamma} / \rho_k^{\beta-1,\gamma}\right) \tag{4-25a}$$

$$H_k^{\alpha\beta\gamma} = H_{k-1}^{\gamma} + \Delta H_k^{\alpha\beta\gamma} \tag{4-25b}$$

The heat lost to the formation is determined from the tubing temperatures $T_{Tk}^{\alpha-1,\beta\gamma}$ at iteration level α-l. From table look-up, the temperature $T_{Tk}^{\alpha\beta\nu}$ is obtained from the enthalpy of the wellbore fluid within Δz for iteration level α. Convergence within the innermost loop is determined by comparing these two temperatures (relative tolerance parameter, EPS [Reeves et al., 1986a, p. 69]).

The pressure equation is integrated within the intermediate iteration loop β. Here, pressure is treated as the independent variable and a constant increment Δp is used (input parameter, DELPW [Reeves et al., 1986a, p. 22]). Thus, Equation (4-11b) becomes

$$\Delta z_k^{\beta\gamma} = \Delta p \Big/ \left(\rho_k^{\beta\gamma} g / g_c + \tau_f^{\beta\gamma}\right) \tag{4-26}$$

Both the density ρ and the frictional shear-stress term τ_f are functions of temperature. For the latter, the dependence enters through the Reynolds number dependence on density and viscosity (see Equation (4-16b)). Thus. the intermediate iteration loop updates the calculation of the space step Δz using the most recent value of the temperature. Coupling to the enthalpy solution then occurs through the heat lost to the formation, which depends linearly upon this space step (see Equation (4-20)). An absolute tolerance of 0.5 ft (0.15 m) is the convergence criterion for the intermediate iteration level β.

The desired results from the wellbore model are the enthalpy and pressure delivered either at the bottom-hole (injection) or at the top hole (production). These two quantities are obtained by interpolation across the final interval $\left(z_k^{\gamma}, z_k^{\gamma} + \Delta z_k^{\gamma}\right)$, which contains the pipe length L (input parameter X [Reeves et al., 1986a, p. 71]).

4.3 WELLS AND SINKS

The mathematical formulation as given in Equations (2-1) through (2-4) makes provisions for both wells (q) and other sinks (q_x). Section 4.1 focuses upon fluid production/ injection from wells. Section 4.2 focuses upon frictional pressure losses within and heat losses from the wellbore. These two factors bear upon both the fluid production/injection and the enthalpy production/injection from wells. That discussion is enlarged to consider the production/injection of brine and radionuclides. Also, since the sink terms are closely related to the well terms, they will be combined into composite terms Q_X.

4.3.1 FLUID

Equation (4-9) gives the most general form for fluid production/injection. Coalescing this expression with a fluid sink/source q_W gives

$$Q_W = Q_{Wo} + M_W \delta p \tag{4-27a}$$

where

$$Q_{Wo} = q_o + q_W \tag{4-27b}$$

and

$$M_W = M \tag{4-27c}$$

Sink terms may be either positive (sink) or negative (source). Furthermore, there may be several independent terms of the form of Equation (4-27), each of which would be characterized by unique grid-block indices i, j, and k.

4.3.2 HEAT

For injection, enthalpy H_l (equal to H(L) in equation (4-12)) is convected through the well skin and into the formation via the relation $H_l q$. For production, the grid-block enthalpy is convected from the formation into the well via the relation

$$H^n + wc_p \delta T q$$

Here, n denotes the next-previous time level, w the time weighting factor (Section 6.1.4) and δT is the temperature change during time-step n+1. As shown, the change in enthalpy is assumed to arise primarily from the change in sensible heat.

Combining the terms for injection and production via wells with a prescribed sink/source yields

$$Q_H = Q_{Ho} + wM_H\delta T \tag{4-28a}$$

where

$$Q_{Ho} = \begin{cases} H_I q + q_H & , \ q < 0 \\ H^n q + q_H & , \ q \geq 0 \end{cases} \tag{4-28b}$$

and

$$M_H = \begin{cases} 0 & , \ q < 0 \\ c_p q & , \ q \geq 0 \end{cases} \tag{4-28c}$$

4.3.3 BRINE

For the brine, no sink/source other than a well is permitted. Otherwise, the appropriate sink/source relation may be formulated in analogy with that for heat. It is:

$$Q_c = Q_{Co} + wM_C\delta\hat{C} \tag{4-29a}$$

where

$$Q_{Co} = \begin{cases} \hat{C}_I q & , \ q < 0 \\ \hat{C}^n q & , \ q \geq 0 \end{cases} \tag{4-29b}$$

and

$$M_C = \begin{cases} 0 & , \ q < 0 \\ q & , \ q \geq 0 \end{cases} \tag{4-29c}$$

4.3.4 RADIONUCLIDE

Two species-dependent sources are permitted for radionuclides. They are (1) leaching of radioactive wastes from within a designated repository region (Section 3.4), and (2) disposal of a soluble radioactive waste from outside the system. Production via wells is also included. (Injection via wells is not included.) When all of these effects are combined, the resulting expression is:

$$Q_r = Q_{ro} + wM_r\delta C_r \tag{4-30a}$$

where

$$Q_{ro} = \begin{cases} q_{Cr} - q_{wr} & , \ q < 0 \\ q_{Cr} - q_{wr} + C_r^n q & , \ q \geq 0 \end{cases} \tag{4-30b}$$

and

$$M_r = \begin{cases} 0 & , \ q < 0 \\ q & , \ q \geq 0 \end{cases} \tag{4-30c}$$

Equation (4-30), since it is written in its most general form, contains the three processes of sink withdrawal, waste leaching, and well production.

5 INITIAL BOUNDARY CONDITIONS

The system description given in Chapter 3 and source/sink considerations given in Chapter 4 only partly define the hydrogeological system. In addition, both initial and boundary conditions must be given. The various options available to the analyst for specifying these two items are discussed in this chapter.

5.1 INITIAL CONDITIONS

5.1.1 PRESSURE

For determination of the initial pressure distribution, the code first assumes hydrostatic equilibrium throughout the entire system and then determines fluid pressures. In addition, however, the analyst may specify a constant Darcy flux in the x-direction. In this case, a numerical integration of the velocity equation (Equation 2-9) is performed and the pressure distribution determined for hydrostatic equilibrium is modified accordingly. Boundary pressures are also determined for the constant-flux option.

Hydrostatic-Equilibrium Condition. In order to define the condition of hydrostatic equilibrium, the analyst provides the initial fluid pressure p_I (input variable PINIT [Reeves et al., 1986a, p. 28]) at any chosen reference point H_I (HINIT [Reeves et al., 1986a, p. 28]), as shown in Figure 5.1. The code then determines the total pressure via

$$\tilde{p}_I = p_I - \rho_o\left(g / g_c\right)\left(h_I - h_D\right) \tag{5-1}$$

where h_D (input variable HDATUM [Reeves et al., 1986a, p. 28]) is the datum position relative to the reference plane (see Figure 5-1) and where ρ_o is the reference density.

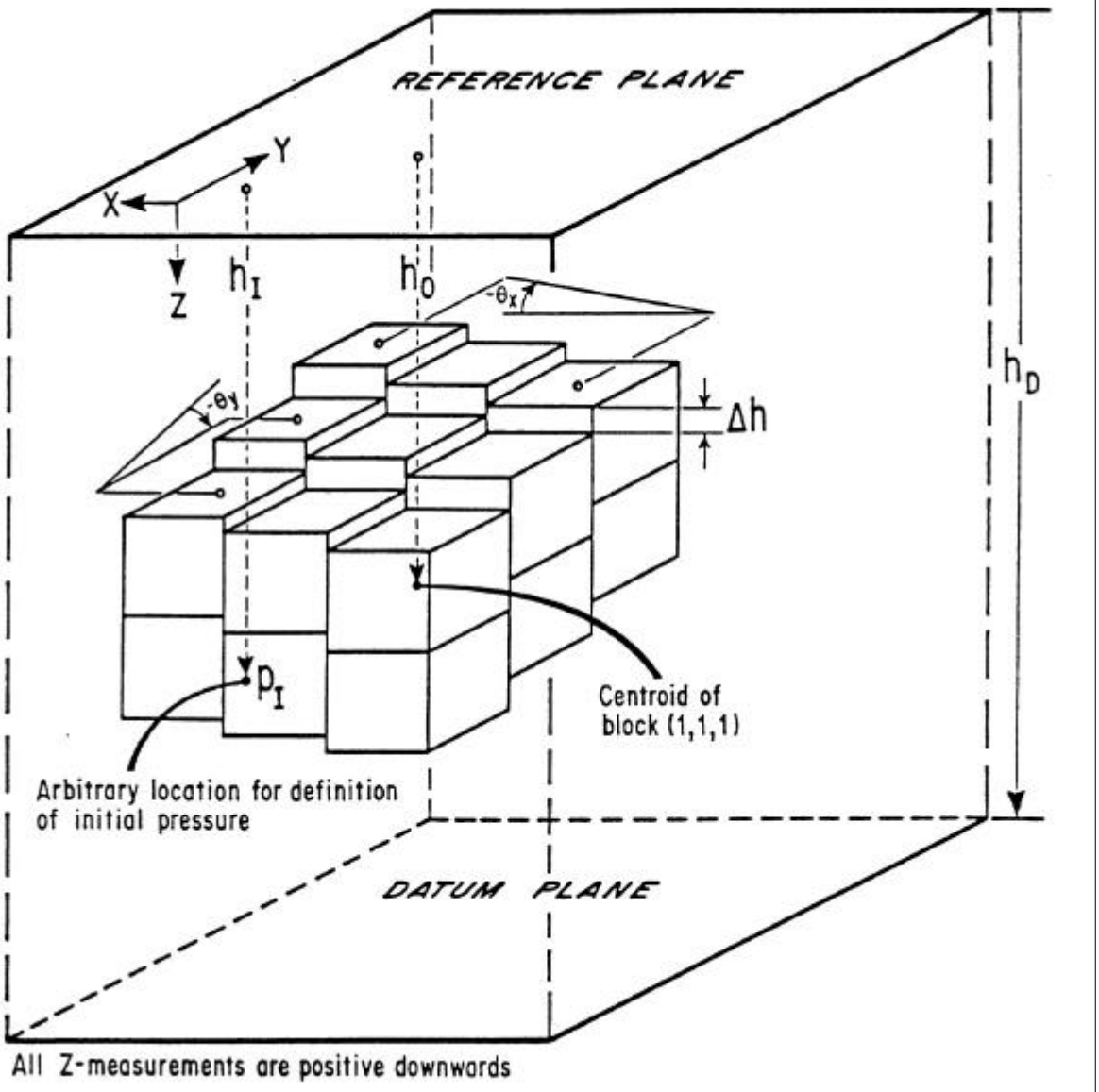
Because of density variations, total pressure $\tilde{p}_I$ is used to fix fluid pressures only for the top k = 1 layer of grid blocks:

$$p_{I1} = \tilde{p}_I + \rho_o\left(g / g_c\right)\left(h_1 - h_D\right) \tag{5-2}$$

where indices i and j have been suppressed. For each column of grid blocks, i.e., constant i and j, the remaining pressures are prescribed via the relation

Figure 5.1. Specification of Geometry and Initial Pressure.

Here P_I = PINIT, h_I = HINIT, h_D = HDATUM, sin θ_x = SINX, sin θ_y = SINY, h_o = DEPTH, and Δh = HADD.



$$p_I(h) = p_{I1} + \int_{h_1}^{h} \rho(p,z)(g/g_c)dz \tag{5-3}$$

In general, the density is a function of both p and z (through initial values of T and C). Because of this complexity, Equation (5-3) is solved with an iterative procedure. In incremental form, it is

$$p_{Ik}^{it+1} = p_{I1} + \sum_{\ell=1}^{k-1} \rho_{\ell+1/2}^{it}(g/g_c)(h_{\ell+1} - h_\ell) \quad k > 1 \tag{5-4a}$$

Here, the density is given by

$$\rho_{\ell+1/2}^{it} = \begin{cases} \rho_o & , \ it = 0 \\ 1/2\left(\rho_\ell^{it} + \rho_{\ell+1}^{it}\right) & , \ it \geq 0 \end{cases} \tag{5-4b}$$

Constant-Flux Initial Condition. In order to specify the constant-flux condition, the analyst provides a Darcy velocity, which is assumed to be directed in the x-direction. The code then uses an algorithm based on the velocity equation, Equation (2-9). The analysis begins with the numerical implementation of this equation, namely

$$u_x = -T_{Wxx}\left(\Delta_x p - \rho g \Delta_x z / g_c\right) \tag{5-5}$$

This equation shows the mass flow between neighboring grid blocks to be proportional to differences in both fluid and gravitational potentials with the proportionality constant being the transmissibility.

Expressing the potential in terms of a deviation δp from hydrostatic equilibrium p_I, one obtains approximately

$$\Delta p - \rho g \Delta z / g_c = \Delta\left(\delta p + \tilde{p}_I\right) \cong \Delta \delta p \tag{5-6}$$

Here and below, the subscripts x and W are assumed implicitly and dropped from the notation. Grid-block indices, however, must be added so that Equation (5-5) becomes

$$u_{i+1/2,jk} = T_{i+1/2,jk}\left(\delta p_{ijk} - \delta p_{i+1,jk}\right) \tag{5-7}$$

with half-integer indices specifying grid-block interfaces. Now, the Darcy velocity $\bar{u}$, specified in the input to SWIFT for Windows as VEL (Reeves et al., 1986a, p. 55), represents the mean flux resulting from an average over the entire y-z face of the system. This velocity is also constant throughout the entire x domain of the system. Hence

$$\bar{u} = \sum_{j,k} u_{i+1/2,jk} \Delta y_j \Delta y_k / \sigma \tag{5-8a}$$

where

$$\sigma = \sum_{j,k} \Delta y_j \Delta z_k \tag{5-8b}$$

The result of combining Equations (5-7) and (5-8) is

$$\bar{u} = \bar{T}_{i+1/2} \left(\delta p_i - \delta p_{i+1} \right) \tag{5-9a}$$

where

$$\bar{T}_{i+1/2} = \sum_{j,k} T_{i+1/2,jk} \Delta y_j \Delta z_k / \sigma \tag{5-9b}$$

and where variations in the pressure changes in the y and z directions have been neglected.

The pressure change for the i = 1 grid blocks is assumed to be zero so that, from Equation (5-9),

$$\delta p_2 = -\bar{u} / \bar{T}_{3/2} \quad , \quad \delta p_3 = -\bar{u} \left(1 / \bar{T}_{5/2} + 1 / \bar{T}_{3/2} \right)$$

and, in general,

$$\delta p_i = -\bar{u} / \tilde{T}_i \tag{5-10a}$$

where

$$1 / \tilde{T}_i = \sum_{\ell=2}^{i} 1 / \bar{T}_{\ell-1/2} \tag{5-10b}$$

Thus, for the constant-flux option, the fluid pressure corresponding to hydrostatic equilibrium is modified as follows:

$$P_{Iijk} \rightarrow P_{Iijk} + \delta p_i \tag{5-11}$$

Constant-Flux Boundary Condition. Whenever a constant flux initial condition is prescribed, the boundary conditions for the x-direction are also specified. The algorithm, in this case, is developed directly from Equation (5-5). Here, it is assumed that the grid-block boundary pressures are prescribed at the same elevation as the grid-block nodes. Thus, there is no potential change due to elevation differences, and Equation (5-7) may be written as follows for the i = 1 grid block

$$u_{ojk} = T_{ojk}\left(p_{ojk} - p_{ljk}\right) \tag{5-12a}$$

where p_1 is the pressure at the grid block center and p_o is the pressure at the boundary. The transmissibility T_{ojk} refers to the external boundary of the grid block i = 1 and is approximated by

$$T_{ojk} = \left(\Delta x_1 + \Delta x_2\right) T_{3/2,jk} / \Delta x_1 \tag{5-12b}$$

Following the same procedure used in the development of Equation (5-9), the relation obtained is

$$\bar{u} = \bar{\bar{T}}_o\left(p_{ojk} - p_{ijk}\right) \tag{5-13a}$$

where

$$\bar{\bar{T}}_o = \sum_{j,k} T_{ojk} \Delta y_j \Delta z_k / \sigma \tag{5-13b}$$

and variations in the pressure drops in the y and z directions have been neglected.

Equation (5-13a) may then be solved directly for the boundary pressure:

$$p_{ojk} = p_{ljk} + \bar{u} / \bar{\bar{T}}_o \tag{5-14}$$

In a similar manner, the pressure for the grid block i = n_x may be expressed as

$$p_{ojk} = p_{n_x jk} - \bar{u} / \bar{\bar{T}}_o \tag{5-15}$$

Equations (5-14) and (5-15) express the boundary pressure p_o of a given grid block in terms of the initial pressure p at the block center. This pressure is then assumed to hold

throughout the simulation for arbitrary values of the grid-block pressure. This means that the flow, in general, will vary from its initial value u to a different value e_W. Denoting the boundary block by ℓ = (i, j, k) and simplifying the notation in a rather obvious manner, Equation (5-12a) may be written

$$e_{W\ell} = T_\ell \left(p_{o\ell} - p_1 \right) \tag{5-16}$$

Here, T_ℓ is the transmissibility appropriate for the outer boundary of the block, as in Equation (5-12b). However, p_1 is here a time-dependent pressure which varies from time-step n to time-step n+l:

$$p_1 = p_1^n + \delta p \tag{5-17}$$

Substituting Equation (5-17) into Equation (5-16), one obtains

$$e_{W\ell} = a_{W\ell} + b_{W\ell} \delta p_\ell \tag{5-18a}$$

where

$$a_{W\ell} = T_\ell \left(p_{o\ell} - p_1^n \right) \tag{5-18b}$$

and

$$b_{W\ell} = -T_\ell \tag{5-18c}$$

Equations (5-18) are the working equations of the model for boundary fluid flows.

These boundary flows give rise to the convective transport of enthalpy and contaminant (brine and radionuclide). Boundary conditions resulting from both convection and dispersion/conduction are considered in Sections 5.3.2 and 5.3.3.

5.1.2 TEMPERATURE

In order to specify the initial temperatures, an interpolation function

$$\left(z_{Ii}, T_{Ii} \right) \ , \ i = 1,2,\ldots, n_T \tag{5-19}$$

is supplied. (In terms of program variables, the notation is (ZT(I), TD(1)), I = 1,2,...,NDT [Reeves et al., 1986a, p. 26]). Assuming the initial temperature to be independent of x and y,

the temperatures of both the system and its over/underburden are determined with linear interpolation.

5.1.3 BRINE AND RADIONUCLIDE CONCENTRATIONS

Both brine and radionuclide concentrations are initialized by direct input.

5.1.4 ROCK MATRIX

The analyst may designate certain global grid blocks as dual porosity blocks. To initialize the rock matrix in these blocks, equilibrium is assumed so that pressure, temperature, brine and radionuclide concentrations are the same both within the fractures and with the rock matrix.

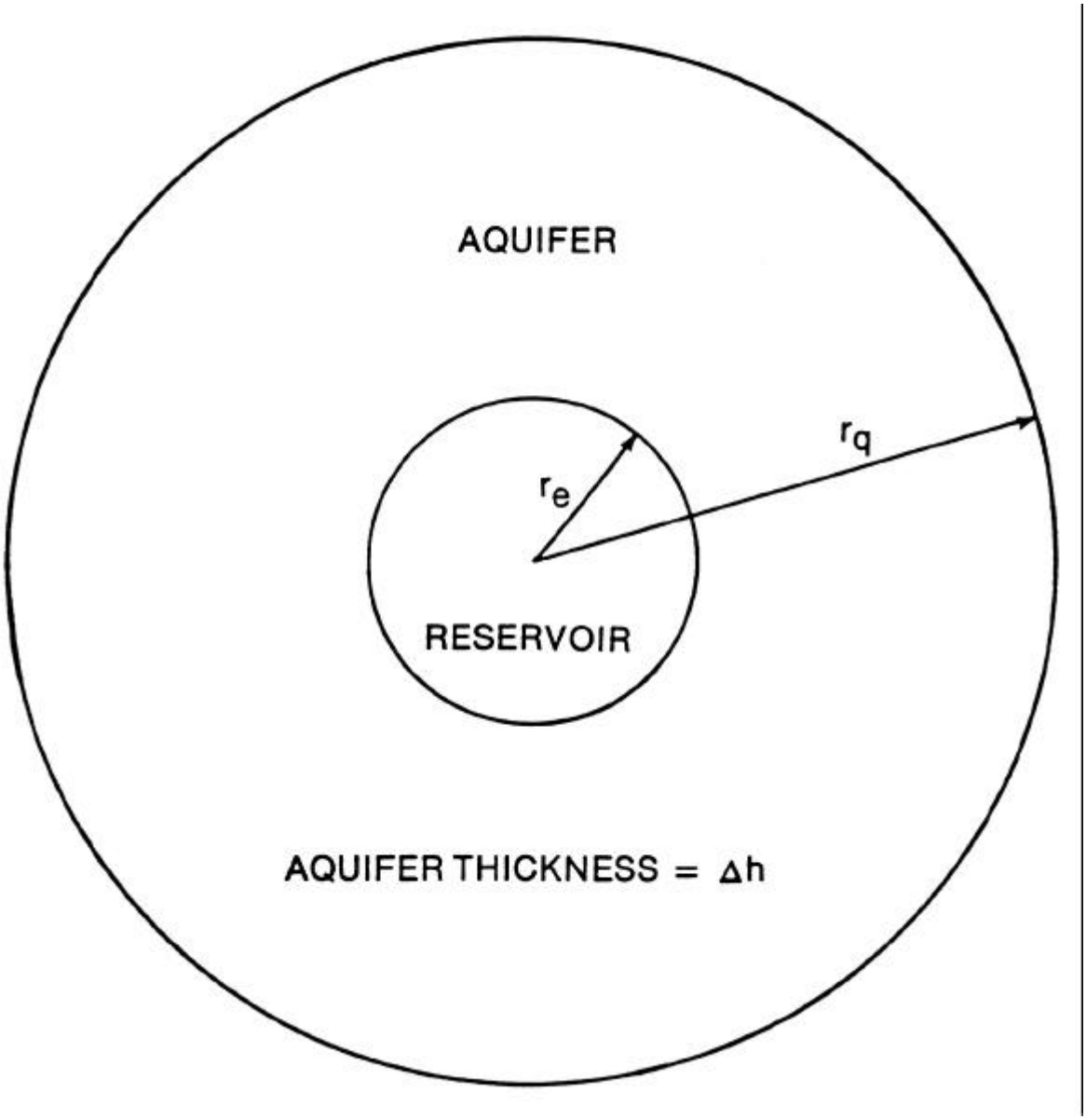
5.2 AQUIFER SUBMODELS

For purposes of this discussion, a distinction is made between the term "reservoir" and the term "aquifer". The former is taken to be that portion of the system for which a detailed simulation is desired, whereas the latter is taken to be the remaining portion of the system. The aquifer provides boundary conditions for the reservoir. Three different aquifer models are provided by the SWIFT code in order to realistically determine reservoir boundary conditions. namely (1) an unsteady-state aquifer, (2) a steady state aquifer, and (3) a pot aquifer. As will be shown, the latter two are special cases of the first. In each, a type three condition is provided for each boundary grid block in which the rate of flow from aquifer to reservoir is a function of the pressure changes within the block.

5.2.1 UNSTEADY-STATE AQUIFER

Definition of the Aquifer Problem. Here it is assumed that the geometrical shape of the aquifer is that of a cylindrical shell (see Figure 5-2). The inner dimension r_e is the external radius of the reservoir. For a Cartesian geometry this dimension, most likely, would be chosen as the radius of a circle of equal surface area. The outer dimension r_q is the external radius chosen for the aquifer itself. This dimension may be chosen to be either finite

Figure 5.2. Geometrical Characterization of the Aquifer.



or infinite, the latter being the default option for which the appropriate influence function is supplied internally within the code. The thickness Δh is representative of the entire aquifer and appears in the input through both a permeability thickness product and a porosity-thickness product.

Starting from Equation (2-1) and assuming constant properties, the flow equation for the aquifer may be written in dimensionless form as follows:

$$\frac{1}{r_D}\frac{\partial}{\partial r_D}\left(r_D\frac{\partial p_D}{\partial r_D}\right)=\frac{\partial p_D}{\partial t_D} \tag{5-20}$$

where the radius and time variables are given by

$$r_D = r / r_e \tag{5-21a}$$

and

$$t_D = t / t_e \tag{5-21b}$$

with

$$t_e = \mu\phi\left(c_W + c_r\right)r_e^2 k \tag{5-22}$$

The range of the radius variable within the aquifer is

$$1 \le r_D \le R_q \ , \ R_q = r_q / r_e \tag{5-23}$$

It is assumed that initially the pressure is constant, i.e.,

$$p\left(r, t=0\right) = p_o \tag{5-24}$$

Thus, the dimensionless pressure variable is

$$p_D = p / p_o \tag{5-25a}$$

Furthermore, the dimensionless pressure drop may be defined as

$$\Delta p_D = 1 - p_D \tag{5-25b}$$

and Equation (5-2) may be rewritten as

$$\frac{1}{r_D}\frac{\partial}{\partial r_D}\left(r_D\frac{\partial \Delta p_D}{\partial r_D}\right)=\frac{\partial \Delta p_D}{\partial T_D} \tag{5-26}$$

The Solutions of Van Everdingen and Hurst. Van Everdingen and Hurst (1949) determined two different solutions for Equations (5-26) corresponding to two different boundary conditions at the reservoir-aquifer boundary. For the constant-terminal-rate case, the condition is taken to be

$$\frac{\partial \Delta p_D}{\partial r_D}\left(r_D = 1, t_D\right) = -1 \tag{5-27a}$$

and for the constant-terminal-pressure case, it is

$$\Delta p_D\left(r_D = 1, t_D\right) = 1 \tag{5-27b}$$

These authors determine a terminal-rate influence function

$$P_I\left(t_D\right) = \Delta p_D\left(r_D = 1, t_D\right) \tag{5-28a}$$

for the rate condition given in Equation (5-27a). They also determine a terminal-pressure influence function

$$Q_I\left(t_D\right) = \int_o^{t_d} \frac{\partial \Delta p_D}{\partial r_D}\left(r_D = 1\right) dt_D \tag{5-28b}$$

for the pressure condition given in Equation (5-27b). Van Everdingen and Hurst present tables of the dimensionless functions P_I and Q_I for three different conditions appropriate for the exterior aquifer boundary R_q, namely, (1) infinite R_q, (2) no flow at (finite) R_q, and (3) zero pressure drop at R_q. The table for P_I, corresponding to an infinite Rq, is coded internally in SWIFT for Windows as a default option.

The significance of these functions arises through the principle of superposition. Thus, if the pressure history is known, the cumulative water flow W_e, in volumetric units, is given by the convolution integral

$$W_e\left(t_D\right) = B_e \int_o^{t_D} \frac{\partial \Delta p}{\partial \lambda} Q_I\left(t_D - \lambda\right) d\lambda \tag{5-29}$$

where

$$B_e = 2\pi\Delta h\phi\left(c_W + c_R\right)r_e^2 \tag{5-30}$$

Furthermore, if the flow-rate history is known, the pressure history (at $r_D = 1$) may be determined by a similar integral involving the terminal pressure function P_I. Such analyses may be applied either to a wellbore bounded by an aquifer or to a reservoir bounded by an aquifer. It is the latter application which is of interest here.

The Approximation of Carter and Tracy. Equation (5-29) gives the cumulative water influx through the periphery of an aquifer as a function of the pressure change occurring there. It is, therefore, the type of relationship needed for computer applications. However, it is awkward to apply in practice. Carter and Tracy (1960) have remedied this situation. By assuming a linear variation in the water influx over a time increment, they obtain a relationship which is quite suitable for computer implementation. Recasting their relation into a form suitable for the SWIFT for Windows code, we obtain, for the flow rate,

$$e_W = a_W + b_W\delta p \tag{5-31a}$$

where

$$a_W = -\left(B_e\Delta p^n - W_e^n P'^{n+1}_I\right)\Big/ t_e\left(P_I^{n+1} - t_D^n P'^{n+1}_I\right) \tag{5-31b}$$

and

$$b_W = -B_e\Big/ t_e\left(P_I^{n+1} - t_D^n P'^{n+1}_I\right) \tag{5-31c}$$

Here the discrete time notation has been introduced in which the time level is indicated by a superscript. Furthermore, e_W is related to W_e by the relation

$$-e_W = \left(W_e^{n+1} - W_e^n\right)\left(t^{n+1} - t^n\right) \tag{5-32}$$

In accordance with the adopted sign convention, an increase in storage in the aquifer corresponds to a negative (outward) flow for the aquifer.

The results given in Equation (5-31) depend upon the terminal-rate-influence function $P_I(t_D)$. As indicated above, tables of this function for the infinite-aquifer case are coded

internally within SWIFT for Windows. However, one has the option of using any function desired. Tables for P_I, corresponding to two other cylindrically symmetric configurations, are given in Van Everdingen and Hurst (1949). These tables have been extended by Sikora (1962). In addition, other tables, corresponding to more general configurations, are given in Katz et al. (1963) and Katz and Coats (1968).

Distribution of Flow Rate Over Boundary Grid Blocks. In order to distribute the flow rate given in Equations (5-31) to the various boundary blocks, a simple allocation factor α_ℓ is used:

$$\sum \alpha_P \leq 1 \tag{5-33}$$

where the sum is over all boundary blocks positioned adjacent to the vertical boundaries of the system and α_ℓ measures the relative exposure of such blocks to the aquifer. Furthermore, Equations (5-31) are modified to yield

$$e_{W\ell} = a_{W\ell} + b_{W\ell}\delta p_\ell \tag{5-34a}$$

where

$$a_{W\ell} = \alpha_\ell a_W \tag{5-34b}$$

and

$$b_{W\ell} = \alpha_\ell b_W \tag{5-34c}$$

For the unsteady-state model, a variety of options are available, including several methods for automatic specification. For the steady-state aquifer model and pot-aquifer (no flow) models, the allocation factors are specified only via input.

5.2.2 STEADY-STATE AQUIFER

Just as for the case of an unsteady-state aquifer, a cylindrical geometry is assumed for the aquifer, as shown in Figure 5-2. It is further assumed that the aquifer is finite and that a pressure boundary condition is imposed at its outer extremity, $r = r_q$. If the response of the aquifer is sufficiently rapid relative to the time period of interest, then a steady-state approximation may be used. Thus, the rate of flow coming into the reservoir from the aquifer is given by

$$e_W = M_A\left(p_o - p\right) \tag{5-35}$$

where M_A is the mobility of the aquifer (= $2\pi Kh/\mu\ell n(r_e/r)$), P_o is the boundary pressure at the outer aquifer boundary r = r_q, and p is the pressure at the inner aquifer boundary r = r_e. Both Schilthius (1936) and Katz et al. (1963) have considered the model.

Before introducing Equation (5-33) into a numerical algorithm, it is rewritten in terms of two successive time levels. Furthermore, allocation to the various boundary blocks is performed using the allocation factors described in the last section. The result is:

$$e_{W\ell} = a_{W\ell} + b_{W\ell}\delta p_\ell \tag{5-36a}$$

where

$$a_{W\ell} = M_A\left(p_o - p_\ell^n\right)\alpha_\ell \tag{5-36b}$$

and

$$b_{W\ell} = -M_A\alpha_\ell \tag{5-36c}$$

In effect, each boundary block ℓ is allocated a section of the aquifer which responds only to the pressure p_ℓ of that particular boundary block. Equations (5-36) are the working equations used by SWIFT for Windows.

The code requires that the analyst prepare a table of values $M_A\alpha_\ell$ for the input. In doing this, one may, for example, choose α_ℓ to be the fractional area exposed to the aquifer, as described in the last section. For M_A, one might take the steady-state solution to Equation (5-20), which is, in dimensionless form,

$$e_{WD} = M_{AD}\left(1 - p_D\right) \tag{5-37a}$$

with

$$M_{AD} = 1/1n\left(R_q\right) \tag{5-37b}$$

The dimensionless quantities here are related to their dimensional counterparts via

$$e_{WD} = e_W / e_o \ , \ e_o = 2\pi\Delta h k p_o / \mu \tag{5-38a}$$

$$M_{AD} = M_A / M_{Ao} \ , \ M_{Ao} = e_o / p_o \tag{5-38b}$$

and Equation (5-23) for the aquifer radius.

5.2.3 POT AQUIFER (NO-FLOW)

For the steady-state-aquifer model, it is assumed that the mobility of the aquifer is sufficiently large that transient effects are of negligible importance in assessing the response of the aquifer to changes within the reservoir. For the so-called pot-aquifer model, the same assumption is made. Thus, insofar as the aquifer is concerned, both are steady-state models. The only difference between the two is the boundary condition for the external aquifer radius. For the steady-state model, a constant pressure is applied, whereas for the pot aquifer, a no-flow condition is applied.

The pot aquifer therefore acts as a unit with a total accumulation governed strictly by the flow to the reservoir, i.e. ,

$$-e_W = \left(c_W + c_R\right) V_A \frac{\partial p}{\partial t} \tag{5-39}$$

where V_A is the aquifer pore volume. Pressure p represents the pressure of the entire aquifer including, of course, the reservoir-aquifer boundary.

Finite-differencing the time derivative and allocating over the individual boundary grid blocks ℓ yields the customary form

$$e_{W\ell} = a_{W\ell} + b_W \delta p_\ell \tag{5-40a}$$

where

$$a_{W\ell} = 0 \tag{5-40b}$$

and

$$b_{W\ell} = -\alpha_\ell \left(c_W + c_R\right) V_A / \Delta t \tag{5-40c}$$

5.2.4 HEAT AND BRINE TRANSFERS BETWEEN RESERVOIR AND AQUIFER

The previous three sections have focused upon the flow between the reservoir being simulated and the aquifer surrounding it. This flow gives rise to a convective transport of enthalpy, brine, and radionuclides. For the f ormer two processes, dispersive/conductive transfers between reservoir and aquifer are also permitted. For transport directed from aquifer to reservoir, the initial conditions prescribed for the boundary block are assumed to characterize the aquifer with which it communicates. Consequently, these initial values are used explicitly, with the computed flow, to characterize the input of both energy and contaminant to the reservoir. For transport directed from reservoir to aquifer, the current conditions of the boundary block are used. These quantities are then used implicitly, in addition to the computed flow, to characterize the output of both energy and contaminant to the aquifer. The appropriate equations are developed in Sections 5.3.2, 5.3.3. and 5.3.4.

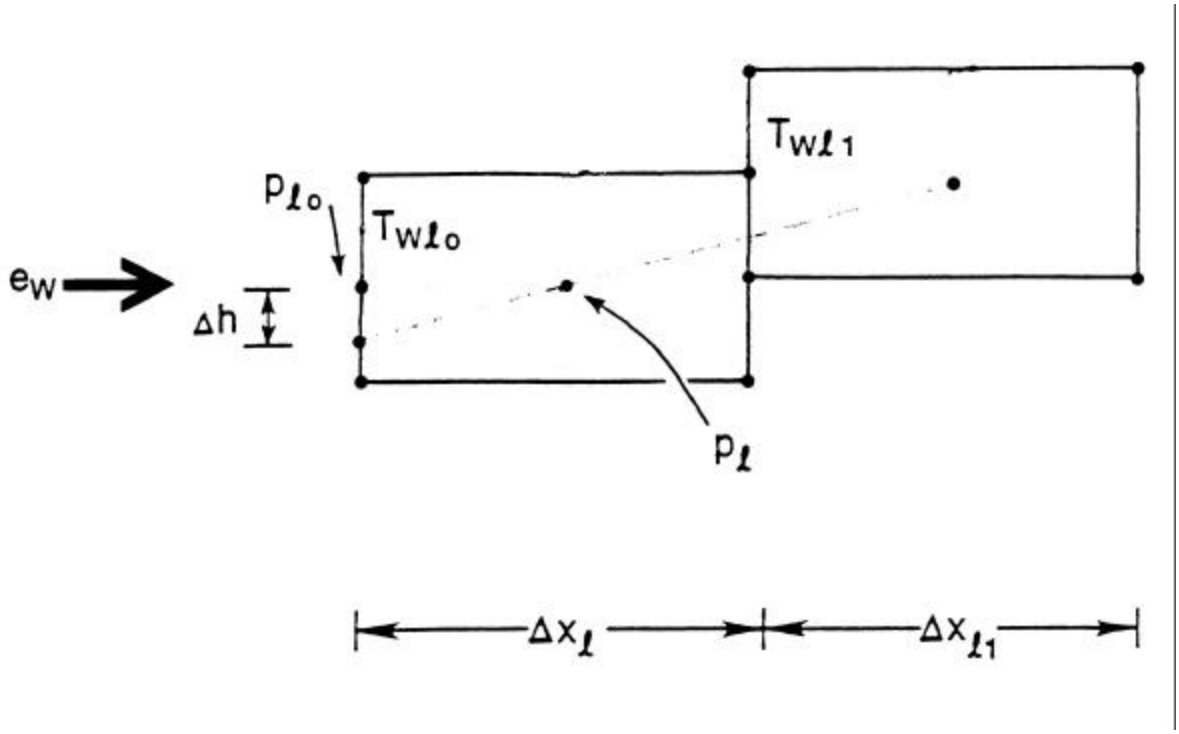
5.3 DIRICHLET BOUNDARY CONDITIONS

In Section 5.2 several aquifer models are presented. By using one of these models, one may specify reservoir boundary conditions indirectly in terms of the state of the aquifer. In the case of Dirichlet conditions, one specifies directly the values p_o, T_o, and C_o for the dependent variables at the center position of a block face. The face-centered specified dependent variable is connected to the block centered calculated variable by a transmissibility.

5.3.1 FLOW

To understand the manner in which Dirichlet conditions are applied, it is instructive to consider the boundary block ℓ shown in Figure 5.3 for which the applied boundary pressure is $P_{\ell o}$. As illustrated, this pressure is applied along the grid block at the same elevation as the grid-block node. If, for example, the presence of a river of surface elevation Δh below that of the grid block were being simulated, so that the pressure ($p = 0$) were known at that elevation, the prescribed boundary pressure would still need to be corrected to the center of the grid block ($p_{\ell o} = \rho g \Delta h / g_c$). The net flow may be expressed as

Figure 5.3. Dirichlet Boundary Conditions for Boundary Block ℓ.



$$e_{W\ell} = T_{W\ell o}\left(p_{\ell o} - p_{\ell}\right) \tag{5-41}$$

a variant of Darcy's law. Quantity $T_{W\ell o}$ is the transmissibility, which in general is defined as in Sections 6.1.1 and 6.1.2. Here, however, for the bounding face it is obtained in terms of the next interior transmissibility, i.e.,

$$T_{W\ell o} = \left(\Delta x_{\ell} + \Delta x_{\ell 1}\right) T_{W\ell 1} / \Delta x_{\ell} \tag{5-42}$$

This relation assumes that permeabilities and thicknesses are the same for Blocks 1 and 2. It is also important the next interior transmissibility be nonzero. The direct in assigning the next interior transmissibility is user defined by the face type index (see Read R1-28, Reeves et al., 1986a. p. 39).

Applying the time-weighting procedure (Section 5.1.4), Equation (5-41) becomes

$$e_{W\ell} = a_{W\ell} + b_{W\ell}\delta p_{\ell} \tag{5-43a}$$

Here δp_{ℓ} is the incremental pressure change over time step Δt,

$$a_{W\ell} = T_{W\ell o}\left(p_{\ell o} - p_{\ell}^{n}\right) \tag{5-43b}$$

with n denoting the previous time level, and

$$b_{W\ell} = -T_{W\ell o} \tag{5-43c}$$

Equations (5-43) are the working equations for Dirichlet pressure boundary conditions.

5.3.2 HEAT TRANSPORT

Here two mechanisms must be accounted for, namely conduction/dispersion and convection. In addition, a distinction is made between outflow ($e_{W\ell} < 0$) and inflow ($e_{W\ell} \geq 0$). The basic equation for the ℓ-th grid block is

$$e_{H\ell} = e_{W\ell} H_{\ell o} + T_{H\ell o}\left(T_{\ell o} - T_{\ell}\right) \tag{5-44}$$

Analysis of the conduction/dispersion term parallels that of fluid conduction. However, the convection term is expanded in two different ways, depending on the direction of flow. For flow directed into the grid block the enthalpy is evaluated at the boundary face

$$H_{\ell o} = H\left(T_{\ell o}, p_{\ell o}\right) \ , \ e_{W\ell} > 0 \tag{5-45a}$$

where the functional relation is given by Equations (2-13) and (2-14). For both boundary conditions specified with the constant-flux initial condition (Section 5.1.1) and aquifer-influence boundary conditions (Section 5.2), inwardly-directed ($e_W > 0$) transfers of brine are determined from the specified initial conditions. For flow directed out of the grid block, the enthalpy at the boundary is approximated by its value at the node:

$$H_{\ell o} = H_{\ell}\left(T_{\ell}, P_{\ell}\right) \ , \ e_{W\ell} < 0 \tag{5-45b}$$

At the time level n+w (see Section 6.14), which is appropriate for the numerical algorithm, the latter equation is evaluated by the approximation

$$H_{\ell} = H_{\ell}^{n} + wc_{\rho}\delta T_{\ell} \tag{5-46}$$

where w is the time weighting, c_p is the heat capacity and δT_{ℓ} is the change over the increment Δt.

Substituting Equations (5-45) and (5-46) into Equation (5-44) and writing the temperatures at time level n+w, the working equations are obtained, namely,

$$e_{H\ell} = a_{H\ell} + wb_{H\ell}\delta T_{\ell} \tag{5-47a}$$

where

$$a_{H\ell} = \begin{cases} e_{W\ell}H_{\ell o} + T_{H\ell o}T_{\ell o} - T_{\ell}^{n} & , \ e_{W\ell} \geq 0 \\ e_{W\ell}H_{\ell}^{n} + T_{H\ell o}T_{\ell o} - T_{\ell}^{n} & , \ e_{W\ell} < 0 \end{cases} \tag{5-47b}$$

and

$$b_{H\ell} = \begin{cases} -T_{H\ell o} & , \ e_{W\ell} \geq 0 \\ -T_{H\ell o} + e_{W\ell}c_p & , \ e_{W\ell} < 0 \end{cases} \tag{5-47c}$$

5.3.3 BRINE TRANSPORT

In this case the basic equation for the mass flow through the boundary of the ℓ-th grid block is

$$e_{C\ell} = a_{C\ell} + wb_{C\ell}w\hat{C}_{\ell} \tag{5-48a}$$

where

$$a_{C\ell} = \begin{cases} e_{W\ell}\hat{C}_{\ell o} + T_{C\ell o}\left(\hat{C}_{\ell o} - \hat{C}_{\ell}^{n}\right) , & e_{W\ell} \geq 0 \\ e_{W\ell}\hat{C}_{\ell o} + T_{C\ell o}\left(\hat{C}_{\ell o} - \hat{C}_{\ell}^{n}\right) , & e_{W\ell} < 0 \end{cases} \tag{5-48b}$$

and

$$b_{C\ell} = \begin{cases} -T_{C\ell o} , & e_{W\ell} \geq 0 \\ -T_{C} + e_{W}\hat{C} , & e_{W\ell} < 0 \end{cases} \tag{5-48c}$$

5.3.4 RADIONUCLIDE TRANSPORT

For the case of radionuclide transport, the well-type conditions are emphasized, and only a simple convection-type boundary condition is permitted. The appropriate relation is

$$e_{r\ell} = e_{W\ell}C_{r\ell 0} \tag{5-49}$$

For flow directed into the system, the boundary concentration is assumed to be negligible. For flow directed out of the system, this concentration is approximated by its value at node ℓ. Hence,

$$e_{r\ell} = a_{r\ell} + wb_{r\ell}\delta C_{r\ell} \tag{5-50a}$$

where

$$a_{r\ell} = \begin{cases} 0 , & e_{W\ell} \geq 0 \\ e_{W\ell}C_{r\ell}^{n} , & e_{W\ell} < 0 \end{cases} \tag{5-50b}$$

and

$$b_{r\ell} = \begin{cases} 0 , & e_{W\ell} \geq 0 \\ e_{W\ell} , & e_{W\ell} < 0 \end{cases} \tag{5-50c}$$

5.4 HEAT LOSS TO OVER/UNDERBURDEN

Typically, the over/underburden regions will consist of aquitards which are effectively isolated hydrologically from the reservoir. Nevertheless, these zones may contribute significantly to the heat transport from a heat source such as a repository. Over and

underburden zones are therefore included in SWIFT for Windows via boundary conditions for the purpose of simulating only the heat transport. The technique used is similar to that for the unsteady aquifer in that a boundary zone is considered. Here the zone is modeled by the relation

$$K_m \frac{\partial^2 T}{\partial z^2} + q_H = \rho_R c_{pR} \frac{\partial T}{\partial t} \tag{5-51}$$

This is a simple heat-conduction equation in which all transport and storage effects within the fluid are neglected (c.f., Equation (2-2)). In addition, transport in the lateral x and y directions is ignored in order to limit storage requirements, and a subscript ℓ is added to indicate the boundary block.

A table of initial temperatures $T_o(z)$ versus depth is required as code input (T_o = TD and Z = ZT in Reeves et al. , 1986a, p. 26). This table provides an initial condition and one boundary condition:

$$T(z, t = 0) = T_o(z) \tag{5-52a}$$

$$T(z = L, t) = T_o(L) \tag{5-52b}$$

where L is taken to be a sufficiently large distance into the over or underburden that temperature changes there from the ambient temperature may be neglected. The other boundary condition is obtained by matching with the reservoir temperature, which, for one particular boundary block, is denoted by T_ℓ:

$$T(z = 0, t) = T_\ell(t) \tag{5-52c}$$

Here, for simplicity, the interface is taken to be the plane z = 0.

The set of equations, Equations (5-51) and (5-52), is solved by a rather unique application of the finite difference method. Dependent variable T is divided into two components, namely

$$T = T_1 + wT_2\delta T_\ell \tag{5-53}$$

Here, temperature T_1 evolves, as a particular solution of Equation (5-51), strictly from a redistribution of the temperature at the n-th time step. The initial-boundary conditions for the step from time t_n to time t_{n+1} are, in this case, given by

$$T_1\left(z, t = t_n\right) = T\left(z, t = t_n\right) \tag{5-54a}$$

and

$$T_1\left(z = 0, t\right) = T\left(z = 0, t = t_n\right) \tag{5-54b}$$

Dimensionless temperature T_2 then evolves strictly from a change in boundary conditions appropriate for the n-th time level:

$$T_2\left(z, t = t_n\right) = 0 \tag{5-55a}$$

and

$$T_2\left(z = 0, t\right) = 1 \tag{5-55b}$$

As shown in Equation (5-53), the dimensionless temperature is normalized by the time-weighting factor (see Section 5.1.4), and the temperature change over Δt.

The unique feature of the superposition specified by Equation (5-53) is the fact that it permits implicit coupling between solution of the over/underburden transport, Equation (5-51), and the solution of the reservoir transport (Equation (2-2)). Equation (5-53) may be differentiated and combined with appropriate factors to yield the heat loss to the over/underburden region:

$$e_{L\ell} = a_{L\ell} + w b_{L\ell} \delta T_\ell \tag{5-56}$$

Here $a_{L\ell}$ is obtained f rom T_1 and $b_{L\ell}$ is obtained from T_2, and a subscript ℓ has been added throughout to indicate the boundary block.

5.5 RADIATION BOUNDARY CONDITIONS

A body at absolute temperature T_o surrounded by a black body at temperature θ_o will gain heat with a flux in accordance with Stefan's law:

$$e_h = \bar{\sigma}\bar{\varepsilon}\left(\theta^4 - T_o^4\right) \tag{5-57}$$

where e_h is the flux, $\overline{\sigma}$ is the Stefan-Boltzmann constant, and ϵ is the emissivity of the surface. If the temperature difference is not too great, then Equation (5-57) may be replaced by the expression

$$e_R = \beta\left(\theta_o - T_o\right) \tag{5-58}$$

where β is the coefficient of surface heat transfer.

It is because of the connection between Stefan's Law, Equation (5-57), and Equation (5-58) that the latter has been called the "radiation boundary condition" (Carslaw and Jaeger, 1959). However, Equation (5-58) is identical to Newton's Law of Cooling for forced convection. It is also appropriate for a relatively thin skin placed between the conducting media being simulated and a constant-temperature surface. Mathematically, it is simply a type-three boundary condition.

In order to implement Equation (5-58), the radiative transfer occurring at the boundary is assumed to be balanced by conductive/dispersive transport just inside the boundary. Thus, for a particular boundary block ℓ.

$$e_{R\ell} = \beta_\ell\left(\theta_{\ell o} - T_{\ell o}\right) = T_{H\ell}\left(T_{\ell o} - T_\ell\right) \tag{5-59}$$

Solving for the boundary temperature, one obtains

$$T_{\ell o} = \left(\beta_\ell \theta_{\ell 0} + T_H T_\ell\right)\Big/\left(T_{H\ell} + \beta_\ell\right) \tag{5-60}$$

which relates the boundary temperature to the grid-block temperature. Expanding about time level n, one obtains further

$$T_\ell = T_\ell^n + w\delta T \tag{5-61}$$

where w is the weighting factor (see Section 6.1.4).

The final working equations are obtained by combining Equations (5-59) through (5-61). The result is

$$e_{R\ell} = a_{R\ell} + b_{R\ell}\delta T \tag{5-62a}$$

where

$$a_{R\ell} = \theta_{\ell o} T_{H\ell} - T_{H\ell} T_\ell^n \beta_\ell\left(T_{H\ell} + \beta_\ell\right) \tag{5-62b}$$

and

$$b_{R\ell} = -T_{H\ell}\beta_{\ell} / \left(T_{H\ell} + \beta_{\ell}\right) \tag{5-62c}$$

The analyst must supply the temperature θ_o (program variable T2 [Reeves et al., 1986a, p. 40]) and the coefficient of surface heat transfer β (program variable T3 [Reeves et al., 1986a, p. 40]).

5.6 FREEWATER SURFACE

In the simulation of shallow aquifer systems, it is often desirable to include the effects of the freewater surface, also referred to as water table conditions. This may be accomplished by allowing finite-difference blocks to drain (dewater), resulting in reduced block-to-block transmissibilities and release of water in accordance with the drainage capacity of the block.

In the code, pressures are evaluated at block centers. The position of the freewater surface is determined by the pressure. The block saturation is the fractional amount of the block below the water table. For a particular block. the position is calculated relative to reference pressure as:

$$S_{fw} = \frac{p^n - p_O}{\rho \Delta z \left(g / g_c\right)} + 0.5 \tag{5-63}$$

In other words, for the water table to be at the top of the block,

$$S_{fw} = 1 \tag{5-64a}$$

or

$$p^n = p_O + 1/2 \rho \Delta z (g / g_c) \tag{5-64b}$$

and for the water table to be at the bottom of the block

$$S_{fw} = 0 \tag{5-65a}$$

or

$$p^n = p_0 - 1/2 \rho \Delta z (g / g_c) \tag{5-65b}$$

The amount of water contained by storage in a partially saturated freewater block is:

$$V_{fw} = \Delta x \Delta y \Delta z \theta_o \tag{5-66a}$$

or

$$V_{fw} = S_{fw} V \tag{5-66b}$$

Otherwise for blocks that are fully saturated

$$V = \Delta x \Delta y \Delta z \theta \left[1 + C_r \left(p - p_0 \right) \right] \tag{5-67}$$

The freewater option allows for dewatering in all blocks, not just the uppermost layer (e.g., SWIP, INTERCOMP, 1976). The approach is consistent with other codes such as the modular U.S. Geological Survey code (McDonald and Harbaugh, 1984). Details concerning the implementation are presented in Section 6.6.

6 NUMERICAL IMPLEMENTATION

Chapters 2 through 5 focus on the mathematical implementation of the SWIFT for Windows code model. This chapter introduces the numerical implementation. Here, the object is to give the analyst sufficient information that the code can be used knowledgeably, but without the burden of the details involved in deriving the working equations. Such details are reserved for the appendices.

6.1 TRANSIENT FLOW, HEAT, AND BRINE TRANSPORT

Equations (2-1) through (2-3) are discretized for a particular grid block m = (i, j, k) in the following manner:

$$\overline{R}_W = \delta\left(V\rho\right) / \Delta t \tag{6-1a}$$

$$\overline{R}_H = \delta\left(V\rho U + V_R\rho_R U_R\right) / \Delta t \tag{6-1b}$$

$$\overline{R}_C = \delta\left(V\rho\hat{C}\right) / \Delta t \tag{6-1c}$$

where the notation is given in a separate section.

The accumulation of fluid, heat, and brine is represented by the right-hand sides of Equation (6-1). For the heat equation, it should be noted that there are two such terms. One represents an accumulation of internal energy in the fluid and the other an accumulation in the rock. Using Equation (2-15), the term for the rock is expanded relative to the reference temperature as follows:

$$V_R\rho_R U_R = V\rho_R\overline{c}_{pR}\left(T - T_o\right) \tag{6-2}$$

The heat capacity $\rho_R\overline{c}_{pR}$ is expressed as the heat gained per degree temperature rise per unit fluid volume via

$$\overline{c}_{pR} = c_{pR}\left(1-\phi\right) / \phi \tag{6-3}$$

The various transport and source-sink terms are contained in the quantities R on the left-hand side of Equation (6-1). They are defined as follows:

$$\overline{R}_W = \Delta_\ell\left[T_{W\ell m}\left(\Delta_m p - \rho g \Delta_m z / g_c\right)\right] - \overline{Q}_W - \Gamma_W + A_W V\rho\left(1-\hat{C}\right) \qquad (6\text{-}4a)$$

$$\overline{R}_H = -\Delta_\ell\left(Hu_\ell\right) + \Delta_\ell\left(T_{H\ell m}\Delta_m T\right) - \overline{Q}_H - \Gamma_H \qquad (6\text{-}4b)$$

$$\overline{R}_C = -\Delta_\ell\left(\hat{C}u_\ell\right) + \Delta_\ell\left(T_{C\ell m}\Delta_m \hat{C}\right) - \overline{Q}_C - \Gamma_C + A_C V\rho\left(1-\hat{C}\right) \qquad (6\text{-}4c)$$

The summation convention is used for the three spatial coordinates and is indicated by repeated indices. Also, all source/sink and boundary conditions, except for dualporosity and salt-dissolution terms, have been coalesced in the terms $\overline{Q}_X$. In addition, since Equations (6-1) and (6-4) are grid-block specific, some of the units are slightly different than in Equations (2-1) to (2-3). The quantity Γ, for example, is given in units of mass per time in the above equations rather than in units of mass per bulk volume per time. Similarly, Darcy flow u now has units of mass per time rather than volume per area per time.

6.1.1 FLOW TRANSMISSIBILITIES FOR A CARTESIAN SYSTEM

The transmissibility is generally a tensor of rank two. For the flow, however, T_W may be assumed to be diagonal without any loss of generality:

$$T_{W\ell m} = 0 \ , \ \ell \neq m \qquad (6\text{-}5a)$$

Index ℓ (or m) denotes one of the three coordinate directions, x, y, or z. The diagonal components are defined as follows:

$$T_{Wxx,i+1/2,jk} = \frac{2\Delta y_j}{\left(\Delta x / k_x \Delta z_k\right)_i + \left(\Delta x / k_x \Delta z_k\right)_{i+1}}\left(\rho g / \mu\right)_{i+1/2,jk} \qquad (6\text{-}5b)$$

and

$$T_{Wzz,ij,k+1/2} = \frac{2\Delta x_i \Delta y_j}{\left(\Delta x / k_z\right)_k + \left(\Delta z / k_z\right)_{k+1}}\left(\rho g / \mu\right)_{ij,k+1/2} \qquad (6\text{-}5c)$$

Component $T_{W_{yy}}$ has a form which is completely analogous to $T_{W_{xx}}$.

Since the term transmissibility refers to an interface between grid blocks, an average is implied. For the SWIFT model, an algebraic average is used for both density and viscosity, whereas a weighted harmonic average is used for the permeability. In addition, the thickness Δz, in Equation (6-5b), is permitted to vary not only as a function of the depth index k but also as a function of lateral position i (and j).

6.1.2 FLOW TRANSMISSIBILITIES FOR A CYLINDRICAL SYSTEM

Equations (6-5b) and (6-5c), which apply to a Cartesian coordinate system, have the form

$$T_W = (\Delta\sigma / \Delta)\left(k\rho g / \mu\right) \tag{6-6}$$

where $(k\rho g/\mu)$ is the hydraulic conductivity, $\Delta\sigma$ is the cross-sectional area, and Δ is the node-to-node distance between neighboring grid blocks.

For a cylindrical (r-z) coordinate system, this same basic form is maintained. For flow in the radial direction, Equation (6-5b) is replaced by the relation

$$T_{Wrr,i+1/2,k} = \left[2\pi r_{i+1/2}\Delta z_k \big/ \left(r_{i+1} - r_i\right)\right]\left(k\rho g / \mu\right)_{i+1/2,k} \tag{6-7a}$$

For flow in the z direction, an equation very much like Equation (6-5c) is used, namely,

$$T_{Wzz,i,k+1/2} = \frac{2\pi\left(r_{i+1/2}^2 - r_{i-1/2}^2\right)}{\left(\Delta z / k_z\right)_k + \left(\Delta z / k_z\right)_{k+1}}\left(\rho g / \mu\right)_{i,k+1/2} \tag{6-7b}$$

Quantities of the form $r_{i+1/2}$ represent the grid-block interfaces. They are taken to be the logarithmic average

$$r_{i+1/2} = \left(r_{i+1} - r_i\right) \big/ 1n\left(r_{i+1} / r_i\right) \tag{6-8}$$

Such a choice, according to Aziz and Settari (1979), results in a consistent finite-difference approximation which will yield the exact pressure drop at steady state.

6.1.3 TRANSPORT TRANSMISSIBILITIES

The transport transmissibilities, in general, are not diagonal due to their dependence upon the dispersion tensor D, c.f., Equation (2-7). The SWIFT for Windows code, however, uses an approximate procedure which results in either total or partial diagonalization, as specified by the user. Components in the vertical x-z and y-z planes are always lumped with their respective diagonals, as shown in the equations below:

$$\overline{E}_{Xxx} = E_{Xxx} + E_{Xxz} + \zeta E_{Xxy} \tag{6-9a}$$

$$\overline{E}_{Xzz} = E_{Xzz} + E_{Xzx} + E_{Xzy} \tag{6-9b}$$

$$\overline{E}_{Xyy} = E_{Xyy} + E_{Xyz} + \zeta E_{Xyx} \tag{6-9c}$$

Here

$$\zeta = \begin{cases} 0 \ , & \text{MINITN} \geq 1 \\ 1 \ , & \text{MINITN} = 1 \end{cases} \tag{6-10}$$

where MINITN is an input parameter specifying the minimum number of iterations to be used (Reeves et al., 1986a, p. 75). Subscript X denotes either heat (X = H) or brine (X = C). The x-y and y-z components are either lumped with their respective diagonal components or included directly. Since the latter procedure uses iterations, the code requires that the minimum number of iterations be set at a value greater than unity to include directly the cross terms which arise from the x-y and y-x components.

In any event, the lumped diagonal transmissibility components for a Cartesian system are determined in a manner quite similar to that of Equation (6-5), namely

$$T_{X\ell z} = 0 \ , \ \ell = x \text{ or } y \tag{6-11a}$$

$$T_{Xxx,i+1/2,jk} = \frac{2\Delta y_j}{\left(\Delta x / \overline{E}_{Xxx}\Delta z_k\right)_i + \left(\Delta x / \overline{E}_{Xyy}\Delta z_k\right)_{i+1}} \tag{6-11b}$$

and

$$T_{Xzz,ij,k+1/2} = \frac{2\Delta x_i \Delta y_j}{\left(\Delta z / \overline{E}_{Xzz}\right)_{k+1}} \tag{6-11c}$$

and component T_{Xyy} is determined in an analogous manner to that of T_{Xxx}. The x-y and y-x cross terms are examined in Appendix A.

For a cylindrical system, the fully diagonalized tensor is always used and the transmissibility components are determined in an analogous manner to that of Equation (6-7).

6.1.4 TIME WEIGHTING

Equations (6-1) and (6--4) provide an algorithm for stepping from time step n to time step n+l. It is understood there that the quantities R are evaluated at an intermediate level n + w, where w is the time-weighting parameter defined as

$$w = \begin{cases} 1/2 \ , \ \text{centered} - \text{in} - \text{time (CIT)} \\ 1 \quad , \ \text{backward} - \text{in} - \text{time (BIT)} \end{cases} \tag{6-12}$$

This parameter is used to evaluate the various variables entering into Equation (6-4) using the prescription

$$\Psi^{n+w} = \Psi^{n} + w\delta\Psi \tag{6-13}$$

where ψ denotes a general time-dependent variable.

There are two qualifications which should be made here. First, transmissibilities are always evaluated at time-level n. This is a part of the chosen algorithm for handling the nonlinearity of the flow and transport equations. Second, the time weighting is always chosen to be w = 1 for the flow equation. The flow equation has no convection term and, hence, no numerical dispersion. Thus, there is no need to choose centered differencing.

6.1.5 SPACE WEIGHTING

The need for space weighting arises through the convection term. To illustrate this, the x-component of this term is expanded for the brine equation (Equation 6-4c):

$$\Delta_x\left(\hat{C}u_x\right) = \left(\hat{C}u_x\right)_{i+1/2} - \left(\hat{C}u_x\right)_{i-1/2} \tag{6-14}$$

where several subscripts and superscripts have been suppressed for the sake of clarity. The point is this: The evaluation of $\hat{C}$ is required at the gridblock interfaces. Since this quantity is calculated only at the grid-block centers, interpolation is clearly called for. This is done by means of a user-supplied space weighting parameter

$$\gamma = \begin{cases} 1/2 \ , \ \text{centered} - \text{in} - \text{space (CIS)} \\ 1 \quad , \ \text{backward} - \text{in} - \text{space (BIS)} \end{cases} \tag{6-15}$$

The interpolate is

$$\Psi_{i-1/2} = \Omega\,\Psi_i + (1-\Omega)\Psi_{i-1} \tag{6-16}$$

where

$$\Omega = \frac{(1-\bar{\gamma})\Delta x_{i-1}}{\bar{\gamma}\Delta x_i + (1-\bar{\gamma})\Delta x_{i-1}} \tag{6-17}$$

Parameter $\bar{\gamma}$ is determined both by the value of γ and by the direction of the Darcy velocity:

$$\bar{\gamma} = \begin{cases} \gamma \quad , \ u_{i-1/2} > 0 \\ 1-\gamma \ , \ u_{i-1/2} < 0 \end{cases} \tag{6-18}$$

Thus, for backward-in-space differencing,

$$\Psi_{i-1/2} = \begin{cases} \Psi_{i-1} \ , \ u_{i-1/2} > 0 \\ \Psi_i \quad , \ u_{i-1/2} < 0 \end{cases} \quad \text{(BIS)} \tag{6-19}$$

Furthermore, for centered-in-space differencing with equal increments

$$\Psi_{i-1/2} = \left(\Psi_{i-1} + \Psi_i\right)/2 \ , \ \Delta x_{i-1} = \Delta x_i \quad \text{(CIS)} \tag{6-20}$$

6.1.6 SOLUTION TECHNIQUES

Appendix A presents the detailed numerical development of the working equations which are implemented in the SWIFT for Windows code. An overview is given here. Basically, there are three coupled finite-difference equations given in Equations (6-1) and (6-4). The linear coupling present in the accumulation and source/sink terms is treated via the technique of formal Gaussian elimination. All remaining nonlinear coupling is treated via the technique of iteration. There is one exception, however, with regard to the transmissibility. This quantity is lagged by one time step to eliminate instabilities which might arise from its nonlinearity. After expanding the various difference operators appearing in Equations (6-1) and (6-4) and using the time- and space-weighting procedures discussed above, a set of three coupled matrix equations is obtained. Starting with the flow equation and proceeding to the heat and brine, these three equations are solved sequentially within an interactive loop until convergence is attained. Either direct or two-line successive over-relaxation methods are implemented within the code.

6.1.7 CONVERGENCE

Nonlinearity arises in the SWIFT for Windows code through four terms, namely, the density, the pore volume (porosity), the viscosity, and the cross terms deriving from the nondiagonal transport transmissibilities. The dependence on viscosity is treated by lagging by one time step the evaluation of the transmissibilities. Furthermore, of the remaining terms, the density is assumed to be the most sensitive indication of changes in the dependent variables. Therefore, the density is used for the development of the convergence criteria.

For a given iteration it+1, the relative changes in density due to temperature and concentration changes since iteration, it, are given by

$$\varepsilon_T = \max_m \left[c_T \left(T_m^{it+1} - T_m^{it} \right), \varepsilon'_{\rho Tm} \right] \tag{6-21a}$$

and

$$\varepsilon_C = \max_m \left[c_C \left(\hat{C}_m^{it+1} - \hat{C}_m^{it} \right), \varepsilon'_{\rho Cm} \right] \tag{6-21b}$$

Here, ε'_m is the maximum deviation within the rock matrix (see Equation (6-39) below and subscript m ranges over all grid blocks).

These relative changes in density are used in following manner:

1. If the total change is less than the prescribed maximum, i.e.,
$$\varepsilon_T + \varepsilon_C \le \tau_L \quad (6\text{-}22)$$
then the solution is said to have converged. The tolerance is set at $\tau_L = 0.001$.

2. However, if Condition (6-22) is not satisfied, then the following two tests are made:
$$\varepsilon_T \le \tau_L / 2 \quad (6\text{-}23)$$
and
$$\varepsilon_T \le \tau_L / 2 \quad (6\text{-}24)$$

3. If neither Condition (6-23) nor Condition (6-24) is satisfied, then all three equations for pressure. temperature, and brine concentration are solved again and a similar set of convergence criteria is applied.

4. If Condition (6-23) is satisfied, then only the brine and pressure equations are solved on the following iterate.

5. If Condition (6-24) is satisfied, then only the temperature and pressure equations are solved on the following iterate.

The convergence criteria above are specialized in a straightforward manner if fewer than three principal equations are solved. However, it should be noted that the above criteria do not apply to the case of pressure solution only. There, the only iteration control is the specification of the minimum and the maximum number of iterations.

6.2 RADIONUCLIDE TRANSPORT

The finite-difference equation for grid block m = (i, j, k) corresponding to Equation (2-4) is

$$\overline{R}_r = \delta\left(V\rho C_r + V_R\rho_R W_r\right) / \Delta t \quad (6\text{-}25a)$$

where

$$\overline{R}_r = -\Delta_\ell\left(C_r u_\ell\right) + \Delta_\ell\left(T_{C\ell m}\Delta_m C_r\right) - \overline{Q}_r - \Gamma_r + \sum_s k_{rs}\lambda_s\left(V\rho C_s + V_R\rho_R W_s\right) - \lambda_r\left(V\rho C_r + V_R\rho_R W_r\right) \tag{6-25b}$$

For simplicity, all source-sink terms, other than the fracture-matrix interaction Γ_r, are coalesced into the single term Q_r. Equation (6-25a) shows an accumulation of radionuclide r in both liquid (C_r) and solid phases (W_r). These two phases are connected by the Freundlich isotherm of Equation (2-8), namely

$$W_r = \kappa_r\left(\rho C_r\right)^{\eta_r} \tag{6-26}$$

6.2.1 SOLUTION TECHNIQUES

Appendix B presents a detailed development in which the time and space-weighting procedures are used and the difference operators are expanded to obtain the working equation. This equation may be solved either in tandem with the solution of the primary equations for pressure, temperature and brine for a fully transient calculation, or they may be solved in a stand-alone fashion following steady-state solution of the primary equations. In either case, many of the parameters, such as transport transmissibilities T_C, Darcy velocity u, pore volume V, and density ρ, are predetermined from the primary-equation solution and serve as purely linear parameters for the radionuclide-transport solution. This means that the only nonlinearity present in this case arises through the Freundlich isotherm, for which an iteration loop is applied. Here, the isotherm is linearized by a Taylor-series expansion about $C_r = 0$, and the expansion coefficient is updated on each pass through the loop.

6.2.2 CONVERGENCE

For a given iteration it+1, the change in radionuclide concentration relative to iteration, it, is defined as

$$\varepsilon_r = \max_m\left[\left(C_{rm}^{it+1} - C_{rm}^{it}\right)\Big/ C_{rm}^{it+1}\right] \tag{6-27}$$

where m is the grid-block index. Convergence is achieved whenever

$$\varepsilon_r \le t_L \tag{6-28}$$

where the tolerance is $\tau_L = 0.001$. The maximum number of iterations is set at 15.

6.3 STEADY-STATE FLOW AND BRINE TRANSPORT

Equations (2-17) and (2-18) are discretized for a particular grid block m = (i, j, k) in the following manner:

$$\Delta_\ell\left[T_{w\ell\ell}\left(\Delta_\ell p - \rho g \Delta_\ell z / g_c\right)\right] - \overline{Q}_W + A_W V \rho\left(1 - \hat{C}\right) = 0 \tag{6-29a}$$

and

$$-\Delta_\ell\left(\hat{C} u_\ell\right) + \Delta_\ell\left(T_{C\ell\ell}\Delta_\ell \hat{C}\right) - \overline{Q}_C + A_C V \rho\left(1 - \hat{C}\right) = 0 \tag{6-29b}$$

All source/sink and boundary conditions, except for salt dissolution terms, have been coalesced in the terms Q_X. Salt-dissolution terms appear in both flow and brine equations. The coefficients given therein arise from Equations (3-3) and (3-4). They are:

$$A_C = k_C f_C \tag{6-30a}$$

and

$$A_W = A_C c_C / \left(1 + c_C\right) \tag{6-30b}$$

where c_C is determined from the input-density range by Equation (2-16).

6.3.1 SOLUTION TECHNIQUE

Appendix C presents a detailed development of the working equations which are employed in the SWIFT for Windows code. Here, however, it is sufficient just to note the general solution procedure. Equations (6-29) constitute a set of coupled nonlinear finite-difference equations. Starting with the flow and then proceeding to the brine transport, these equations are solved sequentially within an iteration loop. Densities and pore volumes are updated within each iteration, as in the transient solution of the primary equations. However, in contrast to the transient solutions, the transmissibilities are also updated as a function of the iteration index.

6.3.2 CONVERGENCE

For a given iteration it+1, the changes in pressure and brine concentration relative to iteration it, are defined as

$$\varepsilon_W = \max_m \left[\left(p_m^{it+1} - p_m^{it} \right) \Big/ p_m^{it+1} \right] \tag{6-31a}$$

$$\varepsilon_C = \max_m \left[\left(\hat{C}_m^{it+1} - \hat{C}_m^{it} \right) \Big/ \hat{C}_m^{it+1} \right] \tag{6-31b}$$

where m is the grid-block index. Convergence is achieved whenever

$$\varepsilon_W \le \tau_L \quad \text{and} \quad \varepsilon_C \le \tau_L \tag{6-31c}$$

where the tolerance is the same as that used for the transient solutions, i.e., $\tau_L = 0.001$.

There are both lower and upper limits set for the number of iterations. These limits default to one and five, respectively, but may be controlled by the analyst (variables MINITN and MAXITN [Reeves et al., 1986a, input p. 75]).

6.4 FLOW, HEAT AND BRINE TRANSPORT WITHIN THE ROCK MATRIX

The gridding within the rock matrix is discussed in Chapter 7. Suffice it to say here, however, that the matrix within each global grid block is partitioned into either prismatic or spherical units. A representative of such unit is then gridded for numerical solution of the flow and transport equations. This grid is called the local grid as opposed to that for the fracture flow and transport. The latter is called the global grid.

For the local grid, the primary equations, Equations (2-19) - (2-21) may be discretized for grid block i in the following manner:

$$\overline{R}'_W = \delta\left(V'\rho'\right) / \Delta t \tag{6-32a}$$

$$\overline{R}'_H = \delta\left(V'\rho'U' + V'_R \rho_R U'_R\right) / \Delta t \tag{6-32b}$$

$$\overline{R}'_C = \delta\left(V'\rho'\hat{C}'\right) / \Delta t \tag{6-32c}$$

where the notation is given in a separate section. The right-hand sides of Equations (6-32 a-c) contain the accumulation and the quantities R on the left-hand sides contain the transport terms. They are defined as follows:

$$\overline{R}'_W = \Delta\left(T'_W \, \Delta p'\right) + \Gamma'_W \tag{6-33a}$$

$$\overline{R}'_H = -\Delta\left(H' u'\right) + \Delta\left(T'_H \, \Delta T'\right) + H' \Gamma'_W + \Gamma'_H \tag{6-33b}$$

$$\overline{R}'_C = -\Delta\left(\hat{C}' u'\right) + \Delta\left(T'_C \, \Delta \hat{C}'\right) + \hat{C}' \Gamma'_W + \Gamma'_C \tag{6-33c}$$

6.4.1 TRANSMISSIBILITIES

Because of the one-dimensional treatment of the rock matrix and because of the constant properties of the matrix within a global grid block, the transmissibilities here may be defined in a somewhat simpler fashion than in Sections 6.1.1 and 6.1.3. They are:

$$T_{Xi} = \begin{cases} \left(4\pi s_{i-1/2}^2 \Delta s_{i-1}\right)\eta'_A & , \text{ sphere} \\ \left(1 / \Delta s_{i-1}\right)\eta'_A & , \text{ prism} \end{cases} \tag{6-34}$$

Here, $s_{i-1/2}$ denotes the lower boundary of matrix gridblock i and Δs_{i-1} denotes the distance from node i -1 to node i. The representative prism is taken to have unit cross section as evidenced by the numerator of Equation (6-34b).

The transport parameter q, is given by

$$\eta'_X = \begin{cases} g\rho' k'/\mu' & , \; X = W \\ \rho' E'_H & , \; X = H \\ \rho' E'_C & , \; X = C \end{cases} \tag{6-35}$$

where X denotes the transport process.

6.4.2 FRACTURE/MATRIX SOURCE TERMS

The source terms Γ' given in Equations (6-33) represent the flow and transport from fracture to one spherical or prismatic unit of the rock matrix. They are related to the fracture loss terms Γ for the global system by the number (M_o) of such units within the global grid block, i.e,.

$$\Gamma_W = M_o \Gamma'_W \tag{6-36a}$$

$$\Gamma_H = M_o\left(H\Gamma'_W + \Gamma'_H\right) \tag{6-36b}$$

$$\Gamma_C = M_o\left(\hat{C}\Gamma'_W + \Gamma'_C\right) \tag{6-36c}$$

where the fracture sink terms Γ are referenced to the global grid block m and the matrix quantities are referenced to n_s, the index of the boundary grid block of the matrix.

The quantities Γ and Γ' are, of course, unknown a Priori. However, they may be determined by a careful coordination of the local rock-matrix solutions and the global solution for the fractures. In particular, a forward-elimination procedure is used on the pressure equation, i.e., Equations (6-32a) and (6-33a), to yield

$$\Gamma'_W = \Gamma'_{Wo} + \Omega'_W \delta p \tag{6-37a}$$

A similar procedure is applied to the heat and brine equations, i.e., Equations (6-32b,c) and (6-33b,c), to yield

$$\Gamma'_H = \Gamma'_{Ho} + w\Omega'_H \delta T \tag{6-37b}$$

and

$$\Gamma'_C = \Gamma'_{Co} + w\Omega'_H \delta\hat{C} \tag{6-37c}$$

The terms Γ'_{Xo} and Ω'_X, are functions of time but may be considered as constants for a single time-step Δt. Dependent variables δp, δT, and δC pertain to the interface grid block n_s. However, since the node for this grid block is positioned on the boundary between fracture and matrix, these variables are also the global variables as has been anticipated by suppressing the prime notation for the state variables. Equations (6-36) and (6-37) may therefore be combined to yield sink terms of the form

$$\Gamma_W = \Gamma_{Wo} + \Omega_W \delta p \tag{6-38a}$$

$$\Gamma_H = \Gamma_{Ho} + w\Omega_H \delta T \tag{6-38b}$$

$$\Gamma_C = \Gamma_{Co} + w\Omega_C \delta\hat{C} \tag{6-38c}$$

These conditions may be applied directly to the global equations, and solutions to the global equations may then be determined. Consequently, terms Γ'_X may then be determined.

Source terms Γ'_X may then be evaluated, and back substitution may be applied to the equations for the rock matrix to yield the incremental changes $\delta p'$, $\delta T'$, and $\delta C'$.

6.4.3 SOLUTION TECHNIQUES

In addition to the procedure for determination of the fracture/matrix coupling, the same nonlinearity and interdependence of the variables is present here as it is for the global equations. Furthermore. the method of analysis is the same here as for the global solution, Section 6.1.6. An iteration loop is used for the nonlinearity within both the transport and source terms. Again, the nonlinearity and coupling is removed from the transmissibility by evaluating this quantity at time-step n. A detailed explanation of the solution technique is given in Appendix D.

6.4.4 CONVERGENCE

The iteration loop here is the same as that used for the global solution. Consequently, the same density-based criteria are applied. Thus, for a given iteration it+1, the relative changes in density due to temperature and concentration changes with respect to iteration it are given by

$$\varepsilon'_T = \max_i \left\{ c_T \left[\left(T'_i\right)^{it+1} - \left(T'_i\right)^{it} \right] \right\} \tag{6-39a}$$

and

$$\varepsilon'_C = \max_i \left\{ c_C \left[\left(\hat{C}'_i\right)^{it+1} - \left(C'_i\right)^{it} \right] \right\} \tag{6-39b}$$

These relative changes ε' are, of course, specific to a particular global grid-block m and are coalesced with the relative changes for the global grid block as shown in Equations (6-21). The specified criteria for convergence are then applied to the coalesced changes, as discussed in Section 6.1.7.

6.5 RADIONUCLIDE TRANSPORT WITHIN THE ROCK MATRIX

The finite-difference equation for matrix grid-block i, corresponding to Equation (2-22) is

$$\overline{R}'_r = K'_r \, \delta\left(V'\rho'C_r\right) / \Delta t \tag{6-40a}$$

where

$$\begin{aligned}\overline{R}'_r = &-\Delta\left(C'_r \, u'\right) + \Delta\left(T'_C \, \Delta C'_r\right) + \sum_s \; k_{rs}\lambda_s K'_s \, V'\rho'C'_s \\ &- \lambda_r K'_r \, V'\rho'C'_r + C'_r \, \Gamma'_W + \Gamma'_r\end{aligned} \tag{6-40b}$$

(The notation is given in a separate section.)

In contrast to the global equations, where a general Freundlich isotherm is permitted, a linear isotherm is assumed here, which results in the constant retardation K' defined in Equation (2-27). The implication of this fact is that no iterations are required in the solution algorithm. The fracture/matrix source terms, included in Equation (6-40b), may be evaluated by a careful coordination of the solution procedure for the local and global calculations, as discussed in Section 6.4.2.

For a chain of radionuclides, Equations (6-40) are solved sequentially, once for each nuclide of the chain. If the nuclides have been numbered properly within the input, then the calculations will proceed from parent to daughter, and the production term for the daughter, the third term on the right-hand side of Equation (6-40b), may be calculated from the most current parent concentrations. The details of the analysis of Equations (6-40) are discussed in Appendix E.

6.6 FREEWATER SURFACE

The freewater surface option, as described in Section 5.6, impacts the accumulation and transmissibilities of the pressure equation. In this section the implementation of transient and steady-state flow solutions is presented.

Equations (5-63) and (5-66) are used to modify the expansion of the accumulation terms in Equation (A-14a) as follows

$$\delta\left(S\rho V\right) = S\delta\left(\rho V\right) + \rho V\delta S \tag{6-41}$$

$$= S\left(V^{n}\delta p + \rho^{n+1}\delta V\right) + \rho V\delta S \tag{6-42}$$

Here the first term is the same as Equation (A-14a), as multiplied by the block saturation. This represents compressible fluid accumulation or confined aquifer storage. The second term is the release of water from drainage or an unconfined yield. The latter term dominates in typical application, thus allowing transient, as well as steady-state, phreatic conditions.

To account for the reduced cross-sectional area available for flow in a partially saturated block, the fluid transmissibilities as defined in Section 6.1 are modified as follows:

$$T_{Wxx} = T_{Wxx}S \tag{6-43a}$$

$$T_{Wyy} = T_{Wyy}S \tag{6-43b}$$

$$T_{Wzz} = T_{Wzz} \tag{6-43c}$$

where the saturation is allowed to vary between 0.001 and 1.0. Only the horizontal components are modified to reflect the reduced area available for flow. A minimum saturation is maintained such that blocks that fully desaturate may be resaturated in subsequent time steps.

Within the model special consideration has been given to the accumulation terms in Equation (6-42). Following a series of numerical investigations, the authors have found it necessary to separate the implicit and explicit components. Depending upon whether the freewater surface is above, within, or below a grid block before and after a time step (before or after an iteration for steady-state), special handling of the terms is required. The logic is detailed in subroutine SCOEF.

7 NUMERICAL CONSIDERATIONS

Two major topics are considered in this section, namely geometrical gridding procedures and numerical criteria. Both three-dimensional Cartesian and axially symmetric cylindrical coordinates are used in SWIFT for Windows. Discretization in a Cartesian geometry is done through direct input of the increments. Since this is a very straightforward process, little explanation is needed for the basic gridding procedure. However, the code does permit some important mesh modifications. These are discussed in this chapter. In a cylindrical geometry, the radial mesh may be generated automatically, a procedure which is also presented here. In either of the above cases, the geometry is called a global geometry in order to distinguish it from yet another type of discretization. Certain grid blocks may be designed as doubly porous. For such blocks a local mesh is automatically generated. The method used is given in the final section on geometry.

The subject of numerical criteria is related to that of geometrical discretization. Convective transport, which is simulated by the code, introduces limitations on the space steps. It also introduces limitations upon the time steps. This is discussed in a section below. For radionuclide transport the time steps are also controlled by the half-lives of the various constituents. Single-species decay is treated in one section, and radionuclide chains are treated in another.

7.1 GEOMETRY

7.1.1 DISCRETIZATION OF THE GLOBAL SYSTEM IN CARTESIAN COORDINATES

Figure 5-1 depicts both definitions used and options available for characterization of the global systems. In this figure one will note the reference plane. The existence of this plane is assumed implicitly by the data setup. Initially the center of the (1, 1, 1) grid block is coincident with this plane, and it remains so unless altered by a nonzero translation h_o (input parameter DEPTH [Reeves et al., 1986a, p. 31]). Angles θ_x and θ_y (SINX and SINY in the input [Reeves et al., 1986a, p. 31]) then position the remainder of the system with respect to grid-block (1, 1, 1). The sign convention here is that these angles are positive if they open in

the direction of positive z (downward). If additional adjustment of elevation is desired, then parameter Δh (positive downward) may be used (input parameter HADD [Reeves et al., 1986a, p. 35]) on a grid-block-by-gridblock basis.

With the usual sign convention, the datum is positioned with respect to the reference plane by h_D (HDATUM [Reeves et al., 1986a, p. 28]). Quantity h_I (HINIT [Reeves et al., 1986a, p. 28]) is also located relative to the reference plane. It and p_I (PINIT (Reeves et al., 1986a, p. 28]) are used to define the initial pressure distribution, as discussed in Section 5.1.1.

Not shown in Figure 5-1 are two other geometrical options, namely a grid-block thickness adjustment and a grid-block pore-volume adjustment. The significance of both of these is shown in Figure 7-1. Lehman and Quinn (1982) used this gridding in their model-comparison studies for the Pasco Basin. As indicated there, extensive modifications are included via the Δh parameters in order to fit the surface topography. Furthermore, the grid-block regions indicated as "not used" are removed from the simulation by setting pore volumes to zero. Internally, what this means to the code is that the matrix solver eliminates the corresponding coefficient matrix elements both from the matrix storage and from the matrix solutions in order to reduce computer resource requirements.

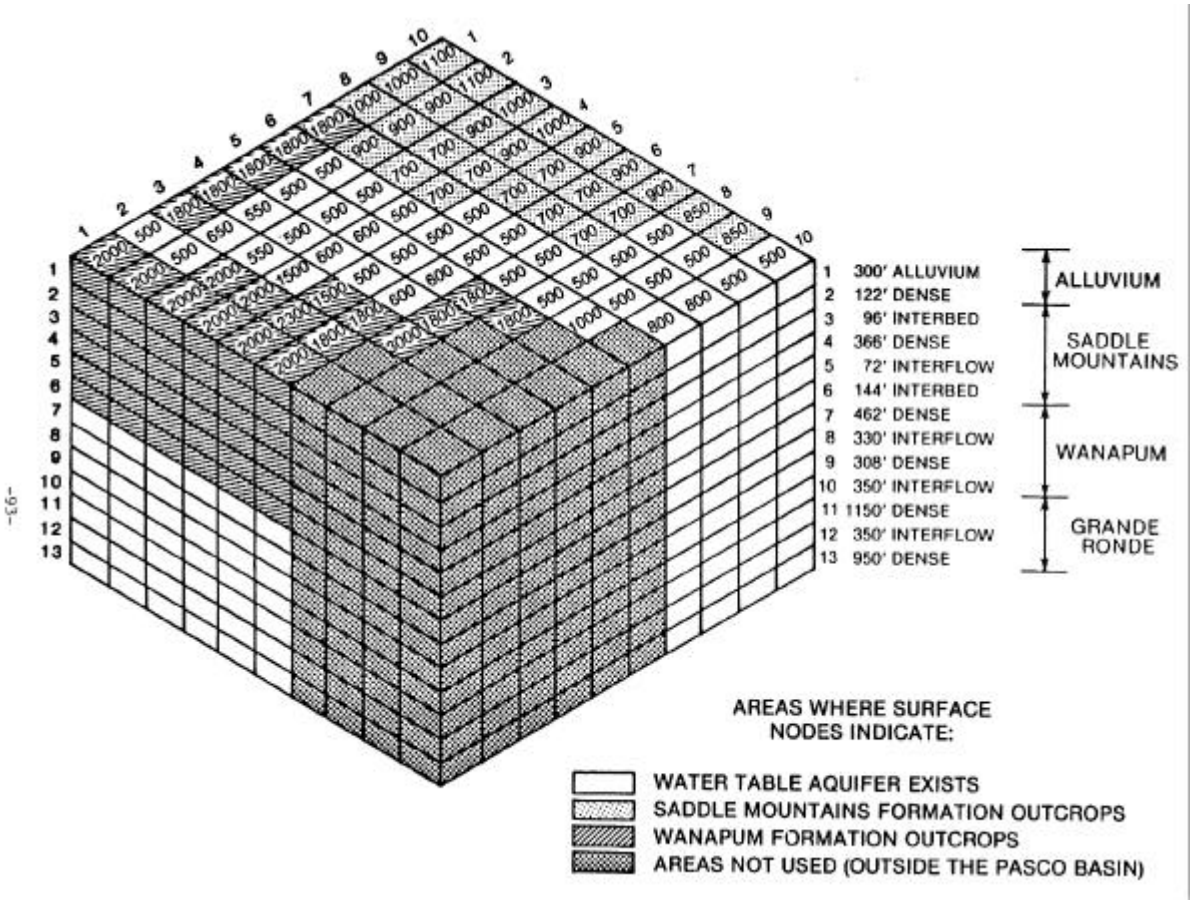
7.1.2 DISCRETIZATION OF THE GLOBAL SYSTEM IN CYLINDRICAL COORDINATES

For a Cartesian (x, y, z) coordinate system, the SWIFT for Windows code requires the user to generate his own mesh by specifying all values of the increments Δx, Δ, and Δz. For radial (r, z) coordinates, however, the mesh may be either user-generated or automatically generated. Automatic generation is based on a special steady-state solution of the flow equation, which gives, for the pressure difference between two points,

$$p^2 - p^1 \approx 1n\left(r_2 / r_1\right) \tag{7-1}$$

This same relation is the basis for the well index given in Equation (4-3) of Chapter 4.

Figure 7.1. A Gridding of the Pasco Basin in Southeastern Washington.



Four parameters are used in the automatic generation, namely (1) the number of grid blocks n (input variable NX [Reeves et al., 1986a, p. 10]), the radius r_1 (R1V) (Reeves et al., 1986a, p. 33) to the center of the first grid block, which is interpreted as the outer radius of the disturbed skin zone, (3) the wellbore radius $\hat{r}_1$ (RWW), and (4) the grid boundary $\hat{r}_{n+1}$ (REV) (Reeves et al., 1986a, p. 331).

The radial mesh, which is shown schematically in Figure 7-2, is generated by assuming equal pressure drops between adjacent mesh points for the steady-state solution of Equation (7-1). Using the notation of Figure 7-2, this means that

$$\frac{r_{i+1}}{r_i} = A \tag{7-2}$$

and

$$\frac{\hat{r}_i}{r_i} = A^{1/2} \tag{7-3}$$

A combination of Equations (7-2) and (7-3) gives

$$\frac{\hat{r}_{n+1}}{r_1} = A^{n-1/2} \tag{7-4}$$

Equation (7-4) is solved by SWIFT for Windows for the common ratio A, and then Equations (7-2) and (7-3) are used as recursion relations to define all r_i and $\hat{r}_i$.

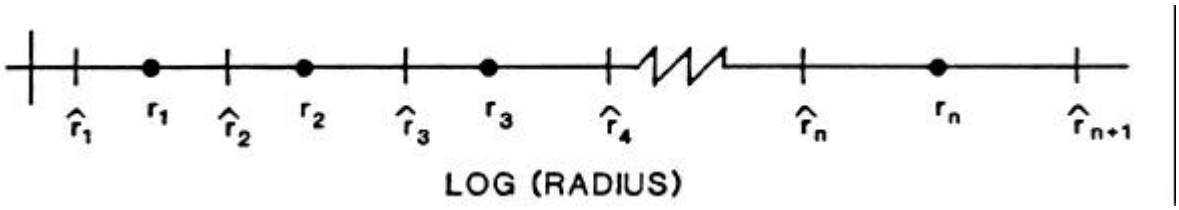
Frequently the analyst will not know the outer skin radius, r_1, which must be input to the code. In such an event, it is suggested that Equation (7-4) be extended by taking

$$\frac{r_1}{\hat{r}_1} = A^{1/2} \tag{7-5}$$

and

$$\frac{\hat{r}_{n+1}}{\hat{r}_1} = A^{n} \tag{7-6}$$

Figure 7.2. Schematic of Radial Mesh Including Grid-Block Centers, r_i, and Grid-Block Boundaries, $\hat{r}_i$.



Equation (7-6) may be solved for A. This parameter may then be used in Equation (7-5) to calculate the skin radius r_1 from the wellbore radius $\hat{r}_1$. In such a case, where the skin thickness is chosen somewhat arbitrarily, the properties of the skin are reflected solely by the well index.

7.1.3 DISCRETIZATION OF THE LOCAL SYSTEM

The basic concept of a matrix substructure was presented in Chapter 2, with the discussion focused on Figure 2-1. Here that discussion is continued. Basically, there are two levels of discretization for the rock matrix. For the first level, the matrix is divided into either prisms or spheres (see Figure 7-3), as specified by the analyst. Prisms would be appropriate for the parallel non-intersecting fractures typified in Figure 2-1a, and spheres would be an appropriate approximation for the intersecting sets of fractures typified in Figure 2-1b. The prisms are arbitrarily given a cross-sectional area of unity and a depth a (input parameter SAD [Reeves et al., 1986a, p. 46]), whereas the spheres are taken to have radius a. All local units, whether prismatic or spherical, are assumed to be homogeneous with each such unit within a given global block having the same hydrological, thermal and geochemical parameters. Such parameters may vary, however, from one global block to another, depending upon the specified matrix rock type (input parameter IRT [Reeves et al., 1986a, p. 57]).

The second level of discretization is the gridding of a representative local unit. Since all units within a given global block are exposed to the same boundary conditions and initial conditions, it is only necessary to simulate one representative unit and then to scale the flux Γ at the fracture/matrix interface by the number of matrix units in the global block. In addition, no unit-to-unit transport is permitted so that each representative prism or sphere is characterized by one-dimensional flow and transport.

Figure 7-4 shows a typical one-dimensional gridding for a representative matrix unit. As shown, nodes are positioned at each boundary in order to facilitate the application of boundary conditions. The remainder of the node and block boundaries are positioned automatically by the code, subject to the three parameters: a (input parameter SAD), $\Delta s_{n_s-1/2}$

Figure 7.3. Typical Matrix Structural Units Imbedded Within a Primary Grid Block for a Dual-Porosity Implementation.

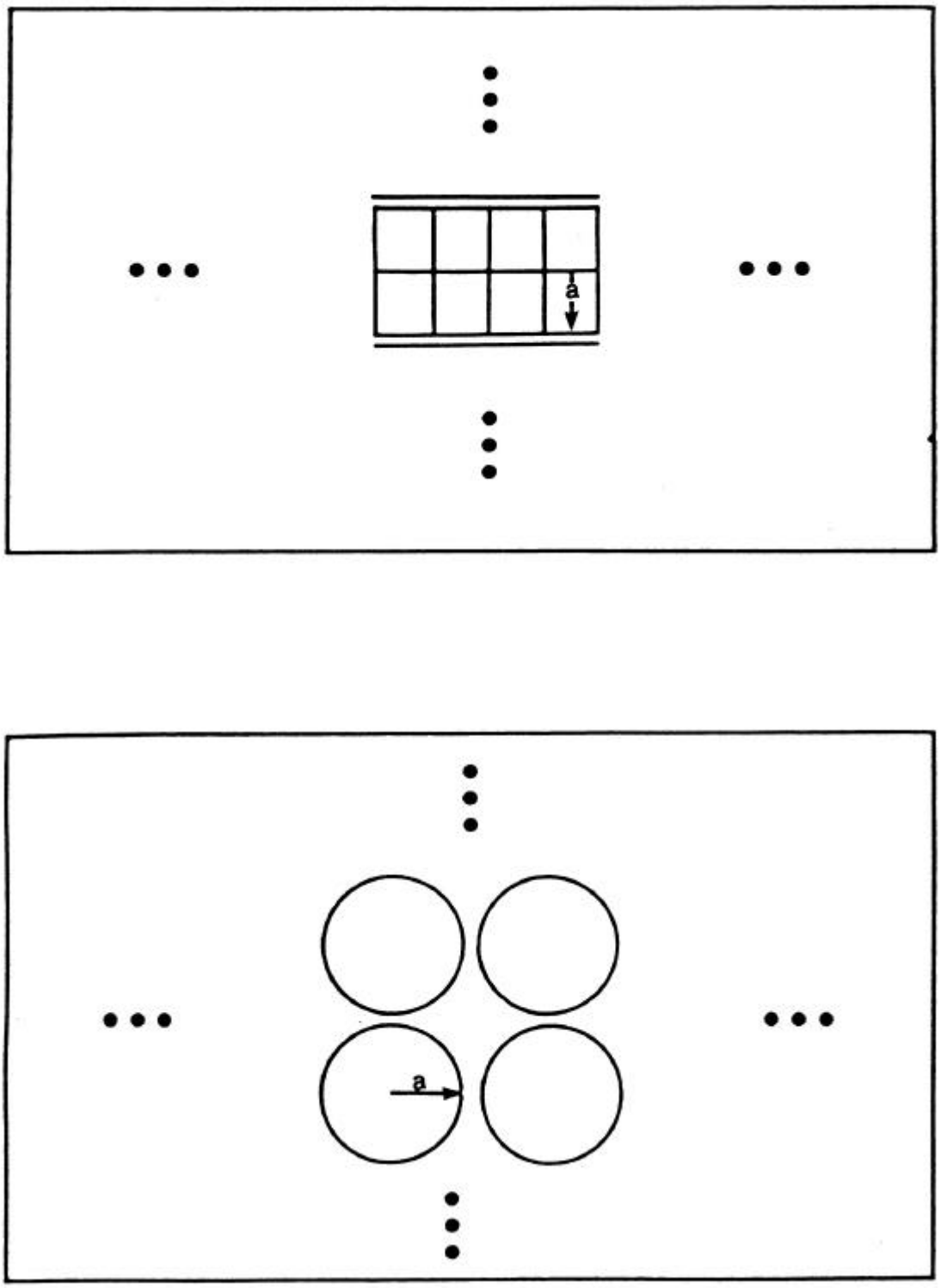
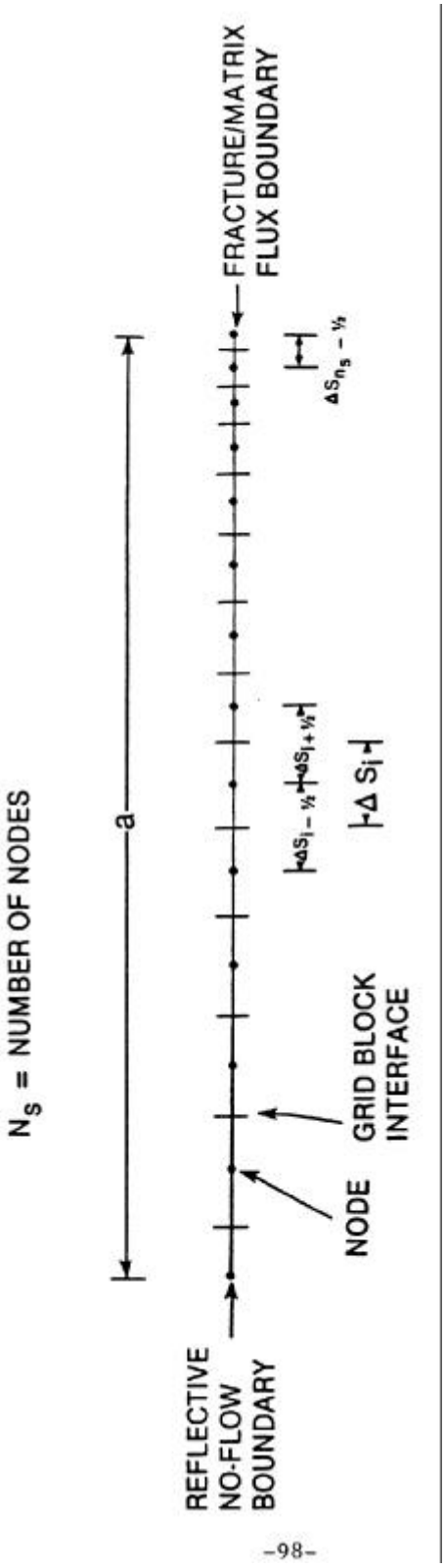


Figure 7.4. Discretization of the Local (Matrix) System.



(DSD [Reeves et al., 1986a, p. 46]) and n (NSD [Reeves et al., 1986a, p. 15]). Each is specified as a function of matrix rock type. The procedure used by the code employs a grading factor A:

$$\Delta s_{i-1/2} = A\Delta s_{i+1/2} \tag{7-7a}$$

By summing all increments, a geometric series is formed, and an equation for A is obtained:

$$\Delta s_{n_s-1/2}\left(1 - A^{n_s-1}\right) = a(1 - A) \tag{7-7b}$$

This equation is solved for A using nonlinear techniques. Equation (7-7a) then is used to generate all of the nodal positions starting from the known value of Δs at the fracture/matrix interface.

As discussed in Aziz and Settari (1979), the positioning of grid-block interfaces is crucial to the definition of a consistent mesh. (A consistent mesh is one for which the numerical solution, with sufficiently fine discretization, will converge to the analytic solution, if the latter exists.) Following these authors (pp. 75-79, 88), we define

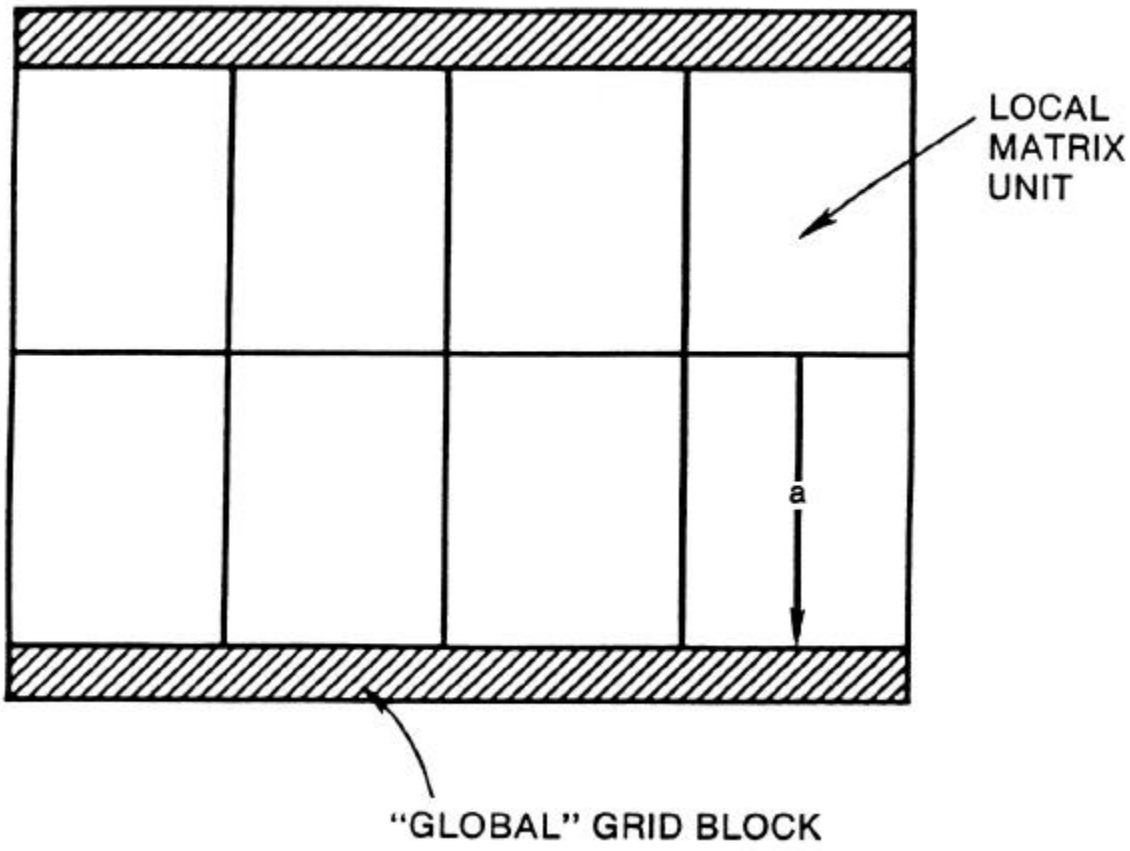
$$s_{i+1/2} = (s_i + s_{i+1})/2 \quad \text{(prism)} \tag{7-8a}$$

$$s_{i+1/2} = \begin{cases} (s_i s_{i+1})^{1/2} & , \ i \neq 1 \\ (s_i + s_{i+1})/2 & , \ i = 1 \end{cases} \quad \text{(sphere)} \tag{7-8b}$$

For the sphere the definition of $s_{3/2}$, of necessity, departs from a consistent definition. However, this introduces only a local inconsistency in what is physically a relatively unimportant region of the mesh.

In addition to a dual-porosity mode, the SWIFT for Windows code may also be applied in a discrete-fracture mode. For the latter case, the individual fractures are gridded separately by the global mesh as typified by Figure 7-5. The rock matrix, as shown in that figure, is characterized by the same structure as for the dual-porosity mode. The only difference is that the rock matrix is conceptualized as being external to the global grid block. The analyst may choose to mix discrete fractures and dual porosity in the same simulation in order to characterize the dominant fractures.

Figure 7.5. Matrix of Structural Units Adjacent to Primary Grid Blocks for a Discrete-Fracture Implementation.



7.2 NUMERICAL CRITERIA

7.2.1 NUMERICAL CRITERIA FOR DISPERSION AND OVERSHOOT

One source of difficulty in the numerical solution of transport equations, such as Equations (2-2) through (2-4), is the treatment of the convection term. For conventional finite-difference and finite-element solvers, either numerical dispersion or the overshoot -undershoot phenomena may be introduced. Other methods, or variations of the above-mentioned techniques, have been introduced to help alleviate this problem. Examples are the method of characteristics (Garder et al., 1964; Bredehoeft and Pinder, 1973), higher-order Galerkin (Price et al., 1968; Pinder, 1973), and various upstream-weighting and asymmetric weighting strategies (Nolen and Berry, 1972; Christie et al., 1976). The Distributed-Velocity Method (Campbell et al., 1981) holds some promise in this area also.

The SWIFT for Windows code gives the user several options. For example, he may elect to use second-order correct central-difference approximations in both time and space. These techniques have the advantage that no numerical dispersion is introduced. The disadvantage they introduce are limitations on both block size and time step. These limitations are necessary to prevent calculated concentrations from exceeding the injection or solubility levels or from being less than the initial values, i.e., overshoot-undershoot. The user also may choose to use first-order correct backward-difference approximations in both time and space. These techniques have no overshoot-undershoot, but they introduce numerical dispersion. Thus, limitations on block size and time step are again called for. The radionuclide, brine and heat-transport equations, Equations (2-2) - (2-4), which contain convective terms, are largely responsible for the importance of numerical dispersion. The flow equation, since it does not contain a convective, i.e., a first-order derivative term, has a truncation error which is much less significant. The time-step and block-size restrictions are not overly severe for many problems. In such cases, the analyst may elect to use the backward-difference schemes due to their inherent stability against overshoot.

A table is presented below (Table 7.1) which contains information on numerical dispersion and overshoot. It is based on a simple analogue of Equations (2-2) through (2-4), namely

$$D\frac{\partial^2\psi}{\partial x^2} - v\frac{\partial\psi}{\partial x} = \frac{\partial\psi}{\partial t} \tag{7-9}$$

where the dependent variable ψ may denote either temperature T, brine concentration $\hat{C}$, or radionuclide concentration C. Quantities v and D' denote the retarded interstitial velocity and dispersion. respectively.

Table 7.1. Numerical criteria for brine, heat, and radionuclide transport.

Scheme	Numerical Dispersion	Dispersion Criterion	Overshoot Criteria
CIT-CIS	None	None	$v\Delta t/\Delta x+2D\Delta t/\Delta x^2 \le 2$ $v\Delta x/2 \le D$
CIT-BIS	$v\Delta x/2$	$v\Delta x/2 << D$	$v\Delta t/\Delta x+2D\Delta t/\Delta x^2 \le 2$
BIT-CIS	$v^2\Delta t/2$	$v^2\Delta t/2 << D$	$v\Delta x/2 \le D$
BIT-BIS	$v\Delta x /2+v^2\Delta t/2$	$v\Delta x/2+v^2\Delta t/2 << D$	None

Here CIT means centered-in-time, CIS means central-in-space, BIT refers to backward-in-time and BIS refers to backward-in-space.

Definition of terms:

$$k_d = 0 \quad , \text{ brine transport}$$

$$K = \begin{cases} 1+(1-\phi)\rho_R k_d/\phi & , \text{ radionuclide or brine transport} \\ 1+(1-\phi)\rho_R C_{pR}/\phi\rho c_w & , \text{ heat transport} \end{cases}$$

$$V = u/\phi K$$

$$D = \begin{cases} (\alpha_L u + D_m)/K\phi & , \text{ radionuclide or brine transport} \\ (\alpha_L u\rho c_p + K_m)/K\phi\rho c_p & , \text{ heat transport} \end{cases}$$

In spite of the simplicity of Equation (7-9), space and time-step criteria derived from it have proven to be quite useful in practice. Such criteria are given in Table 7.1. Additional discussion of these relations is given in Lantz (1971) and INTERCOMP (1976).

7.2.2 ADJUSTMENT TO THE RATE CONSTANT

The SWIFT for Windows code will adjust the decay constant as specified by the analyst (input parameter LADJ [Reeves et al., 1986a, p. 14]). The basis for this adjustment is another analogue equation. If one considers the static case, in which transport is negligible compared to decay, then the radionuclide decay equation, Equation (2-4) becomes, for the parent nuclide,

$$\frac{dC}{dt} = -\lambda C \tag{7-10}$$

(The component subscript has been dropped for convenience.)

The analytic solution of Equation (7-10) across an interval Δt is

$$C^{n+1} = C^n e^{-\lambda \Delta t} \tag{7-11a}$$

which yields

$$\delta C = -C^n \left(q - e^{-\lambda \Delta t} \right) \tag{7-11b}$$

If Equation (7-10) is solved numerically, the result is

$$\delta C = -\lambda'(w)\left(C^n + w\delta C\right)\Delta t \tag{7-12a}$$

where

$$w = \begin{cases} 1/2 & , \text{ CIT} \\ 1 & , \text{ BIT} \end{cases} \tag{7-12b}$$

for centered-in-time (CIT) and backward-in-time (BIT) differencing.

To see that quantity $\lambda'(w)$, the adjusted rate constant, is indeed a function of the time-weighting constant w, we solve Equation (7-12a) for δC and substitute into Equation (7-11b). The result is

$$\lambda'_i(w) = \frac{e^{\lambda_i \Delta t} - 1}{\Delta t\left[(1-w)e^{\lambda_i \Delta t} + w\right]} \tag{7-13}$$

where the component subscript has been reintroduced. For the BIT algorithm w = 1 Equation (7-12) reduces to Equation (5-3) of the original SWIFT document (Dillon et al., 1978). As

described quantitatively therein, rate adjustment is appropriate for near-static cases, where decay dominates transport, for the case of a parent nuclide. Furthermore, rate adjustment via Equation (7-13) is appropriate for decay-dominant transport of a daughter nuclide i-1, providing that

$$\tau_{i-1} / \tau_i > 100 \tag{7-14}$$

Without adjustment the time step must be controlled by the criterion

$$\Delta t < \begin{cases} \tau_i / 7 & , \text{ BIT} \\ \tau_i & , \text{ CIT} \end{cases} \tag{7-15}$$

where τ_i is the half life of the component. With adjustment, this criterion is removed in many cases.

7.2.3 NUMERICAL CRITERIA FOR SECULAR EQUILIBRIUM

In considering the transport of chains of radionuclides, cases involving secular equilibrium are encountered. For such situations the numerical criteria of Sections 7.2.1 and 7.2.2 frequently may be relaxed. However, before examining the criteria, it is appropriate to give a definition for secular equilibrium.

Considering, for simplicity, the case of parent and daughter nuclei, the former follows a simple exponential decay law

$$N_1 = N_{10} e^{-\lambda t} \tag{7-16}$$

for the case of static flow. Here, for convenience, the concentration notation C has been changed to number-of-nuclei notation N, with subscripts added to denote nuclide identity and initial values. For the daughter component

$$\frac{dN_2}{dt} = \lambda_1 N_1 - \lambda_2 N_2 \tag{7-17a}$$

which yields the solution

$$N_2 = \lambda_1 N_{10} \left(e^{-\lambda_1 t} - e^{-\lambda_2 t} \right) / \left(\lambda_2 - \lambda_1 \right) \tag{7-17b}$$

If the half life of the parent is much greater than that of the daughter, i.e., if

$$\tau_i >> \tau_{i+1} \tag{7-18a}$$

or, for the decay constants,

$$\lambda_i << \lambda_{i+1} \tag{7-18b}$$

then secular equilibrium occurs. This means, for the case of parent and daughter, that

$$\lambda_1 N_1 = \lambda_2 N_2 \tag{7-19}$$

a relation easily derived from Equations (7-16) and (7-17b).

Figure 7-6 provides an example of secular equilibrium for the decay chain

$$140_{Ba} \xrightarrow[12.8d]{\beta^-} 140_{La} \xrightarrow[40.2hr]{\beta^-} 140_{Ce}$$

Since the half life of 140_{Ba} is approximately seven times greater than that of 140_{La}, the condition for secular equilibrium is considered to be satisfied, and secular equilibrium occurs, in this case, after only 15 hours have elapsed from the initial, nonequilibrium condition.

For a long chain, where all elements of the chain (except for the final stable daughter product) are in secular equilibrium, each radionuclide builds up to equilibrium with its parent. When this occurs, a generalized form of Equation (7-19) is valid, namely

$$\lambda_1 N_1 = \lambda_2 N_2 = \lambda_3 N_3 = \ \ldots \tag{7-20}$$

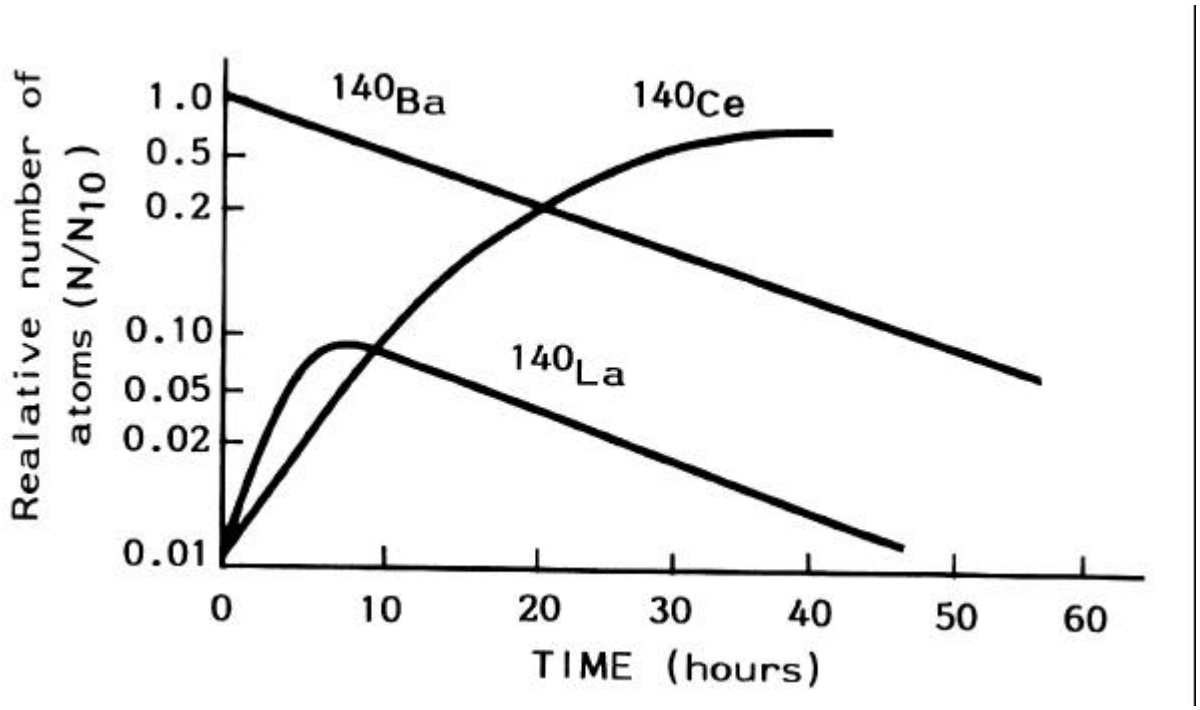
This means simply that under conditions of secular equilibrium, the activities of all components are equal.

A more complete discussion of various types of equilibria is given in Evans (1955). However, the point to be made here is that secular equilibrium under nonstatic conditions is significant insofar as the numerical criteria are concerned, for the following reason. It has been found by experimental numerical methods that the criteria for the daughter components may frequently be ignored in such a case. For example, in the case of the actinide chain

$$^{242}\text{Pu} \rightarrow {}^{238}\text{U} \rightarrow {}^{234}\text{U} \rightarrow {}^{232}\text{Th} \rightarrow {}^{236}\text{Ra}$$

time steps would have to be controlled via the relatively short 1600y half life of ^{226}Ra were it not for its secular equilibrium with ^{238}U. This means that only the time step of the latter, as determined from Table 7.1, need be enforced. Since the criteria for chains in secular

Figure 7.6. Decay of Radioactive ^{140}Ba. After approximately 15 hours the ^{140}Ba and ^{140}La will be in secular equilibrium.



equilibrium have not been derived theoretically, one must proceed with some caution, however, making appropriate numerical checks. Nevertheless, the relaxation of numerical criteria resulting from secular equilibrium is a necessary consideration for applications in which computer efficiency is important.

8 COMPUTER IMPLEMENTATION

The SWIFT for Windows code has been developed to implement the formalism described in the previous chapters. This chapter first presents the program structure, in which the functions of the various subprograms are dealt with in a very general fashion. It then focuses upon the two algorithms used in the matrix-solution routines, since the matrix-solution algorithms, of course, are of crucial importance to the efficiency of three-dimensional simulations.

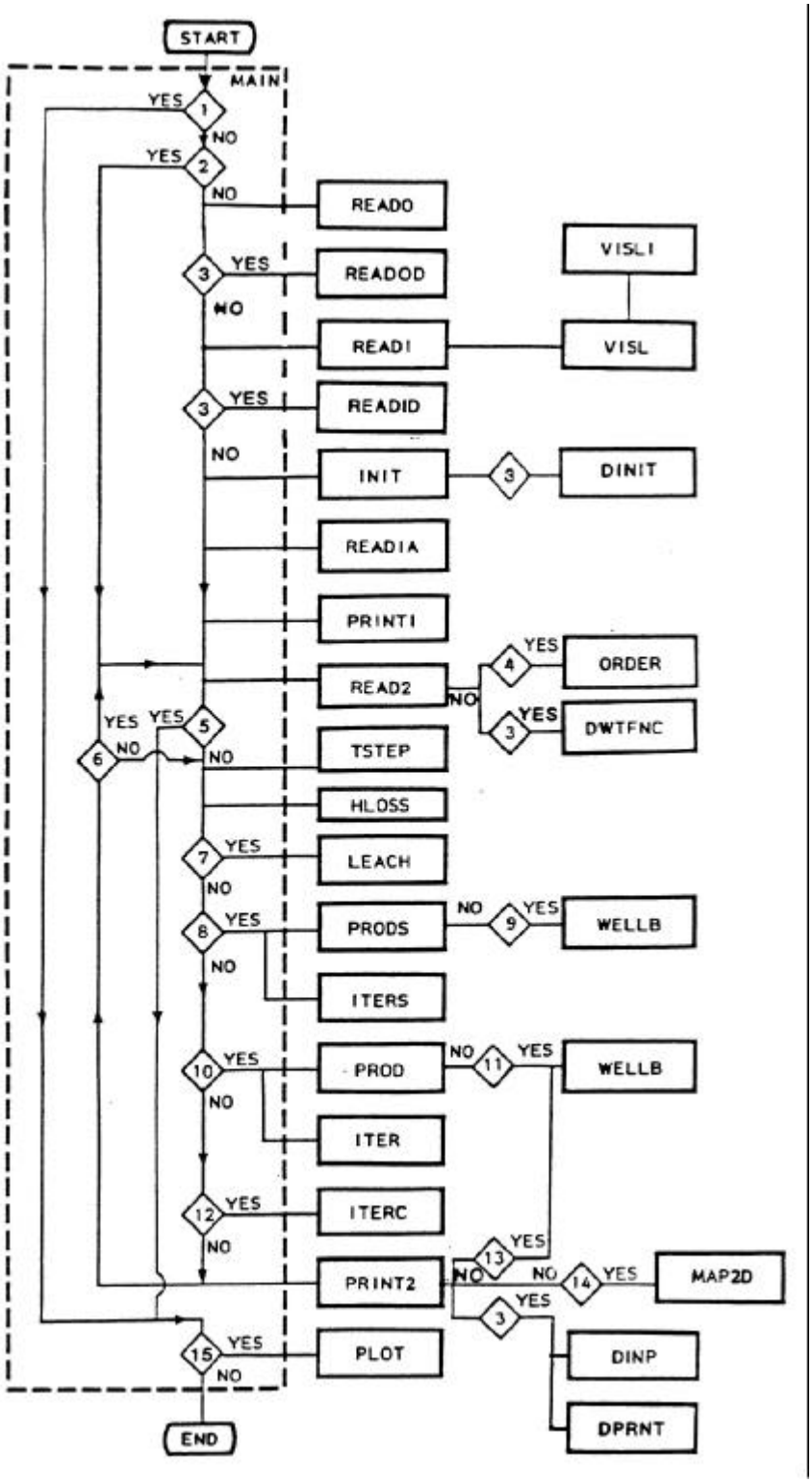
8.1 PROGRAM STRUCTURE

The SWIFT for Windows code consists of a main routine, three integration subroutines, and about 70 supporting subroutines. In this chapter five items are presented. First, the structure of routine MAIN is shown in Figure 8-1. As may be inferred from this figure, the basic organization of MAIN is focused upon the three global integration subroutines ITER, ITERS and ITERC. The other subroutines perform the functions of input, initialization, parameter definition, and output in support of either ITER or ITERS, which integrate the flow, heat, and brine equations, or ITERC, which integrates the radionuclide equations.

The second, third, and fourth items presented show the structure of the three global integration subroutines. Routine ITER is shown diagrammatically in Figure 8-2; ITERS in Figure 8-3; and ITERC in Figure 8-4. As shown, these routines are quite similar. In each case the various finite-difference terms are assembled into matrix equations, which are subsequently solved for the appropriate dependent variable. The matrix solutions in each case may be performed in one of three different routines, depending upon user preference. The user options are two-line successive overrelaxation (L2SOR) and Gaussian elimination (GAUS3D) or GAUS1D) with the latter algorithm specialized within a special subroutine for one-dimensional applications.

The major difference between the three integrators is the complexity of the setup process, with ITER, the coupled-equation transient simulator, understandably the most

Figure 8.1. Structure of the SWIFT for Windows Code.



Conditions for Figure 8.1 are as follows:

1. Are plots desired for a previous run?

2. Is this a restart run?

3. Is dual porosity to be included?

4. Is reduced bandwidth direct method of solution used? (multi-dimensional problems only)

5. Is run to be terminated at this time step?

6. Are the recurrent data read at this time step?

7. Is the waste-leach submodel employed?

8. Is the steady-state pressure solution sought?

9. Are steady-state wellbore calculations to be solved?

10. Is the transient pressure solution sought?

11. Are the transient wellbore calculations to be solved?

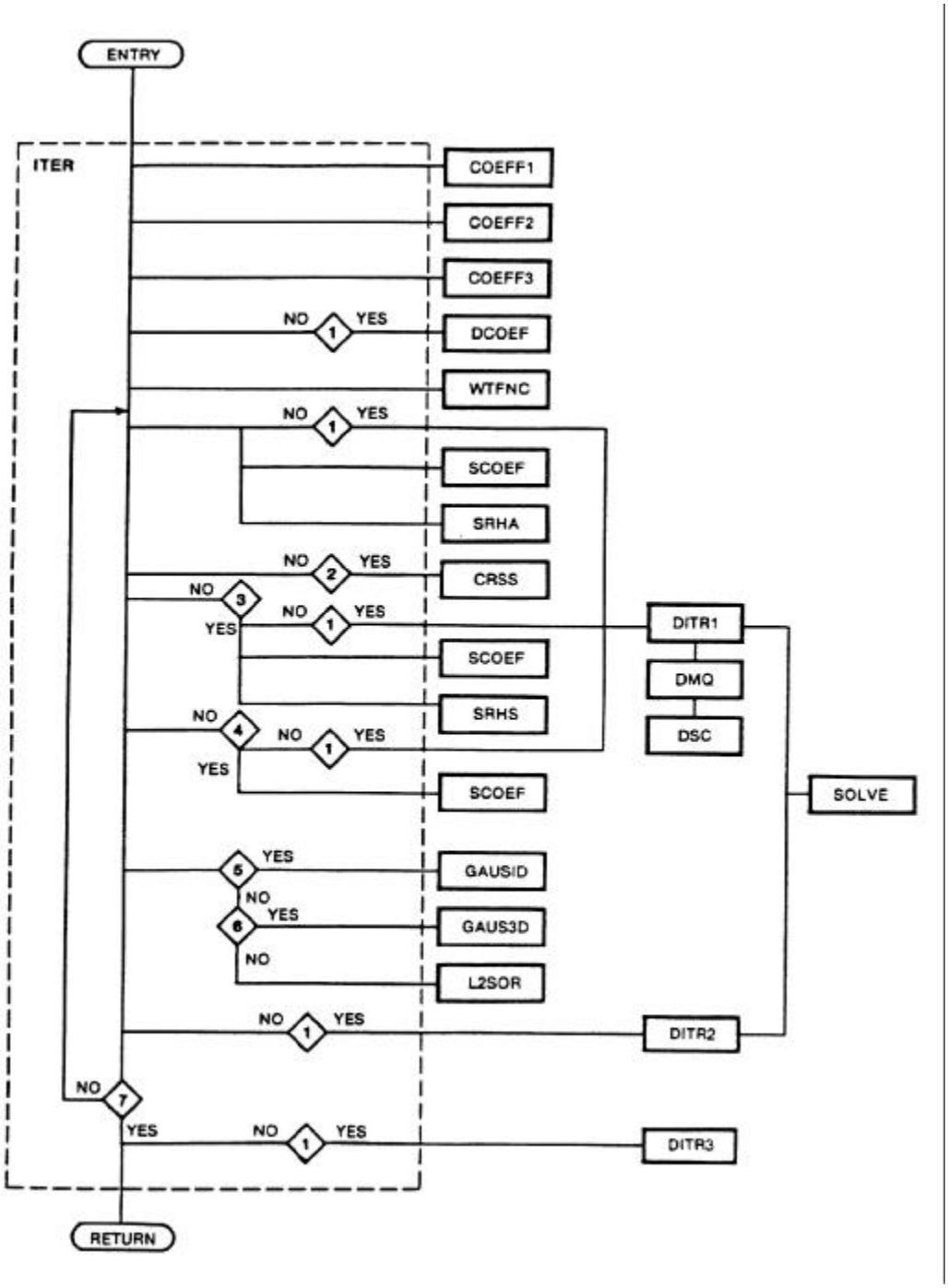
12. Are the radionuclide transport calculations to be solved?

13. In the transient wellbore calculations are the well rates calculated semi-implicitly?

14. Are any two-dimensional contour maps desired?

15. Are any plots desired for this run?

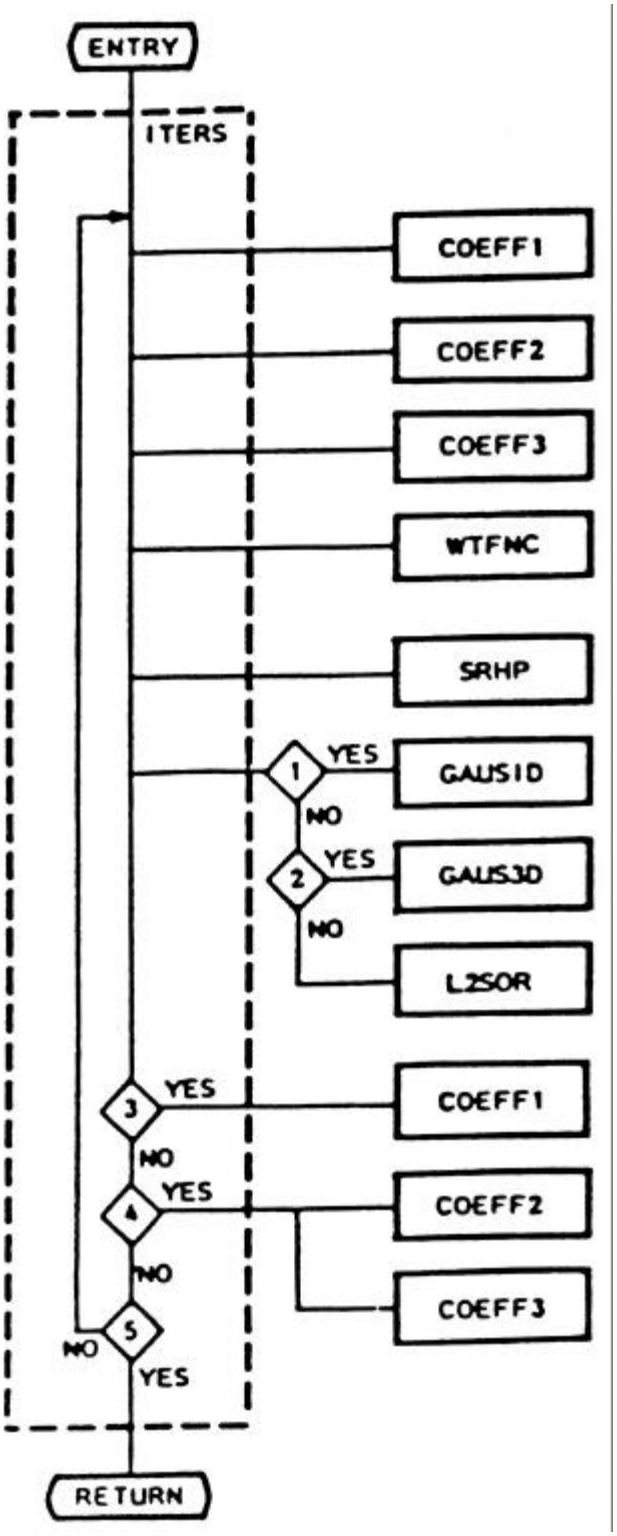
Figure 8.2. Solution of the Transient Flow and Transport Equations in ITER.



Conditions for Figure 8-2 are as follows:

1. Is dual porosity to be considered?
2. Are the cross-derivative dispersion terms to be included implicitly?
3. Is the temperature equation to be solved?
4. Is the brine equation to be solved?
5. Is the problem one-dimensional?
6. Is a direct solution desired?
7. Has convergence been achieved?

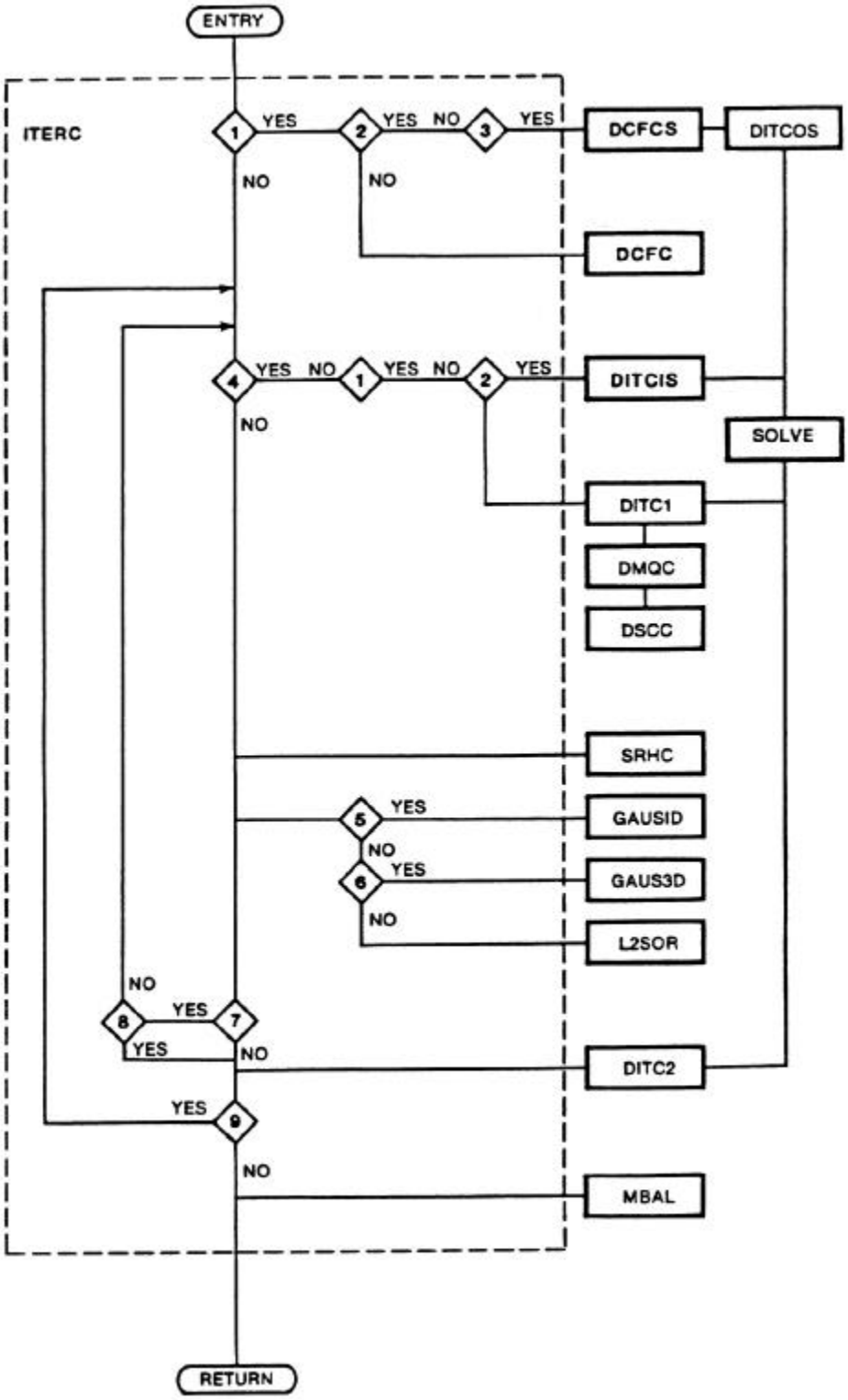
Figure 8.3. Solution of the Steady-State Flow and Brine Equations in ITERS.



Conditions for Figure 8-3 are as follows:

1. Is the problem one-dimensional?
2. Is a direct solution desired?
3. Are Darcy velocities desired?
4. Are dispersions and transmissibilities desired?
5. Has convergence been achieved?

Figure 8.4. Solution of the Radionuclide Equations in ITERC.



Conditions of Figure 8-4 are as follows:

1. Is dual porosity to be considered?

2. Is the simplified solution to be obtained assuming a constant density with no convection or dispersion?

3. Has the time step changed in magnitude?

4. Is this the first iteration?

5. Is the problem one dimensional?

6. Is a direct solution desired?

7. Is a nonlinear Freundlich isotherm to be used?

8. Has convergence been achieved?

9. Is there another radionuclide in the chain?

complex, and with ITERC, the transient radionuclide simulator, the least complex. Comparing ITER (Figure 8-2) with ITERS (Figure 8-3), the major difference is SCOEF.

This routine sets up time-dependent coefficients in the former which are, of course, absent from the latter, since ITERS is a steady-state integrator. In addition, the COEFF routines are called a second time in ITERS in order to calculate converged values of Darcy velocities and selected flow variables and to print them out, if so desired. Comparing ITERS and ITERC, the major difference is the absence of the COEFF subroutines in ITERC. This is due to the fact that flow and transport transmissibilities need not be determined in ITERC, having been previously determined in ITERS or in ITER. Furthermore, mass balances are computed directly in ITERC, whereas in ITERS, as in ITER, such computations are deferred to PRINT2.

The treatment of a dual-porosity medium provides another means for contrasting the three integration routines. (The supporting routines for dual porosity may be recognized in each case by the beginning letter "D".) As shown in Figure 8-3, ITERS contains no facility for dual porosity, since it is recognized that the effects of dual porosity are negligible at steady state. Also, in comparing ITER and ITERC (Figures 8-2 and 8-4), it may be noted that the latter contains an alternate simplified algorithm, as indicated by the final letter "S" in the support routines. This algorithm permits greater computer efficiency for those radionuclide transport cases in which the effects of variable density, dispersion, and convection may be neglected within the rock matrix. For the primary variables, however, whose solution is obtained in ITER, the variable-density effect should be quite important in most cases, and the special facility is unnecessary.

The final item depicting the program structure is Table 8-1. There the functions of the major subroutines are given very briefly.

8.2 MATRIX SOLUTION

In terms of computational efficiency and storage requirements, the most important routines invoked in the solution of the global equations are the matrix solvers. As shown in Figures 8-2, 8-3, and 8-4, two options are available, namely Gaussian elimination and

Table 8.1. Subroutine Function.

Subroutine	Function
AMOODY	Determines the Moody friction factors as a function of Reynolds number and surface roughness so that frictional pressure drops within the wellbore may be evaluated.
COEFF I	Determines and outputs Darcy velocities.
COEFF2	Computes flow transmissibilities and implicit block-to-block convection terms.
COEFF3	Determines dispersion, heat and brine transmissibilities. explicit block-to-block dispersion terms, and explicit salt-dissolution quantities.
CRSS	Evaluates cross-derivative dispersion terms.
DCFCS	Directs setup and factorization of coefficient matrix via SOLVE for the dual-porosity blocks whenever a simplified solution for the radionuclide concentrations is desired, i.e., whenever dispersion and convection are neglected and a constant density is assumed for the rock matrix. Also performs some initialization functions for the radionuclide solutions.
DCFC	Calculates matrix transmissibilities and Darcy velocities and initializes radionuclide variables for the rock matrix for each dual-porosity block.
DCOEF	Calculates matrix transmissibilities and Darcy velocities and initializes primary-variable increments for the rock matrix in the dualporosity blocks.
DDINT	Interpolates linearly in one dimension to determine reference enthalpy.
DINIT	Initializes the pressure, temperature, brine. and radionuclide concentrations in the dualporosity blocks.
DINP	Determines amounts in place in rock matrix for the primary variables.
DITCOS	Sets up, in the dual-porosity blocks, the tridiagonal coefficient matrices for a simplified solution of the radionuclide equations in which dispersion and convection are negligible. Calls *SOLVE* to obtain matrix factorization for each dual-porosity block.
DITC1	Directs set up of tridiagonal matrices via DMQC in those global blocks containing dual porosity. Directs matrix factorization and forward elimination via *SOLVE* in order to determine fracture/matrix coupling terms.
DITC1S	Determines global sink terms, i.e., fracture/matrix coupling, with forward elimination via *SOLVE.*
DITC2	Performs back substitution via *SOLVE* for each radionuclide equation for the dual-porosity blocks in order to obtain the radionuclide concentrations within the rock matrix.
DITR1	Directs set up of tridiagonal matrices via DMQ for the primary equations in those global blocks containing dual porosity. Directs matrix factorization and forward elimination via *SOLVE* to be performed in order to determine fracture/ matrix coupling terms.
DITR2	Performs back substitution via *SOLVE* for each primary equation for the dual-porosity blocks in order to obtain distributions of the primary variables within the rock matrix.

Table 8.1. Subroutine Function (continued).

Subroutine	Function
DITR3	Updates pressure, temperature, and brine concentrations for the rock matrix of each dualporosity block.
DMQ	Sets up tridiagonal matrices for the primary variables within the rock matrix.
DMQC	Sets up tridiagonal matrices for the various radionuclide concentrations for the dualporosity blocks.
DPRNT	Outputs pressure, temperature, brine, and radionuclide concentrations for the rock matrix.
DSC	Evaluates primary-variable accumulation terms for the rock matrix in the dual-porosity blocks.
DSCC	Evaluates radionuclide accumulation terms for the rock matrix in the dual-porosity blocks.
DWTFNC	Generates space-weighting coefficients for the rock matrix.
GAUSID	Solves matrix equation for a I-D system by Gaussian elimination.
GAUS3D	Solves matrix equation for 2-D and 3-D systems by Gaussian elimination.
HLOSS	Integrates the heat-transport equations within the over/underburden.
INIT	Initializes pressures and concentrations.
ITER	Solves transient flow, heat, and brine equations by setting up matrix coefficients and invoking matrix solution.
ITERC	Solves transient-state radionuclide equations by setting up matrix coefficients and invoking matrix solution.
ITERS	Solves steady-state flow and brine equations by setting up matrix coefficients and invoking matrix solution.
LEACH	Calculates radionuclide sources from a repository based on initial inventory, leach rate, decay and solubility.
LSORX	Solves matrix equation using successive implicit solution of two neighboring lines in an x-y plane directed parallel to the x axis.
LSORY	Solves matrix equation using successive implicit solution of two neighboring lines in an x-y plane directed parallel to the y axis.
LSORZ	Solves matrix equation using successive implicit solution of two neighboring lines in a y-z plane directed parallel to the z axis.
L2SOR	Directs solution of matrix equations by the method of two-line--successive overrelaxation.
MAIN	Allocates storage and provides supervisory control of entire calculation.
MAP2D	Prints 2-D contour plots of either pressure, temperature, or concentration.
MBAL	Computes radionuclide material balance based on the initial material in place and input/output to system.
ORDER	Numbers nodes for direct Gaussian solution of the matrix equations using a D4 ordering scheme.

Table 8.1. Subroutine Function (continued).

Subroutine	Function
PRINT1	Echoes time-invariant input variables pertaining to geometrical, hydrological, and geochemical characterization of the system.
PRINT2	Outputs fluid-flow results: well summary, material-balance summary, and profiles of pressure, temperature, brine concentrations, and radionuclide concentrations.
PRODS	Sets up both implicit and explicit well terms.
READ0	Inputs radionuclide-chain parameters, including distribution coefficients.
READ0D	Inputs dimensions for the dual-porosity variables and identities the global blocks containing dual porosity.
READ1D	Inputs geometrical, hydrological, thermal, and geochemical parameters for the rock matrix in a dual-porosity simulation.
READ1	Inputs geometrical, hydrological and dispersion data. Calculates constant portion of transmissibilities.
READlA	Reads rock-type modifications and repository information, including solubilities.
READ2	Inputs time-variant fluid-flow, heat, brine, and radionuclide parameters, including those for solution control, wells, and output control.
SCOEF	Sets up matrix coefficients and load-vector terms arising from both time derivatives and salt dissolution for use in ITER.
SRHC	Determines load-vector terms for the radionuclide transport equations.
SRHP	Determines load-vector terms for the pressure and brine equations.
SOLVE	Uses the Thomas algorithm for factorization, forward elimination, and back substitution for a tridiagonal matrix.
TCALC	Determines temperature given the enthalpy and the pressures.
TODINT	Interpolates linearly in two dimensions to determine reference enthalpy.
VISL	Evaluates viscosity as a function of temperature and brine concentrations.
VISL1	Sets up viscosity-model parameters.
WELLB	Implements the well-bore submodel to relate subsurface temperature and pressure conditions to surface temperature and pressure conditions for both injection and production.
WTFNC	Computes spatial weighting parameters for either centered or backward differencing in space.

two-line successive overrelaxation. Whenever core storage is not limiting, Gaussian elimination is usually the most cost-effective solution scheme. This consideration does depend on the number of iterations required by the overrelaxation technique, as discussed in Section 8.2.2. If, however, the number of grid blocks desired is sufficiently large (greater than about 5000 grid blocks, then the overrelaxation technique is preferred.

In either event, the chosen matrix algorithm for the global solution will be a major consumer of computer resources when compared to other routines. It is therefore appropriate in this section to focus upon both of these options.

8.2.1 TWO-LINE SUCCESSIVE OVERRELAXATION

A variety of point and block-oriented successive relaxation methods have been developed. See, for example, Remson et al. (1971) and Varga (1962) and references cited therein. We have tested single and multiline techniques, concluding that the two-line L2SOR method is best. In terms of convergence, we found this method to be about 1.4 times as fast as the single-line LSOR and about twice as fast as point SOR. Further, for higher multiline methods, the gain in convergence rate was usually lost by the increased work per iteration. Thus L2SOR has been chosen for implementation.

In SWIFT for Windows the tour routines L2SOR, LSORX, LSORY, and LSORZ are used by this method (see Table 8.1). These routines perform basically three different functions, namely (1) determination of the optimal overrelaxation factor, (2) orientation of the two neighboring lines, and (3) iterative movement of these lines through the grid to achieve a convergent solution.

The first two functions are performed at intervals determined by the user-supplied parameter IMPG (Reeves et al., 1986a, p. 75). If, for example, IMPG = 5 (default value), then the optimal relaxation parameters and the optimal directions are determined at the time levels n = 1, 2, 7, 12,... for each transport equation. In each case a spectral radius ρ_{Ji} is estimated for the two-line Jacobi matrix using an iterative procedure (see references cited above). Our experience has shown that three to ten iterations provide an excellent estimate of the spectral radius. The SWIFT for Windows code prescribes internally a maximum of

$$n_\rho = 10 \tag{8-1a}$$

Such iterations are subject to a tolerance of

$$\tau_\rho = 0.02 \tag{8-1b}$$

for the spectral radius.

The optimal overrelaxation parameter ω_i is then estimated by the relation

$$\omega_i = \frac{2}{1+\sqrt{1+\rho_{Ji}}} \tag{8-2}$$

(See references cited above for a derivation of this equation.) Each time the program calculates an overrelaxation parameter the result is printed together with the required number of iterations (output parameters PARAM and ITNS). Subscript i in Equation (8-2) denotes one of the three orientations allowed by the code, as given in Table 8.2 under LSORX, LSORY, and LSORZ. Actually there are three other possible orientations. However, our experience indicates that these additional orientations are unnecessary.

Table 8.2. Two-line orientation for the L2SOR method.

Orientation	Direction of Lines	Planes Containing Lines
x	Parallel to x-axis	All nodal planes z = constant
y	Parallel to y-axis	All nodal planes z = constant
z	Parallel to z-axis	All nodal planes x = constant

Once the optimal parameters ω_i are computed, the optimal direction i is chosen to be that for which ω_i is least. The algorithm then solves implicitly for two lines of nodes at a time, treating all other nodes explicitly. Successive pairs of lines are taken until all nodes have been included explicitly. By this method, an intermediate solution $\tilde{\psi}$ is obtained. In accordance with the method of successive overrelaxation, this intermediate solution at, say, iteration level $\beta + 1$ is then coalesced with the solution at iteration level β in the following manner:

$$\psi^{\beta+1} = \psi^{\beta} + \overline{\omega}\left(\overline{\psi}^{\beta} - \psi^{\beta}\right) \tag{8-3a}$$

where $1 \le \overline{\omega} \le 2$. In assessing the optimal orientation i for implicit solution, it was found that instabilities in the convergence could be minimized if the extrapolation of Equation (8-3) proceeded no further than that of the least over-relaxation parameter

$$\overline{\omega} = \min_{1}(\omega_i) \tag{8-3b}$$

To be specific, assume that the orientation x is chosen as the optimal direction. For a given iteration level $\beta + 1$ solution begins for the z index K = 1 with simultaneous solution of the two lines J = 1 and J = 2. The pairs of lines (3, 4), (5, 6), ... are then solved. If the number of lines n_y is odd, then the K = 1 plane is completed by the solution of a single line. Of course, all nodal values are updated as they become available. Solution then proceeds through all planes K = 2, 3, ... until the entire system is covered. Convergence is assessed relative to a maximum number of iterations

$$n_\beta = 100 \tag{8-4a}$$

and a tolerance

$$\tau_\beta = 10^{-5} \tag{8-4b}$$

Both of these parameters are hard-coded at the above values in subroutine READ2, program variables HK(5) and HK(7).

8.2.2 GAUSSIAN ELIMINATION

The Gaussian-elimination procedure used by SWIFT for Windows is based on the D4 ordering scheme of Price and Coats (1973). Basically, this method consists of two things, namely an alternating diagonal ordering and a partial treatment of sparseness. This method has been shown to reduce the computer time requirements of a standard ordering scheme by factors ranging from two to six. With the D4 scheme the top half of the coefficient matrix needs no elimination, and the variable band width of the lower half matrix may be exploited to reduce the work still further.

To illustrate D4 ordering, the same 6 x 5 grid as that used by Price and Coats (1973) is presented in Figure 8-5. Figure 8-5a gives the D4 ordering of the nodes, and Figure 8-5b gives the standard ordering. Figure 8-6 then shows the corresponding matrices. As shown,

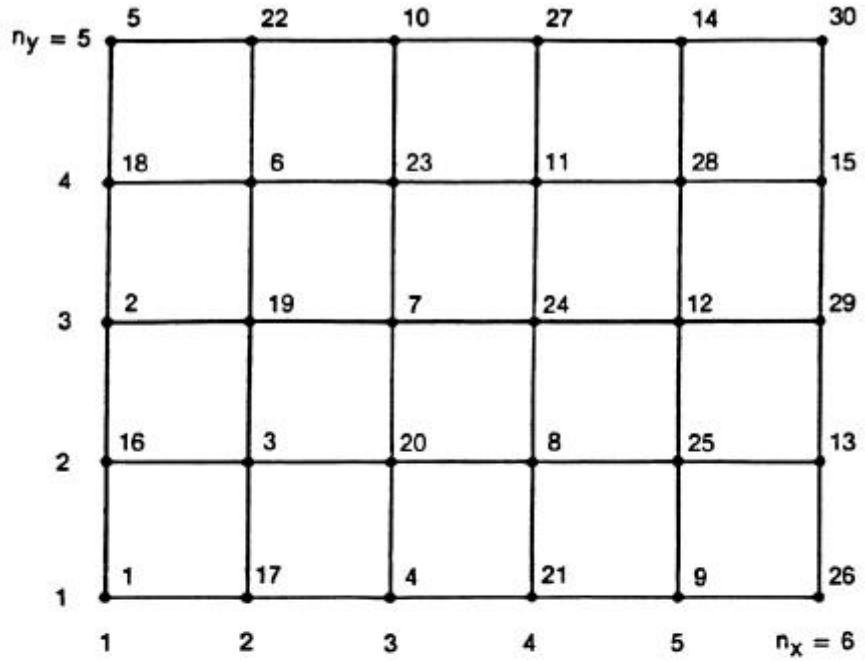
the numbering of nodes is performed along alternating diagonals for the D4 method, and numbering proceeds from the least dimension (y in this case) to the greatest (x). The idea is to perform the numbering such that connected nodes have numbers which differ by at least $n_B/2$, where n_B is the total number of nodes, or grid blocks. This procedure yields a coefficient matrix which needs no Gaussian elimination for its top half. After elimination of the elements of the lower left-hand side, i.e., those enclosed in the lower ellipse, the fill, as indicated by the zeros, produces a band about the lower half of the principal diagonal. This band has the same half-width ($n_y = 5$, for this two-dimensional case) as that obtained directly from the standard method. The D4 and standard orderings may be easily extended to three dimensions.

Two comments may be made with regard to the D4 storage requirements. First of all, the elements enclosed by ellipses in Figure 8-6a are treated by sparseness techniques and are not stored in the coefficient matrix. Only the band of the lower right-hand submatrix of order $n_B/2$ is stored. Since the maximum band widths are equal for D4 and standard orderings, the storage requirement for the former is less than half that of the latter. For example, for a square n_x x n_y system, the storage required by the D4 method is one-third that required for the standard method. Secondly, because of geometrical considerations, it is frequently desirable to block out certain portions of the finite-difference grid. An example of this procedure is given in Figure 7-2. Such unused areas are indicated in the SWIFT input by a zero pore volume. These regions are recognized internally by the code, and the coefficient matrix is then condensed automatically so that these nodes are removed. Thus such unused areas require negligible amounts of core storage.

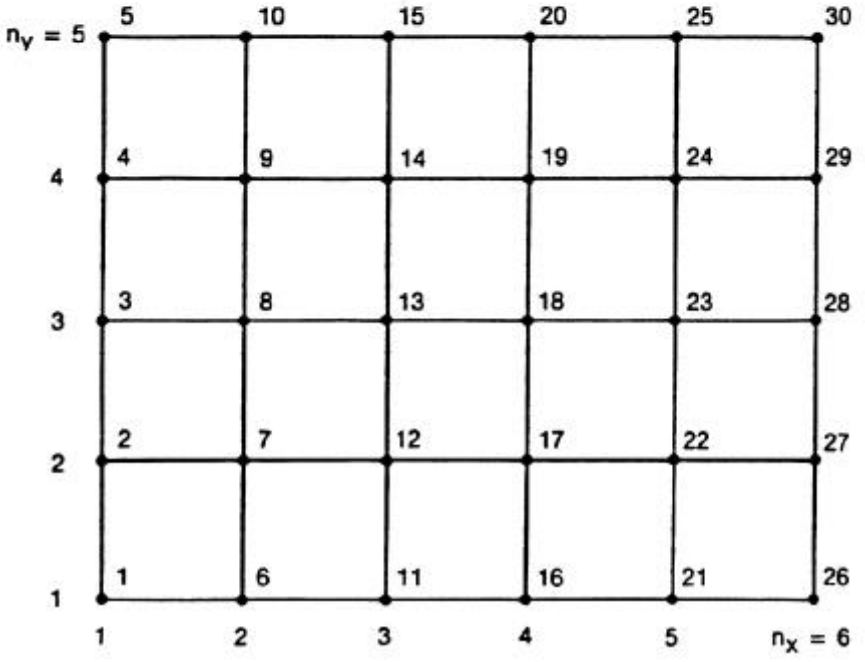
With regard to time requirements, Price and Coats compare the direct method using D4 and standard ordering to several iterative techniques. For the direct techniques, they find that the work required is

$$W_D \approx f_D n_x \left(n_y n_z^3 \right) \tag{8-5a}$$

Figure 8.5. Numerical Grids Ordered by the D4 and Standard Methods.

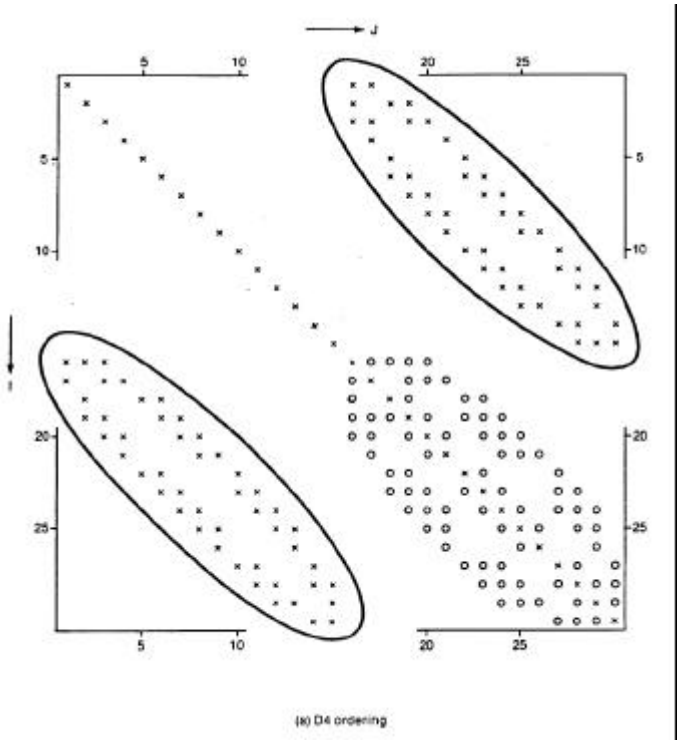


(a) D4 ordering

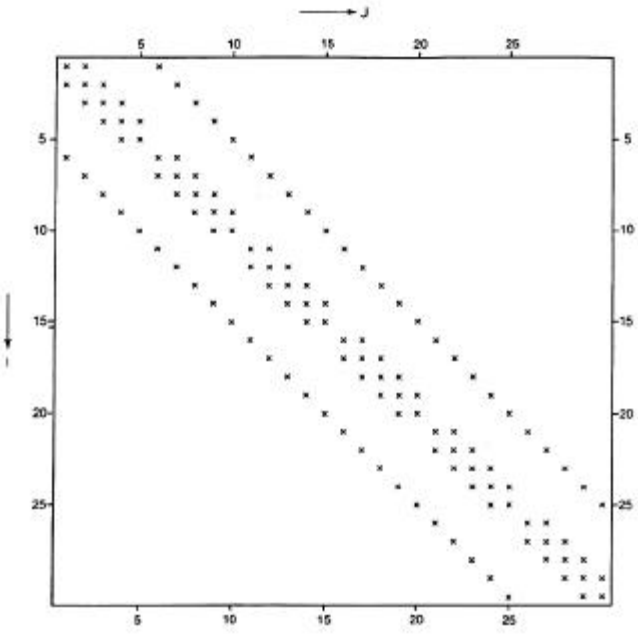


(b) standard ordering

Figure 8.6. Coefficient Matrices Resulting from D4 and Standard Orderings. The encircled matrix elements in (a) are not stored in the Coefficient Matrix.



(a) D4 ordering



(b) standard ordering

For standard ordering f_D is unity. For D4 ordering f_D is about 0.3 in value. Thus, the D4 method, in general, is about three times as fast as the standard method.

For the iterative techniques the work required is

$$W_I \approx c_I N_{it} n_x n_y n_z \qquad (8\text{-}5b)$$

The work required per iteration c_I is given in Table 8.3 for several different techniques. Furthermore, the critical iteration number N_{ito} is also given, for which the work required is the same for both D4 and the iterative technique. The iterative techniques considered are the SIP (strongly implicit) method of Weinstein et al. (1969), the ADI (alternating direction) technique of Peaceman and Rachford (1955), and Douglas and Rachford (1956) and the LSOR method of Young (1954). The basic conclusion from the above table is that for nominal band widths ($n_y n_z$) of roughly 40, the D4 solution is equivalent to about 44 LSOR iterations. Thus, if more than 44 iterations are required, the direct solution is faster.

Table 8.3. Work requirements for iterative methods.

Method	Required Work, C_I	Critical Iteration Number N_{ito}
SIP	37	$0.0081\ (n_y n_z)^2$
ADI	28	$0.0107\ (n_y n_z)^2$
LSOR	11	$0.0272\ (n_y n_z)^2$

The SWIFT for Windows code permits either D4 Gaussian elimination or L2SOR iterative solution. As a rule of thumb, for the scalar implementation used by SWIFT for Windows, it is suggested that L2SOR, which is roughly equivalent to LSOR, be used if the minimum multiple of n_x, n_y, or n_z for any two dimensions is greater than about 50. Nolen et al. (1981) discuss the vector implementation of direct and iterative techniques for two Control Data Corporation computers with built-in vector processors.

9 NOTATION

9.1 ROMAN SYMBOLS

a	thickness of a representative prism or radius of a representative sphere of the rock matrix
a_L	explicit portion of e_L
a_R	explicit portion of e_R
a_X	explicit portion of e_X
A	grading factor both for generating radial mesh in the global system and for generating mesh in the local (matrix) system
A_i	defined parameters with i = 1, 2 used for preliminary Gaussian elimination of the salt-dissolution terms
A_X	salt dissolution constants used for fluid (X = W) and brine (X = C)
A_O	accumulation parameters used in the numerical solution of radionuclide equations
b_L	implicit portion of e_L
b_R	implicit portion of e_R
b_X	implicit portion of e_X
B	viscosity parameter
B_e	scaling parameter for cumulative water flow, used in unsteady-state aquifer model
B_i	defined parameters with i = 0, 1, ..., 4 and used for preliminary Gaussian elimination of the salt-dissolution terms
c_C	coefficient for increase in fluid density with increasing brine content
c_I	required work parameter for iterative solution techniques

c_p specific heat of the fluid

c_{pR} specific heat of the rock (single porosity) or of the fracture-fill material (dual porosity)

c'_{pR} specific heat of the rock matrix

$\overline{c}_{pR}$ modified specific heat

c_R compressibility of the pores (single porosity) or of the fractures (dual porosity)

c'_R compressibility of the matrix porosity

c_T coefficient of thermal expansion

c_W compressibility of the fluid

$\hat{C}$ concentration of inert contaminant

C_{ij} coefficients of the mathematical matrix pertaining to a given block, where the indices refer to radionuclide (i, j = 0) , brine (i, j = 1), heat (i, j = 2) and pressure (i, j = 3)

$\overline{C}_{ij}$ matrix coefficients similar to C_{ij}

$\tilde{C}_{ij}$ matrix coefficients similar to C_{ij}

$\hat{C}_I$ injected brine concentration

C_{Ir} injected radionuclide concentration

C_r concentration of radioactive (trace) components

$\underline{\underline{D}}$ dispersion/diffusion tensor

D' dispersion coefficient within the rock matrix

D_m molecular diffusion

D'_m molecular diffusion within the rock matrix

D'_{mo} molecular diffusion within the rock matrix at the reference temperature

e_A boundary flow rate into the reservoir arising from a model of the aquifer which surrounds it

e_{AD} dimensionless boundary flow rate corresponding to e_A

e_L boundary heat transport to over/underburden

e_R boundary radiation transport

e_X Dirichlet boundary condition for fluid (X = W), heat (X = H), and brine (X = C)

e_O scaling parameter for the boundary flow rate, used for the steady-state aquifer model

$\underline{\underline{E}}_X$ dispersion or conduction/dispersion tensor for heat (X = H), brine (X = C) or radionuclide (X = C) within the global system

E'_X dispersion or conduction/dispersion coefficient for heat (X = H), brine (X = C), or radionuclide (X = C) within the local (rock-matrix) system

$\overline{\underline{\underline{E}}}_X$ lumped tensor corresponding to $\underline{\underline{E}}_X$ for heat (X = H), brine (X = C), or radionuclide (X = C) within the global system

f friction factor for wellbore flow

f_C fraction of solubles to total solid mass in salt-dissolution model

f_D work parameter for direct matrix solution (equals unity for a standard nodal ordering)

f_r pipe roughness factor for wellbore model

F time-dependent form factor for heat flow to the formation from the wellbore

g acceleration of gravity

g_c	units conversion factor equal to g for the English system and equal to unity for the SI system
h	depth
h_D	depth of datum below reference plane
h_I	depth from reference plane to point of the initial-pressure specification
h_T	heat-transfer coefficient
h_O	depth from reference plane to top center of gridblock (1, 1, 1)
H	fluid enthalpy
H_I	enthalpy of injected fluid
it	iteration index
$\underline{\underline{I}}$	unit tensor
$\underline{\underline{k}}$	permeability tensor for the global system
k'	permeability coefficient for the local (matrix) system
k_C	rate constant for salt dissolution
k'_{dr}	equilibrium adsorption distribution coefficient for the rock matrix and radionuclide r
k_ℓ	fractional allocation factor
k_{rs}	product of branching nuclide and daughter-parent mass fraction
K_a	thermal conductivity for annular region between tubing and casing
K_G	thermal conductivity of casing
$\underline{\underline{K}}_m$	heat conductivity tensor for fluid and rock (single porosity) or fluid and fracture-fill material (dual porosity)
K'_m	heat conductivity of fluid and rock for the rock matrix

K'_r	equilibrium retardation factor for the matrix and radionuclide r
K_R	thermal conductivity of the formation surrounding the wellbore
K_s	hydraulic conductivity of the skin surrounding a well
K_T	thermal conductivity of tubing wall
K_W	thermal conductivity of the wellbore fluid
K_O	hydraulic conductivity
L	length of wellbore from top-hole (surface) to bottom-hole elevation
m	grid-block index
m_r	density of radioactive waste, i.e., mass of nuclide r per volume of waste
M	mobility (same as M_W)
M_A	mobility used for steady-state aquifer model
M_{AD}	dimensionless mobility used for steady-state aquifer model
M_X	implicit portion of Q_X
$\overline{M}_X$	implicit portion of $\tilde{Q}_X$
M_O	number of matrix units (prisms or spheres) per global grid block
n_B	total number of grid blocks
n_i	number of increments in a given direction, where i = x, y, or z
n_s	number of nodes within the matrix
n_β	number of iterations used to determine the solution by the method of two-line successive overrelaxation (maximum of 100)
n_ρ	number of iterations used to determine the spectral radius (maximum of 10)
N	number of parent radionuclides
N_{it}	number of iterations for matrix solution

N_{ito} critical iteration number for which direct and iterative matrix solution require equal work

p pressure

p_{bh} calculated bottom-hole pressure

p_{bh}^{sp} specified bottom-hole pressure

p_D dimensionless pressure used in aquifer models

p_I initial pressure at depth h_I

$\tilde{p}_I$ total dynamic potential

p_O reference pressure for system, initial pressure for the unsteady-state aquifer model and aquifer boundary condition for the steady-state aquifer model

P_I terminal-rate influence function, used for unsteady-state aquifer model

P_r Prandtl number

q rate of fluid withdrawal

q_W radionuclide source due to waste leaching

q_X sink/source other than a well for fluid (X = W), heat (X = H), brine (X = C) and radionuclide (X = r)

q_O explicit term of fluid withdrawal rate

q^{sp} specified rate of fluid withdrawal

Q_H mass-specific heat loss from the wellbore (also see Q_X below)

Q_I terminal-pressure influence function used for unsteady-state aquifer model

Q_X total sink term, excluding boundary conditions for fluid (X = W), heat (X = H), brine (X = C), and radionuclide (H = r)

Q_{XO} explicit portion of Q_X

$\overline{Q}_X$ total sink term, including boundary conditions, for flow (X = W), heat (X = H), brine (X = C), or radionuclide (X = r)

$\overline{Q}_{XO}$ explicit portion of $\overline{Q}_X$

r radius

$\bar{r}$ equivalent grid-block radius

$\hat{r}$ grid-block edge for cylindrical coordinates

r_D dimensionless radius used in aquifer models

r_e external radius of reservoir used in aquifer models

r_G outer casing radius

r_{GI} inner tubing radius

r_q external radius of aquifer used in aquifer models

r_T inner tubing radius

r_{TO} outer tubing radius

r_w radius of wellbore

r_l radius of skin

R_C brine source rate due to salt dissolution

R'_C fluid source rate due to salt dissolution

R_e Reynolds number

$\overline{R}_e$ base-10 logarithm of Reynolds number

R_w source term for release of nuclides from the waste matrix, i.e., mass of radionuclide per bulk volume per time

R_X load-vector terms used in numerical analysis of primary and radionuclide equations in a particular grid block for fluid (X = W), heat (X = H), brine (X = C), and radionuclide (X = r)

$\overline{R}_X$ load-vector terms similar to R_X

$\tilde{R}_X$ load-vector terms similar to R_X

$R_X^{(i)}$ load-vector terms similar to R_X

s one-dimensional coordinate for the rock matrix measured from reflection boundary toward fracture/matrix interface

S saturation for free water surface

S mass of undissolved radionuclide per bulk volume

S_{fw} free water surface block saturation

S_X load vector terms used in the numerical analysis for heat (X = H) and brine (X = C)

S_1 storage required for a standard matrix ordering

S_4 storage required for a D4 matrix ordering

t time

t_a total leach time

t_b time of initialization of leach process

t_D dimensionless time used in aquifer models

t_e time-scaling variable used in aquifer models

T temperature

$\overline{T}$ net parallel-coupled transmissibility for an x-z column of grid blocks used in specifying the initial pressure from a prescribed velocity in the x direction

$\tilde{T}$ net series-coupled transmissibility in x direction used in specifying the initial pressures from a prescribed velocity in the x direction

T_G Temperature at outer edge of casing

T_I initial temperature

T_R ambient temperature of rock surrounding wellbore

T_T temperature of fluid inside tubing

$\underline{\underline{T}}_X$ transmissibility tensor for fluid (X = W), heat (X = H), brine (H = C), or radionuclide (X = C)

T'_X transmissibility coefficient within the rock matrix for fluid (X = W), heat (X = H), brine (X = C), or radionuclide (X = C)

T_O reference temperature of system, interface temperature between system and over/underburden and surface temperature for radiation model

u magnitude of $\underline{u}$

$\underline{u}$ Darcy flux vector

$\overline{u}$ average Darcy velocity

U mass-specific internal energy of the fluid

$\overline{\overline{U}}$ heat-transfer coefficient from inner wellbore radius to the formation

$\overline{\overline{U}}_G$ heat-transfer coefficient from inner wellbore radius to outer G casing radius

U_R mass-specific internal energy of the rock (single porosity) or of the fracture-fill material (dual porosity)

U'_R mass-specific internal energy of the rock matrix (dual porosity)

U_O mass-specific internal energy of the fluid at reference fluid conditions

v interstitial velocity

V pore volume

V_{fw} drainable freewater surface pore volume

V_A aquifer pore volume used for pot aquifer model

V_R rock volume

w weighting factor for time integration

W work required for solution of a matrix equation via either direct (subscript D) or iteration (subscript 1) technique

W_e cumulative water flow used for unsteady-state aquifer model

W_r solid-phase concentration of radionuclide r

W_s solid-phase concentration of radionuclides

WI well index

x,y,z Cartesian coordinates

Y_X load vector used in the numerical analysis which contains only the incremental changes in the transport terms for fluid (X = W), heat (X = H), brine (X = C), and radionuclide (X = r)

Z_I interpolation node locations for the initial temperature

Z_X load vector similar to Y used in the numerical analysis which contains only the incremental changes in the transport terms for fluid (X = W), heat (X = H), brine (X = C), and radionuclide (X = r)

9.2 GREEK SYMBOLS

α_ℓ geometrical allocation factor used for aquifer-influence functions

α_L longitudinal dispersivity

α_T transverse dispersivity

β coefficient of surface heat transfer for radiation boundary condition

$\gamma, \overline{\gamma}$ space-weighting factors

γ^* Euler constant (0.57722 ...)

Γ_X total loss to the rock matrix for fluid (X = W), heat (X = H), brine (X = C), or radionuclide (X = r)

Γ'_X source to the rock-matrix unit for fluid (X = W), heat (X = H), brine (X = C), and radionuclide (X = r)

Γ_{XO} explicit term for loss to the rock matrix for fluid (X = W), heat (X = H), brine (X = C) or radionuclide (X = r)

Γ'_{XO} explicit source term for the rock matrix for fluid (X = W), heat (X = H), brine (X = C), and radionuclide (X = r)

δ denotes incremental variable change over a time step, e.g., δp, δT, $\delta C'$, and δC

δ' fractional changes in molecular diffusivity per degree rise in temperature

δ_{ij} Kronecker delta

Δ_i differencing operator for the i-th direction where i = x, y, or z

Δh thickness

ΔH change in total head

Δs spatial increment in s

Δt time increment

Δx_i spatial increment in x_i where $x_1 = x$, $x_2 = y$, and $x_3 = z$

$\bar{\varepsilon}$ emissivity of radiative surface

ε_X convergence parameter for fluid (X = W), heat (X = H), brine (X = W), heat (X = H), brine (X = C), and radionuclide (X = r)

ζ lumping parameter for the dispersion tensor

η Freundlich isotherm parameter

η'_X matrix transport parameter related to hydraulic conduction (X = W), thermal conduction dispersion (X = H), and diffusion-dispersion (X = C)

θ_x slope of system in x direction

θ_y slope of system in y direction

θ_O black-body temperature controlling radiation

κ Freundlich isotherm parameter

$\overline{\kappa}_r$ modified Freundlich isotherm parameter

κ_R thermal diffusivity of rock surrounding the wellbore

λ decay constant

μ viscosity

μ_R viscosity parameter

ν generalized Darcy flow

ρ fluid density

ρ_I fluid density at reference temperature and pressure and unit brine concentration

ρ_{Ji} spectral radius of the Jacobi matrix for lines oriented in the direction i

ρ_N fluid density at reference temperature and pressure and zero brine concentration

ρ_R formation density

ρ_w volumetric waste density, i.e., volume of waste per bulk volume

ρ_O fluid density for the initial conditions

σ area

$\overline{\sigma}$ Stefan-Boltzmann constant

τ radioactive half-life

τ_f shear-stress term for wellbore flow

τ_L tolerance parameter for establishing convergence for the state variables

τ_β tolerance parameter for establishing convergence for two-line successive overrelaxation (10^{-5})

τ_ρ tolerance parameter for establishing convergence of the spectral-radius calculation (0.02)

ϕ porosity

ϕ_O porosity at the reference pressure

χ friction-factor exponent for the transition zone

ψ generalized state variable or time-dependent function

$\tilde{\psi}$ intermediate state variable obtained in method of successive overrelaxation

$\overline{\omega}$ minimum overrelaxation factor

ω_i overrelaxation factor for line orientation i

Ω interpolate for evaluating state variables at grid-block interfaces X

Ω_X implicit parameter for loss to rock matrix for fluid (X = W), heat (X = H), brine (X = L), and radionuclide (X = r)

Ω'_X implicit matrix source term for fluid (X = W), heat (X = H), brine (X = C), and radionuclide (X = r)

9.3 SUBSCRIPTS

A aquifer

C brine

D dimensionless

fw freewater surface

H heat or enthalphy

r radioactive component

R rock formation

w radioactive waste

W water or fluid

X generalized subscript denoting fluid, heat, brine or radionuclide

9.4 SUPERSCRIPTS

it iteration index

n time-step indicator

α index for inner wellbore iteration on enthalpy equation

β index for intermediate wellbore iteration on pressure equation

γ index for outer wellbore iteration on boundary conditions

10 REFERENCES

Aziz, K., and A. Settari, Petroleum Reservoir Simulation, Applied Science Publishers, Ltd., 1979.

Bear, J., and C. Braester, Simultaneous Flow of Immiscible Liquids in a Fractured Medium, from Fundamentals of Transport Phenomena in Porous Media, Elsevier Publishing Company, pp. 177-202, 1972.

Bear. J., Hydraulics of Groundwater, McGraw-Hill, New York, 1979.

Bell, M. J., ORIGEN - The ORNL Isotope Generation and Depletion Code, ORNL-4628, Oak Ridge National Laboratory, Oak Ridge, TN, 1973.

Bredehoeft, J. D., and G. F. Pinder, Mass Transport Flowing Ground-Water, Water Resources Research, 9, 194-209, 1973.

Campbell, J. E., R. T. Dillon, M. S. Tierney, H. T. Davis, P. E. McGrath, F. J. Pearson, Jr., H. Shaw, J. C. Helton, and F. A. Donath, Risk Methodology for Geologic Disposal of Radioactive Wastes: Interim Report. NUREG/CR-0458, SAND78-0029, 1978.

Campbell, J. E., D. E. Longsine, and M. Reeves, Distributed Velocity Method of Solving the Convection-Dispersion Equation, Advances in Water Resources, 4, 102-115, 1981.

Carslaw, H. S., and J. C. Jaeger, Conduction of Heat in Solids, Second Edition, Oxford University Press, 1959.

Carter, R. D., and C. W. Tracy, An Improved Method for Calculating Water Influx, Transactions of the Society of Petroleum Engineers, American Institute of Mining Engineers, 219, 415-417, 1960.

Christie, I., D. F. Griffiths, and A. R. Mitchell, Finite Element Methods for Second Order Differential Equations with Significant First Derivatives, International Journal of Numerical Methods in Engineering, 10, 1289-1396, 1976.

Cooper, H. , The Equation of Ground-Water Flow in Fixed and Deforming Coordinates, Journal of Geophysical Research, 71, 4783-4790, 1966.

Dillon, R. T., R. B. Lantz, and S. B. Pahwa. Risk Methodology for Geologic Disposal of Radioactive Waste: The Sandia Waste-Isolation Flow and Transport (SWIFT) Model, NUREG/ CR-0424, SAND78-1267, Sandia National Laboratories, Albuquerque, NM, 1978.

Douglas, J. Jr., and H. H. Rachford. Jr., On The Numerical Solution of Heat Conduction Problems in Two or Three Space Variables, Transactions of the American Math Society, 82, 1956.

Duguid, J. 0., and M. Reeves, A Comparison of Mass Transport Using Average and Transient Rainfall Boundary Conditions, in Finite Elements in Water Resources, Ed. by W. G. Gray, G. F. Pinder, and C. A. Brebbia, pp. 2.25 - 2.35, 1976.

Evans, R.D., The Atomic Nucleus, McGraw-Hill, New York, 1955.

Finley, N. C. and M. Reeves, SWIFT Self-Teaching Curriculum: Illustrative Problems to Supplement the User's Manual for the Sandia Waste-Isolation Flow and Transport Model (SWIFT), NUREG/CR-1968, SAND81-0410, Sandia National Laboratories, Albuquerque, NM, 1981.

Garder, A. O., D. W. Peaceman, and A. L. Pozzi, Numerical Calculation of Multidimensional Miscible Displacement by Method of Characteristics, Society of Petroleum Engineers Journal, 4, 26-36, 1964.

Grisak, G. E., and J. F. Pickens, Solute Transport Through Fractured Media 1. The Effect of Matrix Diffusion, Water Resources Research, 16(4), 719-730, 1980.

Huyakorn, P. S., B. H. Lester, and J. W. Mercer, An Efficient Finite Element Technique for Modeling Transport in Fractured Porous Media 1. Single Species Transport, Water Resources Research, 19(3), 841-854, 1983.

INTERCOMP Resource Development and Engineering, Inc., Development of Model for Calculating Disposal in Deep Saline Aquifers, Parts I and II, USGS/WRI-76-61, PB-256903, National Technical Information Service, Washington, DC, 1976.

Katz, D. L., M. R. Tek, K. H. Coats, M. L. Katz, S. C. Jones, and M. C. Miller, Movement of Underground Water in Contact with Natural Gas, AGA Monograph on Project No. 31, American Gas Association, New York, 1963.

Katz, D. L., and K. H. Coats, Underground Storage of Fluids, Ulrich Books., Inc., Ann Arbor, MI, 1968.

Kennedy, G. C., and W. T. Holser, Pressure-Volume Temperature and Phase Relations of Water and Carbon Dioxide, Section 16 of: Clark, Sydney, P., Jr., ed., Handbook of Physical Constants, revised edition, Geological Society of America, Memoir 97, 1966.

Lantz, R. B., Quantitative Evaluation of Numerical Diffusion (Truncation Error), Society Petroleum Engineers Journal, 315-320, 1971.

Lehman, L. L., and Quinn, E. J., private communication, 1982.

Lewis, W. K., and Squires, L., Generalized Liquid Viscosity Curve, Oil and Gas Journal, 33, 92-92, 1934.

McAdams, W. H., Heat Transmission, McGraw-Hill, New York, 1955.

McCabe, W. L. , and J. C. Smith, Unit Operations in Chemical Engineering, McGraw-Hill, 1976.

McDonald, M. G. and A. W. Harbaugh, A Modular Three-Dimensional Finite-Difference Ground Water Flow Model. US Geological Survey, Open-File Report 83-875, 1984.

Muller, A. B. , N. C. Finley, F. J. Pearson. Jr. , Geochemical Parameters Used in the Bedded Salt Reference Repository Risk Assessment Methodology, NUREG/CR-1996, SAND81-0557, Sandia National Laboratories, Albuquerque, NM, 1981.

Nolen, J. S. , and D. W. Berry, Tests of the Stability and Time-step Sensitivity of Semi-Implicit Reservoir Simulation Techniques, Society of Petroleum Engineers, Paper No. 2981, 253-266, 1972.

Nolen, J. S., G. Van Hartlemann, S. Meister, W. Kleinitz, and Heiblinger, Jr., Numerical Simulation of the Solution Mining Process, Society of Petroleum Engineers, Paper No. SPE 4850, 1974.

Nolen, J. S., D. W. Kuba, and M. J. Kascic, Jr. , Application of Vector Processors to Solve Finite Difference Equations, Society of Petroleum Engineers, Paper No. SPE 7675, 1981.

Peaceman, D. W., and H. H. Rachford, Jr., The Numerical Solution of Parabolic and Elliptic Differential Equations, Journal of the Society of Industrial Applied Math, 3, 1955.

Pinder, G. W., A Galerkin-Finite Element Simulation of Ground-Water Contamination on Long Island, New York, Water Resources Research, 9(6), 1657-1669, 1973.

Preuss, K., and T. N. Narasimhan, A Practical Method for Modeling Fluid and Heat Flow in Fractured Porous Media, Society of Petroleum Engineers, Paper No. SPE 10509, 1982.

Price, H. S., J. S. Cavendish, and R. S. Varga, Numerical Methods of Higher-Order Accuracy for Diffusion Convection Equations, Society of Petroleum Engineers Journal, September, 1968.

Price, H. S., and K. H. Coats, Direct Methods in Reservoir Simulation, Society of Petroleum Engineers Journal, Paper No. 4278, 14(3), 295-308, 1973.

Pritchett, J. W., and S. K. Garg, Determination of Effective Well Block Radii for Numerical Reservoir Simulations, Water Resources Research, 16(4), 665-674, 1980.

Ramey, H. J., Jr., Wellbore Heat Transmission, Transactions of the American Institute of Mining Engineers, 427-435, April, 1962.

Rasmuson, A., T. N. Narasimhan, and I. Neretnieks, Chemical Transport in a Fissured Rock: Verification of a Numerical Model, Water Resources Research, 18(5), 1479-1492, 1982.

Reddell, D. L., and D. K. Sunada, Numerical Simulation of Dispersion in Groundwater Aquifers, Hydrology Papers, Number 41, Colorado State University, 1970.

Reeves, M., and R. M. Cranwell, User's Manual for the Sandia Waste-Isolation Flow and Transport Model (SWIFT), Release 4.81, NUREG/CR-2324, SAND81-2516, Sandia National Laboratories, Albuquerque, NM, 1981.

Reeves, M., D. S. Ward, N. D. Johns, and R. M. Cranwell, Data Input Guide for SWIFT II, The Sandia Waste-Isolation Flow and Transport Model for Fractured Media, Release 4-84, NUREG/CR-3162, SAND83-0242, Sandia National Laboratories, Albuquerque, NM, 1986a.

Reeves, M., D. S. Ward, P. A. Davis, and E. J. Bonano, SWIFT II Self-Teaching Curriculum: Illustrative Problems for the Sandia Waste-Isolation Flow and Transport Model for Fractured Media, NUREG/CR-3925, SAND84-1586, Sandia National Laboratories, Albuquerque, NM, 1986b.

Reeves, M., D.S. Ward, N.D. Johns, R.M. Cromwell, Theory and Implementation for SWIFT II, The Sandia Waste-Isolation Flow and Transport Model for Fractured Media, Release 4.84, NUREG/CR-3328, SAND83-1159, Sandia National Laboratories, Albuquerque, NM, 1986c.

Remson, I., G. M. Hornberger, and F. J. Molz, Numerical Methods in Subsurface Hydrology, Wiley Interscience, New York, 1971.

Schilthius, R. J., Active Oil and Reservoir Energy, Transactions of the American Institute of Mining Engineers, 118, 33, 1936.

Schlichting, H., Boundary Layer Theory, McGraw-Hill, 1955.

Sikora, V. J., Water-Drive Oil Reservoirs, Chapter 35, in Petroleum Production Handbook, Vol. II, Reservoir Engineering, Frick, T. C., and R. W. Taylor, eds., SPE-AIME, Dallas, TX, 1962.

Sorey, M. L., Numerical Modeling of Liquid Geothermal Systems, US Geological Survey, Professional Paper 1044-D, 1978.

Streltsova-Adams, T. D., Well Hydraulics in Heterogeneous Aquifer Formations, Advances in Hydroscience, 11, 357423, 1978.

Tang, D. H., E. 0. Frind, and E. A. Sudicky, Contaminant Transport in Fractured Porous Media 1. Analytical Solution for a Single Fracture. Water Resources Research, 17(3), 555-564, 1981.

Van Everdingen, A. F., and W. Hurst, Application of the Laplace Transformation to Flow Problems in Reservoirs, Transactions of the Society of Petroleum Engineers, American Institute of Mining Engineers, 186, 305-324, 1949.

Varga, R. S., Matrix Iterative Analysis, Prentice-Hall, Englewood Cliffs, NJ, 1962.

Ward, D. S., M. Reeves, and L. E. Duda, Verification and Field Comparison of the Sandia Waste-Isolation Flow and Transport Model (SWIFT), NUREG/CR-3316, SAND83-1154, Sandia National Laboratories, Albuquerque, NM, 1984.

Weinstein, H. G., H. L. Stone, and T. V. Kwan, Iterative Procedure for Solutions of Systems of Parabolic and Elliptic Equations in Three Dimensions, I&EC Fundamentals, 8(2), May, 1969.

Young, D. M., Iterative Methods for Solving Partial Differential Equations of Elliptic Type, Transactions of the American Math Society, 76, 1954.