530SW86006

EPA/530-SW-86-006
April 1986

SOILINER Model - Documentation and User's Guide (Version 1)

Contract No. 68-01-6871
Assignment No. 48

In Cooperation with

Office of Solid Waste
U.S. Environmental Protection Agency
Washington, D.C. 20460

Hazardous Waste Engineering Research Laboratory
Office of Research and Development
U.S. Environmental Protection Agency

Cincinnati, Ohio 45268

DISCLAIMER

This report was prepared by Russell A. Johnson, Eric S. Wood, and Richard J. Wozmak of GCA Technology Division, Inc., Bedford, Massachusetts 01730, under Contract No. 68-01-6871. The technical project monitors were Les Otte of the Office of Solid Waste and Douglas Ammon of the Hazardous Waste Engineering Research Laboratory. This is a draft report that is being released by EPA for public comment on the accuracy and usefulness of the information in it. The report has received extensive technical review but the Agency's peer and administrative review process has not yet been completed. Therefore it does not necessarily reflect the views or policies of the Agency. Mention of trade names or commercial products does not constitute endorsement or recommendation for use.

FOREWORD

The Enviromental Protection Agency was created because of increasing public and governmental concern about the dangers of pollution to the health and welfare of the American people. Noxious air, foul water, and spoiled land are tragic testimony to the deterioration of our natural environment. The complexity of the environment and the interplay of its components require a concentrated and integrated attack on the problem.

Today's rapidly developing and changing technologies and industrial products and practices frequently carry with them the increased generation of solid and hazardous wastes. These materials, if improperly dealt with, can threaten both public health and the environment. Abandoned waste sites and accidental releases of toxic and hazardous substances to the environment also have important environmental and public health implications. The Hazardous Waste Engineering Research Laboratory assists in providing an authoritative and defensible engineering basis for assessing and solving these problems. Its products support the policies, programs, and regulations of the Environmental Protection Agency; the permitting and other responsibilities of State and local governments; and the needs of both large and small businesses in handling their wastes responsibly and economically.

The Office of Solid Waste is responsible for issuing regulations and guidelines on the proper treatment, storage, and disposal of hazardous wastes, in order to protect human health and the environment from the potential harm associated with improper management of these wastes. These regulations are supplemented by guidance manuals and technical guidelines, in order to assist the regulated community and facility designers to understand the scope of the regulatory program. Publications like this one provide facility designers with state-of-the-art information on design and performance evaluation techniques.

This document describes technical procedures for determining adequate thicknesses of compacted soil liners to prevent migration of hazardous constituents through the liner during the active life and post-closure care period. It includes a performance simulation model that is based on numerical techniques.

Marcie Williams
Director, Office of Solid Waste

Thomas R. Hauser
Director, Hazardous Waste Engineering Research Laboratory

PREFACE

Subtitle C of the Resource Conservation and Recovery Act (RCRA) requires the U.S. Environmental Protection Agency (EPA) to establish a Federal hazardous waste management program. This program must ensure that hazardous wastes are handled safely from generation until final dispostion. EPA issued a series of hazardous waste regulations under Subtitle C of RCRA that are published in 40 Code of Federal Regulations (CFR) 260 through 265 and 122 through 124.

Parts 264 and 265 of 40 CFR contain standards applicable to owners and operators of all facilities that treat, store, or dispose of hazardous wastes. Wastes are identified or listed as hazardous under 40 CFR Part 261. Part 264 standards are implemented through permits issued by authorized States or EPA according to 40 CFR Part 122 and Part 124 regulations. Land treatment, storage, and disposal (LTSD) regulations in 40 CFR Part 264 issued on July 26, 1982, establish performance standards for hazardous waste landfills, surface impoundments, land treatment units, and waste piles.

EPA is developing three types of documents for preparers and reviewers of permit applications for hazardous waste LTSD facilities. These types include RCRA Technical Guidance Documents, Permit Guidance Manuals, and Technical Resource Documents (TRD's).

The RCRA Techical Guidance Documents present design and operating specificatons or design evaluation techniques that generally comply with, or demonstrate compliance with the Design and Operating Requirements and the Closure and Post-Closure Requirements of Part 264.

The Permit Guidance Manuals are being developed to describe the permit application information the Agency seeks, and to provide guidance to applicants and permit writers in addressing information requirements. These manuals will include a discussion of each set of specifications that must be considered for inclusion in the permit.

The Technical Resource Documents present state-of-the-art summaries of technologies and evaluation techniques determined by the Agency to constitute good engineering designs, pratices, and procedures. They support the RCRA Technical Guidance Documents and Permit Guidance Manuals in certain areas (i.e., liner, leachate management, closure covers, and water balance) by describing current technologies and methods for designing hazardous waste facilties, or for evaluating the performance of a facility design. Although emphasis is given to hazardous waste facilities, the information presented in these TRD's may be used for designing and operating nonhazardous waste LTSD facilities as well. Whereas the RCRA Technical Guidance Douments and Permit Guidance Manuals are directly related to the regulations, the information in these TRD's covers a broader perspective and should not be used to interpret the requirements of the regulations.

This document is a draft being made available for public review and comment. It has undergone review by recognized experts in the technical areas covered, but Agency peer review processing has not yet been complete. Public comment is desired on the accuracy and usefulness of the information presented in this document. Comments received will be evaluated, and suggestions for improvement will be incorporated, wherever feasible, before pulication of the next edition. Communications should be addressed to Docket Clerk, Room S-212(A), Office of Solid Waste (WH-562), U.S. Environmental Protection Agency, 401 M Street, S.W., Washington, D.C., 20460. The document under discussion should be identified by title and number; e.g., "SOILINER Model - Documentation and User's Guide (Version 1)" (EPA/530-SW-86-006).

ABSTRACT

This Technical Resource Document provides documentation and a User's Guide for the SOILINER computer model. SOILINER is a finite-difference approximation of the highly nonlinear, governing equation for one-dimensional, unsaturated flow in the vertical dimension. SOILINER was designed to simulate the dynamics of an infiltration event across a compacted soil liner system beneath impounded liquid. Since the governing equation reflects liner heterogeneity and the dependence of liner properties on the degree of saturation, SOILINER is capable of accurately representing infiltration for a variety of soil(clay)liner scenarios. Important features inherent to the SOILINER model include the ability to simulate: (1) multilayered system, (2) variable initial moisture content, and (3) changing conditions on the boundaries of the compacted soil liner flow domain. Due to these features, SOILINER serves as a comprehensive tool for the design of liner configurations, specifically liner conductivity and thickness.

ACKNOWLEDGEMENTS

The authors would like to thank Dan Goode (now with the Nuclear Regulatory Commission) and Dr. Michael Mills of GCA for their contributions to the SOILINER Documentation and Users' Guide. SOILINER, a finite difference model originally authored at GCA by Dan Goode, stood the test of time during our extensive sensitivity analysis with very little modification. We were continually surprised by the versatility of this model and the role it plays as a tool in understanding the dynamics of an infiltration event. Also, the original SOILINER document (Goode and Smith, 1984) was clear and precise, and thus served as the basis for Part I of this report, where only minor changes were made.

Dr. Mills provided technical guidance on many aspects of our most recent effort to review and modify SOILINER. We acknowledge his invaluable suggestions during development of new model features and his efforts to periodically review the contents of this document. His accessibility and eagerness to help is greatly appreciated and contributed immensely to the production of the SOILINER Documentation and Users' Guide.

TABLE OF CONTENTS

TABLE OF CONTENTS

FIGURES

FIGURES

TABLES

PART I

SOILINER DOCUMENTATION

1. INTRODUCTION

On November 8, 1984 the Hazardous and Solid Waste Amendments of 1984 (HSWA) were signed into law. Among the provisions of this new law are minimum technological requirements for hazardous waste landfills, surface impoundments, and waste piles. These provisions are found in Sections 3004(o) and 3015 of the Resource Conservation and Recovery Act (RCRA) and further defined in the Part 264 regulation and associated guidance. Any new surface impoundment must install two or more liners and a leachate collection system between such liners. The lower liner must be designed, operated, and constructed to prevent breakthrough of constituents over the period of operation, including any post-closure care period.

Existing methods of evaluating flow through soil liners have been reviewed, including the: (1) transit time equation, (2) Green-Ampt wetting front model, and (3) transient, linearized, approximate model - see Appendix A of the Technical Resource Document (TRD-EPA/530-SW-84-001). Simplifying assumptions associated with these methods are fairly restrictive and do not allow for an accurate simulation of the infiltration event.

SOILINER is a finite-difference approximation of the highly nonlinear, governing equation of unsaturated, vertical flow. Since the governing equation reflects liner heterogeneity and the dependence of liner properties on the degree of saturation, SOILINER is capable of accurately representing infiltration for a variety of compacted soil liner scenarios. Important features inherent to the SOILINER model include the ability to simulate: (1) multilayered systems, (2) variable initial moisture content, and (3) changing conditions on the boundaries of the compacted soil liner flow domain.

This report is comprised of two parts - documentation of the SOILINER model and a User's Guide. Part I discusses the conceptual model of flow and its mathematical representation (Section 2); the SOILINER finite-difference formulation (Section 3); and model verification (Section 4). Part II presents the modeling capabilities (Section 2), data requirements (Section 3), and procedure for applying SOILINER (Section 4) with examples.

2. CONCEPTUAL MODEL OF FLOW THROUGH LINER

2.1 Description of Physical Problem

The flow domain for liner breakthrough (shown in Figure 2-1) consists of the following: a layer of liquid in the impoundment of depth h_l (L); a compacted soil liner of thickness d (L); a layer of underlying site soil, which may or may not be saturated, to a depth z_w(L); and a constantly saturated ground water layer of the same site soil. For this study, only vertical processes are considered.

A schematic of the initial liquid pressure distribution (ψ) is shown in Figure 2-2. Before installation of the liner, the soil moisture at the site can be assumed to be in static equilibrium with the underlying water table and saturated zone. Departures from this condition can occur if there is significant evaporation from or recharge to the water table, and these departures can be easily quantified. The soil liner is installed on top of the site soil and is compacted. The liner is homogeneous and hence has an initially constant moisture content and constant pressure over its entire thickness.

After the impoundment is filled, the flow system is not in equilibrium, and liquid will flow vertically down from the impoundment into the liner, and eventually into the site soil and saturated ground water zone. Our goals are to simulate this flow and to predict the liner thickness required to prevent leachate from migrating through the liner into the subsoils during the active life and post-closure care period of the unit.

2.2 Mathematical Statements

2.2.1 Governing Equation of Vertical Unsaturated Flow--

The governing equation for transient, unsaturated flow in the vertical direction can be written:

$$\frac{\partial\psi}{\partial t}\frac{d\theta}{d\psi} - \frac{\partial}{\partial z}\left[K(\psi)\frac{\partial\psi}{\partial z} + K(\psi)\right] = 0 \qquad (2\text{-}1a)$$

and, for steady state:

$$\frac{\partial}{\partial z}\left[K(\psi)\frac{\partial\psi}{\partial z} + K(\psi)\right] = 0 \qquad (2\text{-}1b)$$

in which $\psi \simeq \phi - z$ [L] is matric potential or capillary pressure head, where ϕ [L] is piezometric head; θ [-] is volumetric moisture content; $K(\psi) = K_r(\psi)\ K_s$ [LT^{-1}] is unsaturated hydraulic conductivity, where $K_r(\psi)$ [-] is relative hydraulic conductivity, and K_s [LT^{-1}] is saturated hydraulic conductivity; z [L] is the vertical coordinate, negative downward; and t [T] is time. This equation is developed by consideration of conservation of mass. The first term represents the change in storage of liquid mass and the second term is mass flux divergence, or the change in flux over space. The second term contains fluxes due to the matric potential gradient ($K(\psi)\ \partial\psi/\partial z$) and gravitational potential $K(\psi)$. The flux term is developed from the generalization of Darcy's law for water flow in porous media:

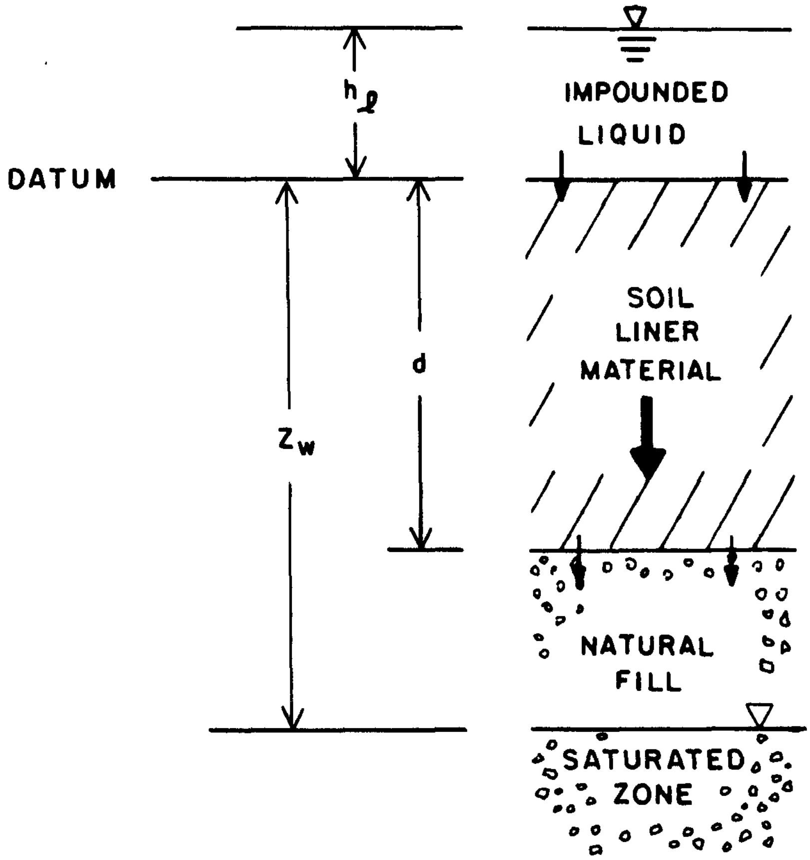

Figure 2-1. Flow domain for liner breakthrough.

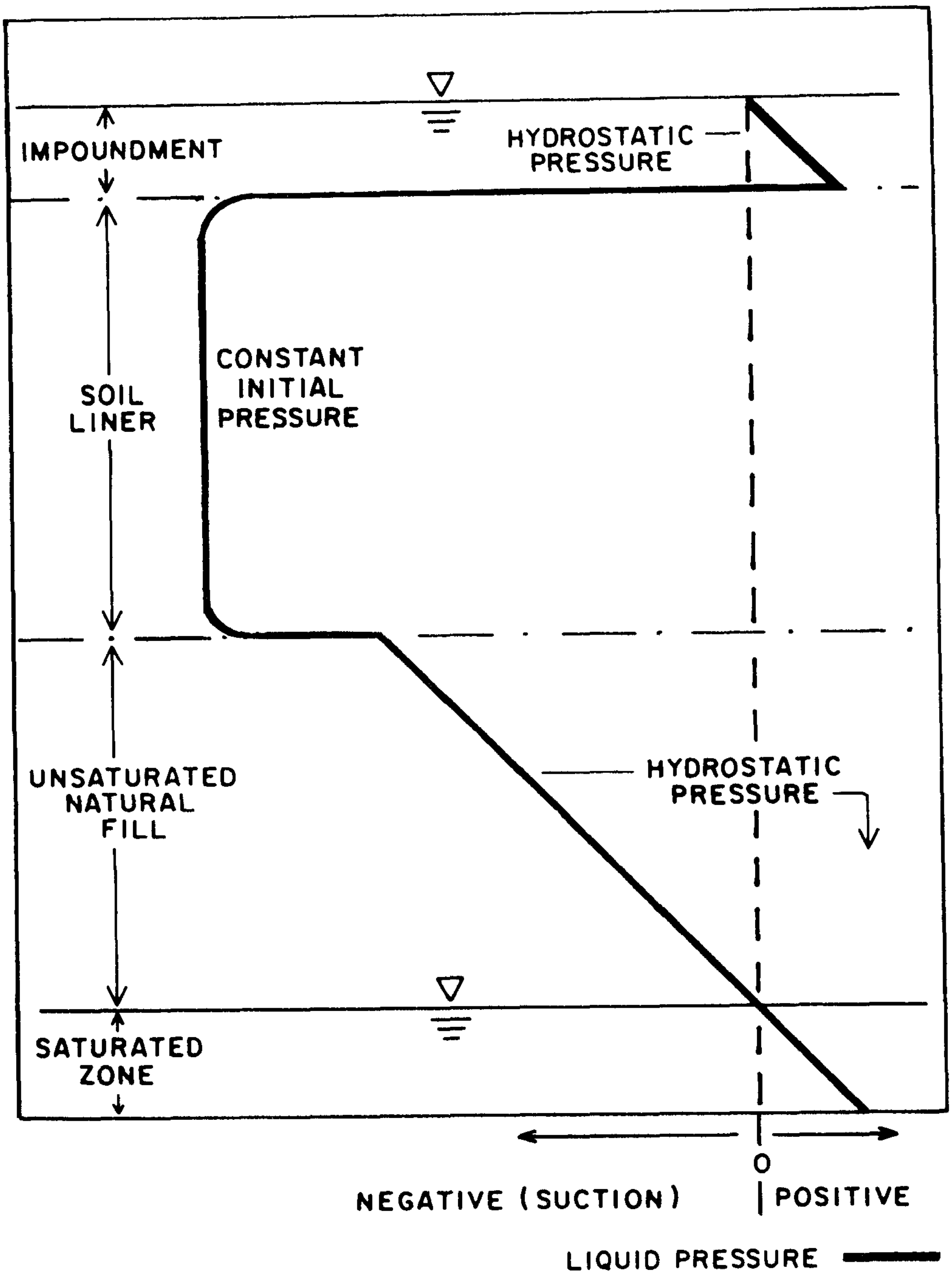

Figure 2-2. Schematic of initial capillary liquid pressure distribution.

$$q = -K(\psi)\frac{\partial\phi}{\partial z} = -K(\psi)\frac{\partial(\psi+z)}{\partial z} = -K(\psi)\left[\frac{\partial\psi}{\partial z} + 1\right] \qquad (2\text{-}2)$$

in which q $[LT^{-1}]$ is flux. Some assumptions implicit in (2-1) are that the fluid has constant density and does not freeze, that the medium does not deform, that the air phase is always at a spatially constant atmospheric pressure, and that the water flow is unaffected by temperature gradients or by solute concentration gradients.

Equation (2-1a), derived for unsaturated flow, is also applicable to modeling temporarily or permanently saturated soil zones. In that case, ψ is known as the pressure head, $d\theta/d\psi$ becomes the specific storativity, and $K(\psi)$ is equal to K_s.

2.2.2 Boundary Conditions--

At the top of the liner, z=0, the pressure head is controlled by the level of liquid in the impoundment (see Figure 2-2). In terms of piezometric head, the matric potential at the liner top is fixed:

$$\psi = \phi_{z=0} - z \qquad \text{at } z=0 \qquad (2\text{-}3)$$

Since the piezometric head is constant in space (hydrostatic) in the ponded liquid,

$$\phi_{z=0} = \phi_{z=h} = h_\ell \qquad (2\text{-}4)$$

and, since z=0, (2-3) becomes:

$$\psi = \phi_{z=0} - 0 = h_\ell \qquad \text{at } z=0 \qquad (2\text{-}5)$$

This is the fixed pressure Dirichlet boundary condition applied at the top of the liner.

By definition, the matric potential at the water table is equal to zero; thus at the column bottom, $z = z_w$ (where z_w is a negative value - see Figure 2-1), the Dirichlet boundary condition is:

$$\phi = z_w \text{ or } \psi = 0 \qquad \text{at } z = z_w \qquad (2\text{-}6)$$

This water table boundary condition is assumed to be controlled by local ground water flow and to be unaffected by the amount of liquid discharging through the liner. That is, the water table elevation is assumed to be constant.

The matric potential matching condition between the liner soil and the site soil is [see Bear, 1979, p. 206]:

$$\psi = \psi_s \qquad \text{at } z = -d \qquad (2\text{-}7)$$

where ψ and ψ_s, refer to the matric potential of the liner and site soil respectively. This is simply a condition of pressure continuity. This condition is incorporated into the governing equation (2-1a) and is implicitly satisfied.

2.3 Soil Moisture Characteristics

Unsaturated flow is subject to negative pressures (also called suction) and this potential constitutes a moving force (Hillel, 1971). The matric suction is due to capillary and adsorptive forces which tend to move water from where the suction is low to where it is high.

In a saturated soil at equilibrium, the suction is zero at the free water surface. Away from the free water surface, interstitial pore spaces are capable of remaining saturated under slight suctions, an example of this being the capillary fringe of a given soil. However, as suction is increased beyond a critical value, the largest continuous pores begin to empty. This critical suction is called the air entry pressure and is a function of soil type. Typically, the air entry pressure is small in coarse-textured, well-aggregated soils. As suction is further increased, more water is gradually taken out of the soil as the larger pores drain first, followed by progressively smaller pores. At some high suction value only the very narrow pores will retain water. Increasing suction is thus associated with decreasing moisture content. This function is represented graphically by a curve known as a soil-moisture characteristic curve, which relates moisture content to pressure.

Figure 2-3 shows characteristic curves for a sandy and a clayey soil. It has been observed experimentally that the soil moisture content matric potential relationship is hysteretic; it has a different shape when soils are wetting than when they are drying. If a soil sample was saturated at a pressure greater than zero and the pressure was then lowered in a stepwise fashion until it reached a level much less than zero, the moisture content at each step would follow the drying curve. If water was added to the soil sample, the pressure would follow the wetting curve. If the soil sample was only partially wetted or dried, the pressure would follow internal lines which are called scanning curves.

To evaluate the storage term of the governing equation (2-1), we must define the soil specific moisture capacity $C(\psi)$ $[L^{-1}]$:

$$C(\psi) = \frac{d\theta}{d\psi} \tag{2-8}$$

which is the slope of the moisture characteristic curve in Figure 2-3, and is calculated as the derivative of the moisture retention function:

$$\theta = \theta(\psi). \tag{2-9}$$

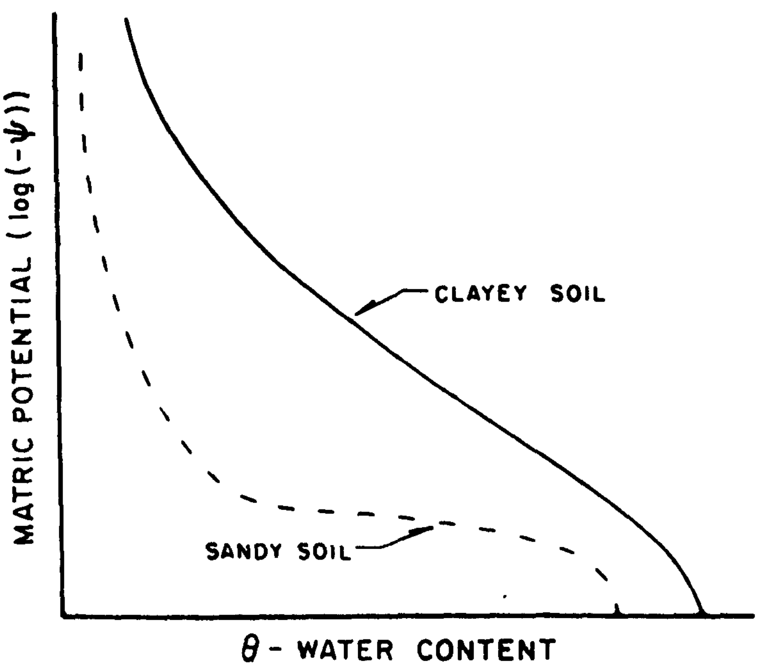

Figure 2-3. The effect of texture on soil moisture characteristics (after Hillel, 1971).

Several mathematical expressions have been proposed for the moisture retention function, or characteristic curve. Brooks and Corey [1964] collected moisture retention data on many granular media and developed a power law relationship for the drainage cycle:

$$\theta(\psi) = \begin{cases} n & \psi > \psi_b \\ \theta_r + (n - \theta_r)\left(\frac{\psi}{\psi_b}\right)^{-\lambda} & \psi \le \psi_b \end{cases} \tag{2-10}$$

in which θ_r [-] is residual (nonreducible) moisture content, ψ_b [L] is bubbling or air entry pressure at which air first enters the draining column, n [-] is porosity, and λ is the pore size distribution coefficient. The experimental data to develop this model included pressures of $0 \ge \psi \ge -500$ cm. The corresponding specific moisture capacity, which is the derivative of $\theta(\psi)$, is:

$$C(\psi) = \begin{cases} 0 & \psi \ge \psi_b \\ -\lambda(n-\theta_r)\frac{1}{\psi}\left|\frac{\psi}{\psi_b}\right|^{-\lambda} & \psi < \psi_b \end{cases} \tag{2-11}$$

This function is discontinuous at saturation, $\psi = \psi_b$. Clapp and Hornberger (1978) designed a model using a power curve and a short parabolic section near saturation to represent gradual air entry. This two part function was used to relate soil moisture and matric potential for a wetting soil. The power curve representing the moisture characteristic for $W \le 0.92$ is:

$$\psi = \psi_s W^{-b}$$

with soil wetness, W, equal to θ/θ_s where θ_s is the saturated water content. Both ψ_s, the saturation matric potential, and the exponent b are empirical and must be estimated. For $0.92 \le W \le 1$, the moisture characteristic can be described by the parabola:

$$\psi = -m(W-n)(W-1)$$

where

$$m = \frac{\psi_i}{(1-W_i)^2} - \frac{\psi_i b}{W_i(1-W_i)}$$

and

$$n = 2W_i - \frac{\psi_i b}{mW_i} - 1$$

ψ_i is the matric potential at $W = 0.92$.

King [1965] fit a hyperbolic function to moisture retention data for several soil types with pressures as low as -100 cm:

$$\theta = n\,\delta \left[\frac{\cosh\,[(\psi/\psi_o)^{\beta} + \varepsilon] - \gamma}{\cosh\,[(\psi/\psi_o)^{\beta} + \varepsilon] + \gamma} \right] \qquad (2\text{-}12)$$

in which δ, ψ_o, β, ε, and γ are fitted parameters. This model was used by Gillham et al. [1976] to provide data for numerical computations.

Rogowski [1971] proposed a simple model requiring few input parameters:

$$\begin{aligned} \theta &= n + \alpha \ln(\psi_b - \psi + 1) \qquad & \psi \le \psi_b \\ \theta &= n & \psi > \psi_b \end{aligned} \qquad (2\text{-}13)$$

in which

$$\alpha = (\theta_{15} - n) \,/\, \ln(\psi_b - \psi_{15} + 1) \qquad (2\text{-}14)$$

where $\psi_{15} = 1.5 \times 10^4$ cm (15 bars) and θ_{15} is the moisture content when $\psi = \psi_{15}$. Again, the derivative of θ is discontinuous at saturation ($\psi = \psi_b$) but only requires three measured parameters: n, ψ_b, and θ_{15}. This model performed well for several soil types, including a clay.

McQueen and Miller [1974] proposed a linear relationship between pF = log $(-\psi)$, with ψ in cm, and moisture content. The three straight segments were a capillary segment from saturation to pF 2.5, an adsorbed segment from pF 2.5 to 5.0 and a tightly adsorbed segment from pF 5.0 to 7.0. As pointed out by McQueen and Miller [1974], their fitting procedure allows convenient approximation of the entire moisture characteristic curve from few data points.

Milly and Eagleson [1980] developed a related continuous function, considering only two unsaturated segments:

$$\begin{aligned} \theta(pF) = {} & \frac{1}{M} \ln \left[\exp\left\{M\,(a_1 - s_1\,pF)\right\} + \exp\left\{M\,(a_2 - s_2\,pF)\right\}\right] \\ & - \frac{1}{M'} \ln \left[\exp\left\{M'\,(a_2 - s_2\,pF)\right\} + \exp\left\{M'\theta_u\right\}\right] + \theta_u \end{aligned} \qquad (2\text{-}15)$$

in which a_1, a_2, s_1, s_2 and θ_u are defined by Figure 2-4, and M and M'

control the curvature of the joining segments. On a linear portion of the capillary range between pF_{min} and pF_0, this function can be approximated as:

$$\theta(\psi) = a_2 - s_2 \log(-\psi) \qquad (2\text{-}16)$$

and the specific moisture capacity is approximately

$$C(\psi) = -\frac{s_2}{\ln(10)} \frac{1}{\psi} \qquad (2\text{-}17)$$

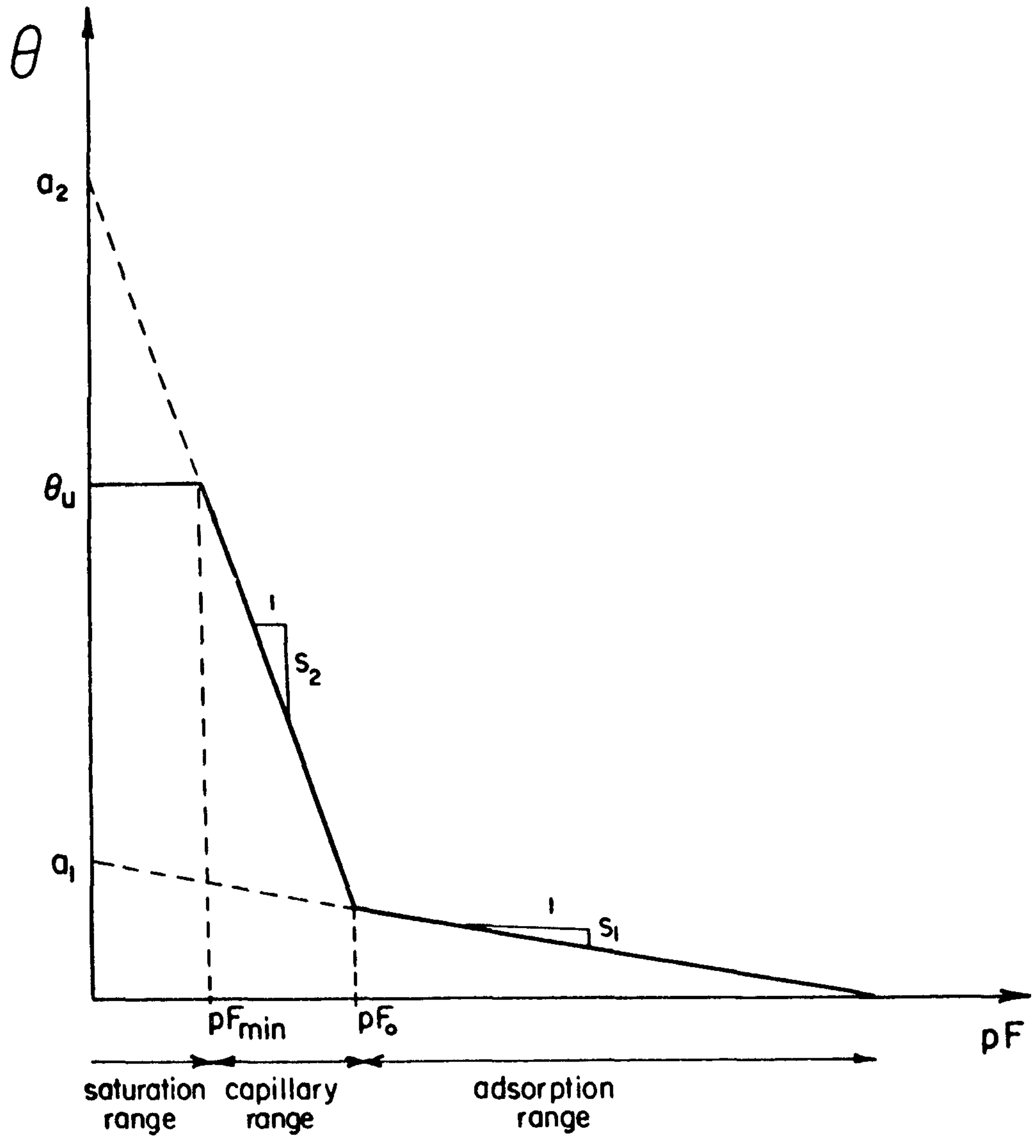

Figure 2-4. A piecewise linear relation between moisture content and the logarithm of suction (after Milly and Eagleson, 1980).

Since (2-15) has continuous slope, the exact form of (2-17), for all ψ, is also continuous and can be written:

$$C(\psi) = \frac{1}{M}\left[p_1\ (-\psi)^{q_1} + p_2\ (-\psi)^{q_2}\right]^{-1}\left[-p_1 q_1\ (-\psi)^{q_1 - 1} - p_2 q_2 (-\psi)^{q_2 - 1}\right] - \frac{1}{M'}\left[p_3\ (-\psi)^{q_3} + p_4\right]^{-1}\left[-\ p_3^{\ q_3}\ (-\psi)^{q_e - 1}\right] \tag{2-18}$$

where:

$$\begin{aligned} p_1 &= e^{M\ a_1} & q_1 &= -\ M\ s_1\ \log\ (e) \\ p_2 &= e^{M\ a_2} & q_2 &= -\ M\ s_2\ \log\ (e) \\ p_3 &= e^{M' a_2} & q_3 &= -M' s_2\ \log\ (e) \\ p_4 &= e^{M'\ u} \end{aligned} \tag{2-19}$$

Equations (2-15) and (2-18) are plotted for sandy and clayey soils in Figure 2-5.

2.4 Unsaturated Hydraulic Conductivity

Darcy's Law for unsaturated flow is the same for saturated flow with the provision that K is a function of suction (negative pressure). The equation states that the flux rate is proportional to the total piezometric head gradient times the hydraulic conductivity (Bear, 1979):

$$q = -K(\psi)\ \frac{\partial \phi}{\partial z} \tag{2-20}$$

For saturated flow, hydraulic conductivity is a function both of the soil type and of the fluid. In this analysis, hydraulic conductivity will refer to the value applicable when the fluid is the proposed impoundment leachate. Furthermore, it will be assumed that the properties (density and viscosity) of the leachate do not differ significantly from those of the native water.

For a given soil, the unsaturated hydraulic conductivity is dependent on the moisture content or matric potential (suction) in the soil. As a function of moisture content, θ, the hydraulic conductivity function shows little hysteresis [Bear, 1979; Maulem, 1976] and since, as stated previously, $\theta(\psi)$ hysteresis for this infiltration problem is ignored, the unsaturated hydraulic

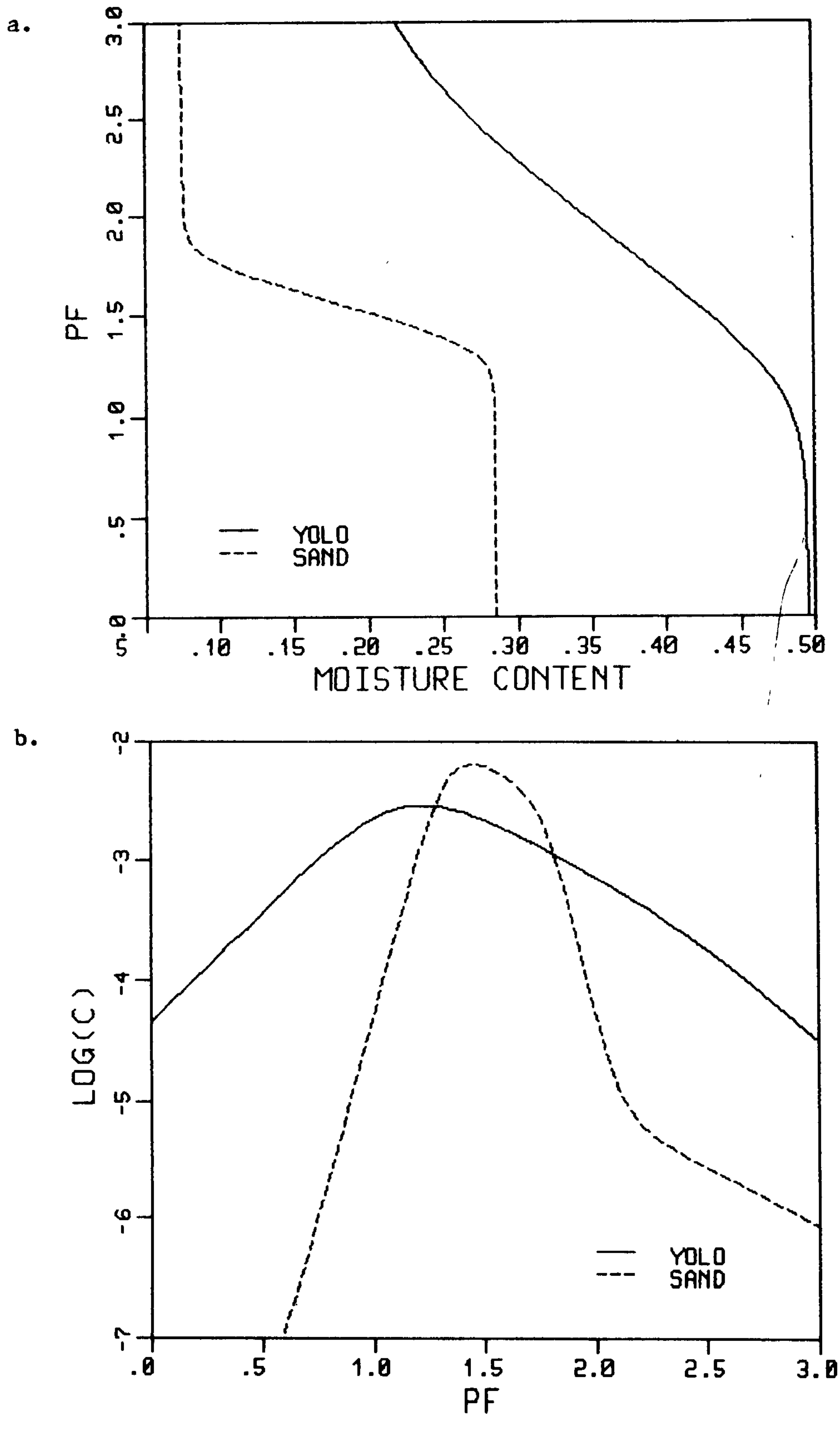

Figure 2-5. Sandy and clayey soil moisture characteristics computed from a continuous functional relationship developed by Milly and Eagelson (1980), a. moisture content and b. log of specific moisture capacity (cm^{-1}) vs. PF = log(-ψ), ψ in cm.

conductivity as a function of matric potential can also be determined uniquely. Figure 2-6 shows a schematic for typical relationships between hydraulic conductivity and matric potential for a sandy and a clayey soil.

As can be seen in Figure 2-6, when the pores are saturated ($\psi_b \leq \psi \leq 0$) the hydraulic conductivity is maximal. However, when the soils begin to drain ($\psi < \psi b$), some of the pores become air-filled and the conductive portion of the soil's cross-sectional area decreases correspondingly. In addition, as suction increases, the first pores to empty are the largest ones, which are the most conductive. Thus, the water is redirected to the smaller, less conductive pores.

It is well understood that under saturated conditions, a sandy soil conducts water more rapidly than a clayey soil. However, under unsaturated conditions, the opposite may occur. Since a sandy soil contains relatively larger pores, these pores quickly empty and become non-conductive as suction increases, thus, steeply decreasing the initially high conductivity value associated with sandy soils. On the other hand, in a soil with smaller pores such as clays, many of the pores remain full and conductive even at appreciable suction. Thus, the conductivity does not decrease as fast in a soil with small pores, and may actually become greater than that of a soil with large pores subject to the same suction. In the field where unsaturated conditions exist, it often happens that water flow is greater and persists longer in clayey soils than in sandy soils.

Numerous functional relationships have been proposed for the unsaturated hydraulic conductivity. These models include

$$\text{Gardner [1958]:} \quad K(\psi) = a \;/\; (b + (-\psi)^m) \qquad (2\text{-}21)$$

where a, b, and m are constants;

$$\text{Gardner [1958]:} \quad K(\psi) = K_s \exp\,(-a\psi) \qquad (2\text{-}22)$$

where K_s is the saturated hydraulic conductivity;

$$\text{Brooks and Corey [1964]:} \quad K(\psi) = K_s \left|\frac{\psi}{\psi_b}\right|^{-\lambda} \qquad (2\text{-}23)$$

for $\psi \leq \psi_b$, where ψ_b is the air entry pressure and λ is an index of pore-size distribution, and

$$\text{Maulem [1978]:} \quad K(\psi) = K_s\,(S_e)^{0.015\,w\,+\,3.0} \qquad (2\text{-}24)$$

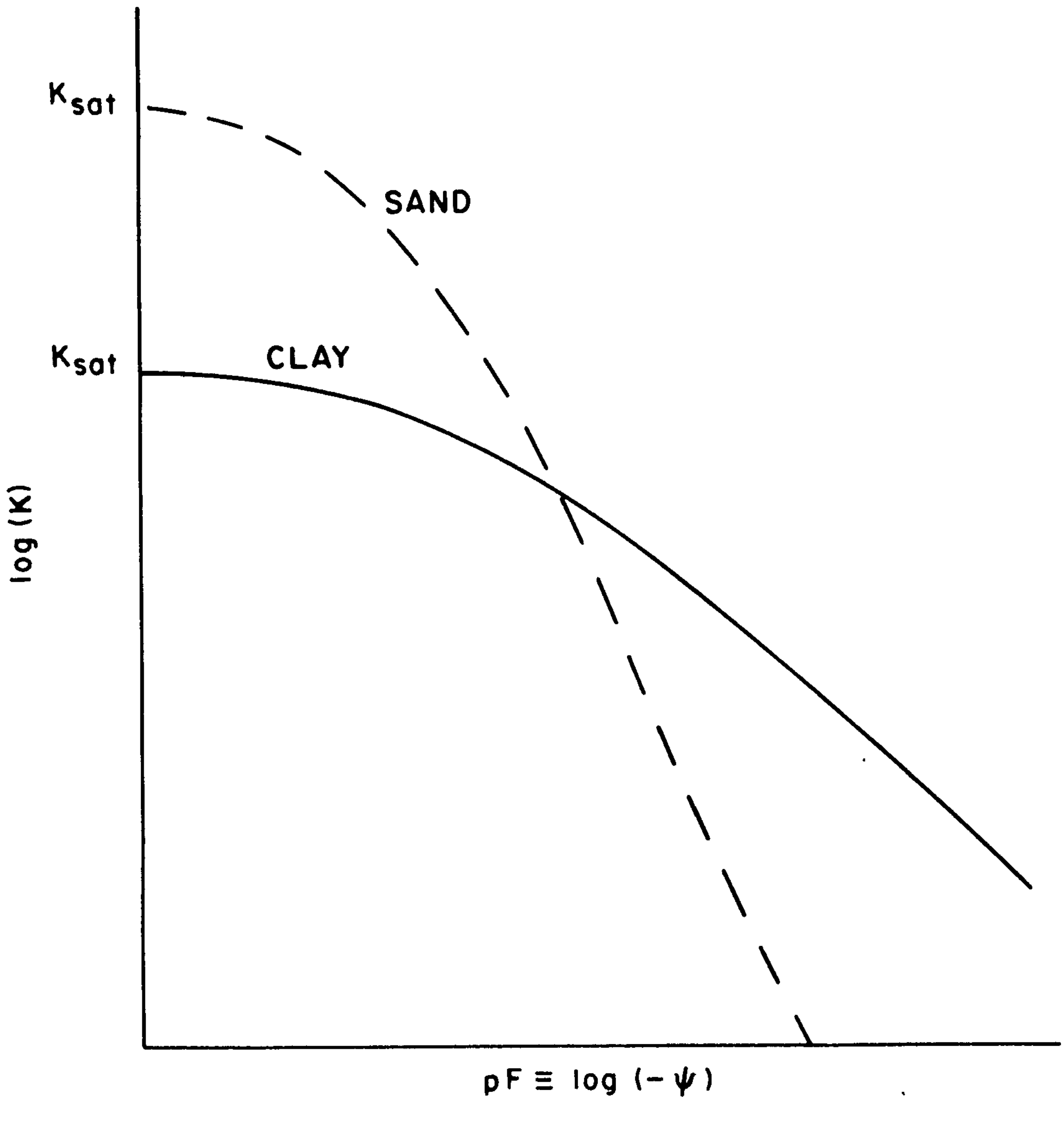

Figure 2-6. Schematic of unsaturated hydraulic conductivity for a sand and clay soil.

where S_e is effective saturation:

$$S_e = \frac{\theta - \theta_r}{n - \theta_r} \tag{2-25}$$

and

$$w = \int_{\psi=+\infty}^{\psi=0} \gamma_w \psi \, d\theta \tag{2-26}$$

in which γ_w is the specific weight of water [F/L^3]. These formulas (2-24 - 2-26) require use of cgs units. Equation (2-26) "represents the amount of work required to drain a unit volume of a saturated soil" [Maulem, 1978]. Thus, equation (2-24) is dependent on the moisture characteristic curve. This model was shown to improve the prediction of unsaturated hydraulic conductivity for fine-grained soil [Maulem, 1978] relative to previous techniques.

There are several other integral techniques which derive the hydraulic conductivity function from the moisture characteristic curve [see Maulem 1976; Elzeftawy and Cartwright 1981; Jackson et al., 1965]. Such techniques are especially attractive when used in conjunction with numerical models because of the numerical integration required. The accuracy of these methods, of course, depends on the accuracy of the soil moisture characteristic curve, and on the applicability of the physical assumptions made in a particular technique.

2.5 Characteristic Curves Available in SOILINER

During the numerical solution of the governing equation for vertical unsaturated flow, the soil moisture content, θ, and the hydraulic conductivity, K, must be determined. The characteristic curves used by SOILINER to relate soil moisture content with matric potential, ψ, are modeled using a power curve function combined with a parabolic function near saturation to represent gradual air entry. This method is described in detail by R.B. Clapp and G.M. Hornberger (1978). A conductivity function presented by Campbell (1974) is used in SOILINER to relate soil moisture content with unsaturated hydraulic conductivity, K. This function is estimated from the soil moisture power curve stated above and the soil's saturated hydraulic conductivity.

The power curve function representing the moisture characteristic is

$$\psi = \psi_s W^{-b} \tag{2-27}$$

with soil wetness, W, equal to θ/θ_s, where θ_s is the saturated water content, or porosity. Both the matric potential at saturation, ψ_s, and the exponent b, which is the slope of the line segment for $W \le W_i$, are empirical and must be estimated using a power curve regression analysis performed on the log transforms of the parameters ψ and W. It has been shown that b is a function of soil texture, and increases as soil texture moves from coarse to fine grained. The use of this function implies a sharp discontinuity in matric potential near saturation. This is illustrated in Figure 2-7 as the dotted line segment for $W > W_i$. Although it is thought that some coarse grained sands may have a small matric potential at $W = 1$, most soils, particularly medium to finer grained soils show a gradual air entry region near saturation. A modification to equation 2-27 is made to account for this gradual air entry. Along a typical characteristic curve, there exists a point where $d\psi/dW$ changes from an increasing function to a decreasing function as W increases. This inflection point has the coordinates (W_i, ψ_i) as shown in Figure 2-7, and the interval $W_i \leq W \leq 1$ is described by the parabola:

$$\psi = -m(W-n)(W-1) \qquad (2\text{-}28)$$

The parameters m and n are calculated such that:

1. equation 2-28 passes through points (W_i, ψ_i) and (1,0) and,

2. $d\psi/dW$ of equations 2-27 and 2-28 are equal at the inflection point (W_i, ψ_i).

The expression for the parameters m and n are

$$m = \frac{\psi_i}{(1-W_i)^2} - \frac{\psi_i b}{(W_i\ (1-W_i)} \qquad (2\text{-}29)$$

$$n = 2W_i - \frac{\psi_i b}{mW_i} - 1$$

Campbell (1974) derived a simple formula for estimating the unsaturated conductivity given the matric potential at a specific time. This equation is

$$k = W^{2b+3} \qquad (2\text{-}30)$$

where $k = K/K_s$. This formula has proven to be reasonably accurate over a wide range of b values (wide range of soil textures) and for W values near saturation.

In order to utilize equations 2-27, 2-28, and 2-30 for determining θ and K_s, for a given soil and matric potential, values for b, ψ_s, θ_s, K_s, m and n must be known. Table 2-1 provides representative average values of these parameters for 11 soil textures. These values were obtained by applying the power curve to desorption data from 1446 soil types. The soil samples were collected from 34 localities throughout the United States. Each soil type was assigned to a textural class based on its clay content. Values for m and n corresponding to each textural class were estimated by setting W_i to 0.92. According to Clapp and Hornberger (1978), there was nothing in the soils data indicating where this inflection point may occur. Rogowski (1971) stated that this point usually occurs in the interval $0.8 < W < 1$ and that 0.9 is a useful estimate. The value 0.92 was chosen so that m would always be

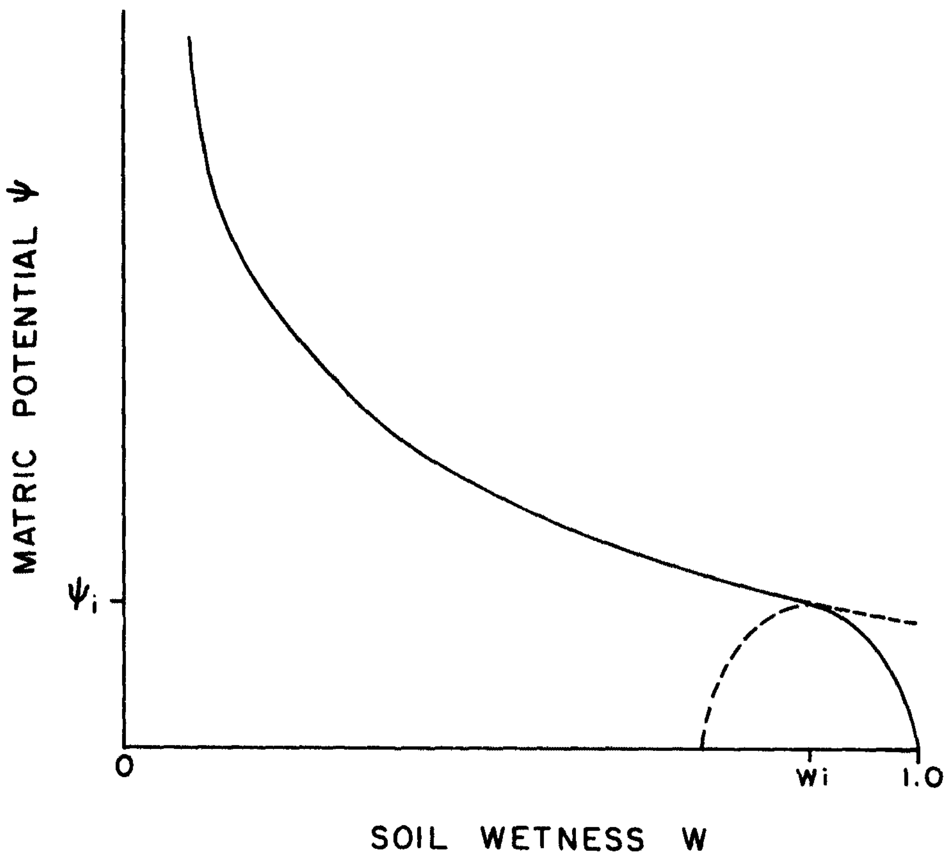

Figure 2-7. The moisture characteristic curve using equations 2-27 and 2-28 for the hyberbolic and parabolic sections, respectively (the broken line segments are disregarded).

positive. For m to be positive, W_i must be greater than b/(b+1). Figures 2-8a and 2-8b show the results of the two part function using the parameters in Table 2-1 and the appropriate values for m and n.

The use of these equations in SOILINER makes it possible to determine, for each value of matric potential, the hydraulic conductivity and moisture content at each node. The advantages of selecting this method as opposed to other methods described in Sections 2.3 and 2.4 are as follows:

1. With this approach only four parameters are needed (b, ψ_s, θ_s, K_s) to give a description of the hydraulic properties of a soil. In addition, this method accounts for gradual air entry near saturation.

2. The parameter, b, can be estimated from matric potential/moisture content data or taken from Table 2-1 and used in equation (2-30), so that unsaturated hydraulic conductivity need not be measured directly.

3. The equations are straightforward and can be used to derive other functional relationships if accurate soil moisture data is available.

4. Since this method categorizes soils into textural groups and uses an average characteristic curve for each group, the user can simulate soils in which no moisture suction data is available. This is done by simply choosing the appropriate textural class for that soil.

The disadvantages of using this method to relate ψ, θ, and K in the SOILINER model include the following:

1. The values presented in Table 2-1 were derived from soil moisture data obtained during the desorption process. Since the model usually simulates a wetting front moving through a liner, inaccuracies (due to the effects of hysteresis) may occur.

2. Errors may occur in using average values for each textural class as opposed to values pertaining specifically to an individual soil and its associated characteristic curve.

The equations have been useful to Clapp and Hornberger in several different soil moisture models, one of which estimated the wetting front suction required by the Green-Ampt equation. However, the average values presented in Table 2-1 have not been verified and should be used with this limitation in mind. In addition, these average parameters should be used only in the absence of soil moisture data specific to a site. If soil moisture data is available or can be obtained, it is recommended that the above described functions be fitted to these data to obtain the most accurate results. Sections 3.2 and 3.3 of the SOILINER User's Guide provides methods for obtaining soil moisture data and describes how these functions are applied to soil moisture data.

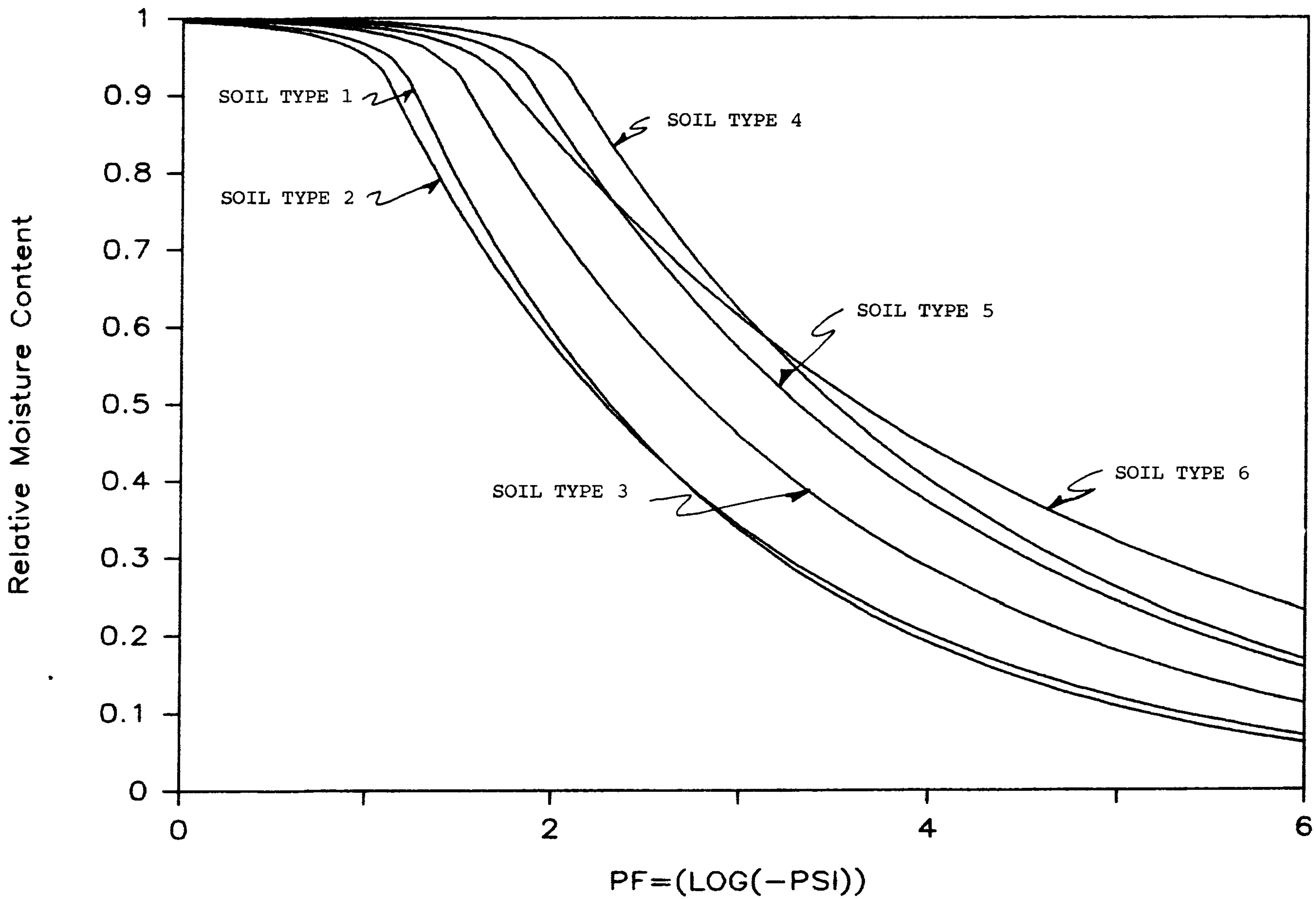

Figure 2-8a. Characteristic moisture curves for soil types ranging from sand to sandy clay loam.

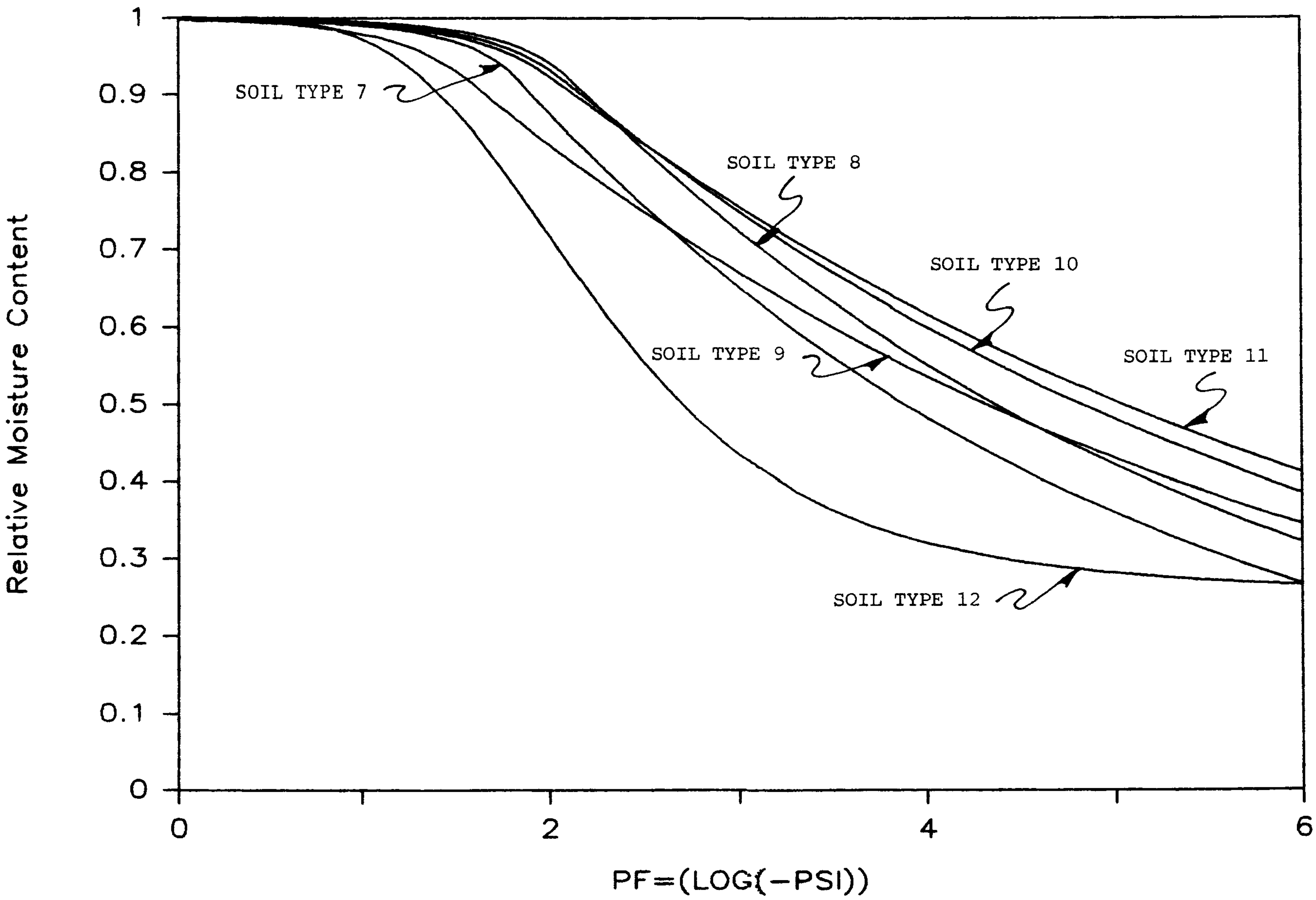

Figure 2-8b. Characteristic moisture curves for soil types ranging from silty clay loam to a compacted clay.

TABLE 2-1. REPRESENTATIVE VALUES OF HYDRAULIC PARAMETERS FOR A NUMBER OF SOIL TEXTURE CLASSES

Soil texture	Mean clay fraction	$\bar{b}$	$\bar{\psi}_s$ (cm)	$\bar{\theta}_s$ (cm^3/cm^3)	K_s* (cm/min)
Sand	0.03	4.05	12.1	0.395	1.056
Loamy sand	0.06	4.38	9.0	0.410	0.938
Sandy Loam	0.09	4.90	21.8	0.435	0.208
Silt loam	0.14	5.30	78.6	0.485	0.0432
Loam	0.19	5.39	47.8	0.451	0.0417
Sandy clay loam	0.28	7.12	29.9	0.420	0.0378
Silty clay loam	0.34	7.75	35.6	0.477	0.0102
Clay loam	0.34	8.52	63.0	0.476	0.0147
Sandy clay	0.43	10.4	15.3	0.426	0.0130
Silty clay	0.49	10.4	49.0	0.492	0.0062
Clay	0.63	11.4	40.5	0.482	0.0077

* From Li et al. (1976)

3. NUMERICAL SIMULATION OF UNSATURATED FLOW

Due to the nonlinearities of the unsaturated flow equation, exact analytical solutions have not been obtained except for a few simple cases. Numerical techniques are available to solve the unsaturated flow equations. These techniques, primarily finite difference and finite element methods, provide solutions of the governing equations that take into account the inherent nonlinearities of the system and the heterogeneity of soil types and variable initial conditions. They all rely on the basic concept of solving for the moisture state at a finite number of points in space and time.

SOILINER utilizes the Finite Difference Method (FDM) to solve the nonlinear, unsaturated flow equation. The FDM has been applied to vertical unsaturated flow by many authors. Freeze [1969] investigated natural ground water recharge and discharge mechanisms. Brutsaert [1971] used a two-dimensional vertical model in a study of soil moisture flow beneath drains and irrigation ditches. Cooley [1971] investigated flow to a pumping well using an axisymmetric two-dimensional vertical model. For one-dimensional vertical flow, the FDM has been verified by, among others, Green et al. [1970], Ragab et al. [1982], Giesel et al. [1973], and Elzeftawy and Dempsey [1976]. Kunze and Nielson [1982], Haverkamp et al. [1977], and Reeder et al. [1980] show excellent agreement between finite difference solutions and Philip's [1958, 1969] quasi-analytical results. Unlike previous computer models mentioned above, SOILINER is specifically designed to evaluate soil liner performance. Although SOILINER can be used in other applications it is best utilized to simulate infiltrating leachate through soil profiles beneath impounded liquid.

3.1 Finite Difference Method

The FDM consists of first breaking the solution domain of a differential equation into subdomains. This process is called discretization. For each subdomain, continuous differential terms in the governing equation are replaced by approximate expressions based on the values of the state variable in the subdomains. Typically, one or more equations are solved for each subdomain. This technique is relatively simple to understand in practice, and is applied to both spatial and temporal differential terms. Freeze and Cherry [1979] describe the application of finite difference techniques to vertical unsaturated flow.

In application to the vertical unsaturated flow equation, the vertical column is divided into a row of short vertical segments. This row of segments is called a grid. SOILINER utilizes mesh-centered grids, where the values of matric potential (ψ) are evaluated at nodes, which are located at the ends of each segment. The vertical segments, or elements, may exhibit either variable (Figure 3-1a) or constant length (see Figure 3-1b). Likewise, solutions in time are obtained by breaking time into discrete steps. Solutions at new times are obtained (at the nodes) using the previous solution(s). In general, the accuracy of this method improves as the grid spacing (Δz) and the time step (Δt) decrease in size.

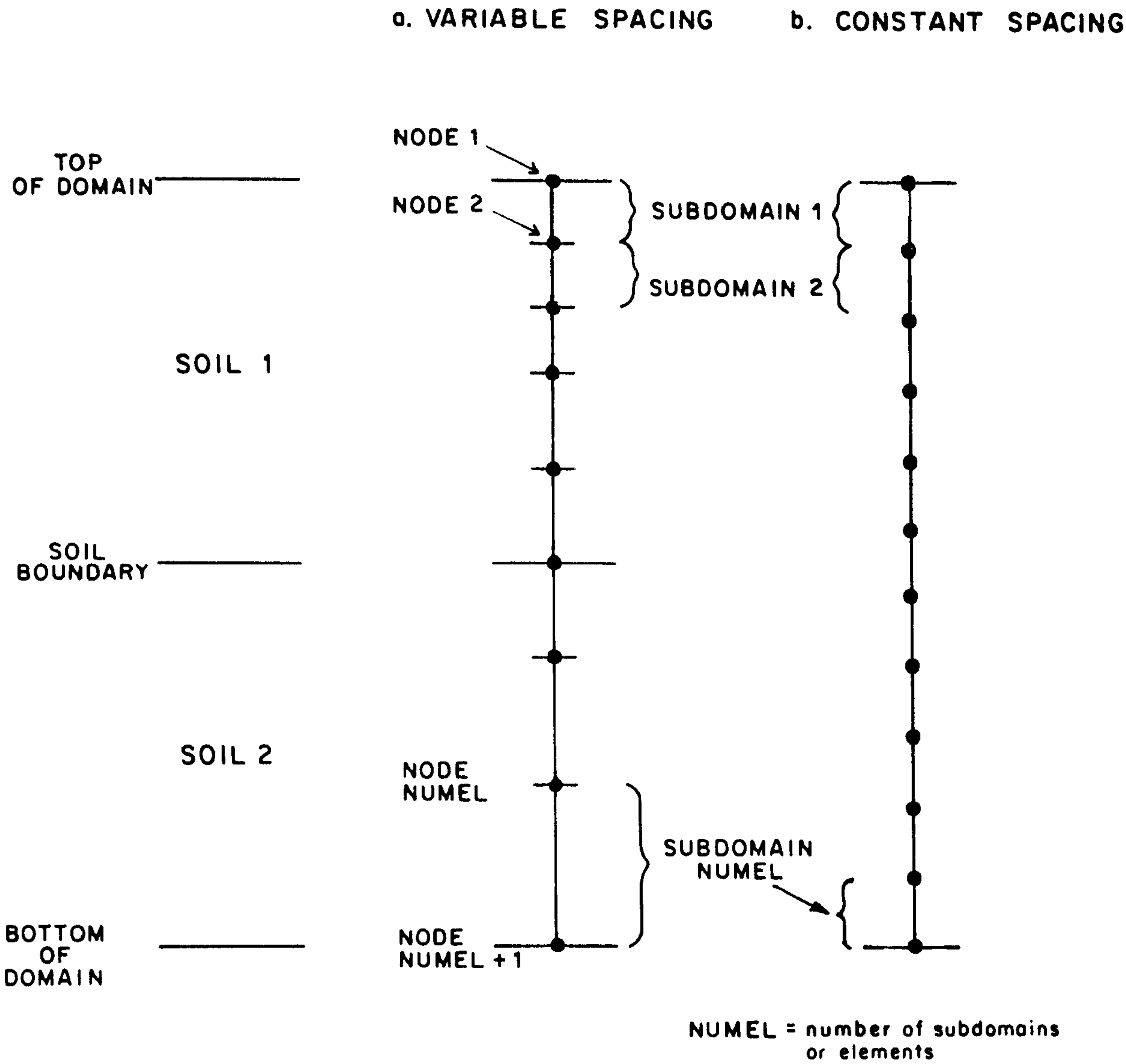

Figure 3-1. Finite difference method spacial discretization using mesh-centered grids.

3.1.1 FDM Spatial Difference Approximations--

The spatial domain of the vertical unsaturated flow equation (2-1a) is the soil column, from the top of the impoundment liner down to the water table. This domain includes the liner and the underlying site soil, and can be discretized as shown in Figure 3-1 where NUMEL + 1 represents the bottom node number. Values of the state variable (matric potential, ψ), and soil properties (hydraulic conductivity, $K(\psi)$, and moisture capacity, $C(\psi)$) are approximated by values at a node i: ψ_i, K_i, and C_i. Note that K_i and C_i are functions of ψ_i. These values are used to approximate the derivatives in (2-1a). The first term in (2-1a) representing pressure driven flux, can be replaced by:

$$\frac{\partial}{\partial z}\left[K(\psi)\frac{\partial \psi}{\partial z}\right] = \left[K_{i+1/2}\frac{\psi_{i+1} - \psi_i}{\Delta z} - K_{i-1/2}\frac{\psi_i - \psi_{i-1}}{\Delta z}\right]\frac{1}{\Delta z} \qquad (3\text{-}1)$$

in which $z = z_{i+1} - z_i$ is the constant node spacing and

$$K_{i+1/2} = (K_i \times K_{i+1})^{1/2}$$
$$K_{i-1/2} = (K_i \times K_{i-1})^{1/2} \qquad (3\text{-}2)$$

are the geometric mean hydraulic conductivities between nodes. The gravitational flux term can be expressed as:

$$\frac{\partial K(\psi)}{\partial z} = \frac{K_{i+1} - K_{i-1}}{2\Delta z} \qquad (3\text{-}3)$$

Equations (3-1) and (3-3) are centered difference approximations [see Pinder and Gray, 1977]. These expressions assume that the node spacing, Δz, is constant, although that is not a general requirement (Section 4 presents a variable spacing, centered-difference algorithm analogous to 3-1). With constant node spacing (3-3) is second order accurate. This means that if conductivity $K(\psi)$ is a linear function of z and z^2 only, then (3-3) is exact.

3.1.2 FDM Temporal Derivative Approximations--

The temporal domain of the vertical unsaturated flow equation (2-1a) is time, t > 0, after infiltration into the liner begins. Time is broken down into time level subdomains in which superscript n represents time level. The nodal values of properties and state variables, which are actually continuous in time, are approximated by values at discrete time levels: $\psi^n{}_i$, $K^n{}_i$, and $C^n{}_i$. To maintain computational efficiency, time level subdomains (Δt) are automatically varied within SOILINER. First, an initial Δt and time step change parameter are set, where Δt is multiplied by the change parameter. Alone, the product of these two parameters would result in an ever-increasing Δt. However, for each successive time step, Δt is also modified by the maximum change in matric potential (ψ) at any given node in the flow domain between the previous and current time levels.

Another feature incorporated into the SOILINER temporal derivative approximation is that the storage term in (2-1a) can be written:

$$C \frac{\partial \psi}{\partial t}_{n+\alpha} = \alpha \left[C \frac{\partial \psi}{\partial t} \right]_{n+1} + (1-\alpha) \left[C \frac{\partial \psi}{\partial t} \right]_{n} = \frac{\psi^{n+1} - \psi^{n}}{\Delta t} C^{n+\alpha} \qquad (3\text{-}4)$$

in which $C^{n+\alpha} = \alpha C^{n+1} + (1-\alpha) C^{n}$; $0 \leq \alpha \leq 1$ is a temporal weighting parameter, and $t = t^{n+1} - t^{n}$ is the time step size. Subscripts have been dropped for convenience. If $\alpha = 0$, (3-4) becomes

$$\left[C \frac{\partial \psi}{\partial t} \right]_{n} = \frac{\psi^{n+1} - \psi^{n}}{\Delta t} C^{n} \qquad (3\text{-}5)$$

which is forward differencing. When solving for ψ^{n+1}, all other terms of (3-5) are known from the last time step and thus ψ^{n+1} is solved <u>explicitly</u>. If $\alpha=1$, (3-4) becomes:

$$\left[C \frac{\partial \psi}{\partial t} \right]_{n+1} = \frac{\psi^{n+1} - \psi^{n}}{\Delta t} C^{n+1} \qquad (3\text{-}6)$$

Both sides of (3-6) contain terms which must be evaluated at the current time step, t^{n+1}, and depend on the solution ψ^{n+1}, thus (3-6) is <u>implicit</u> in ψ^{n+1}. Another standard time differencing scheme is the Crank-Nicolson procedure obtained by setting $\alpha=0.5$ in (3-4). Figure 3-2 provides a schematic of the implicit and explicit temporal discretization strategies.

The fully explicit scheme ($\alpha=0$) is conditionally stable (the solution will not always converge) and should only be used with small time steps. The <u>fully implicit</u> scheme ($\alpha=1$) is unconditionally stable although accuracy is greatly affected by time step size. The Crank-Nicolson scheme ($\alpha=0.5$) is also unconditionally stable and is more accurate for many flow problems [Pinder and Gray, 1977]. Theoretically, any $0 \leq \alpha \leq 1$ is first order accurate except $\alpha=0.5$, which is second order accurate for constant time steps.

3.2 Numerical Solution

SOILINER combines the centered difference approximation for the spatial derivative and the temporal weighting scheme producing a system of algebraic equations to be solved simultaneously. Both the transient and steady state solution strategies ultimately employ the same matrix equation solver. The following is a detailed development of the transient, finite difference equation which also forms the basis of the steady state approximation. Model stability, with respect to the solution strategy chosen (transient vs. steady state), will then be discussed.

3.2.1 Transient, Finite Difference Equation--

The governing equation for vertical unsaturated flow (see Section 2) can be written:

$$C(\psi) \frac{\partial \psi}{\partial t} - \frac{\partial}{\partial z} \left[K(\psi) \frac{\partial \psi}{\partial z} + K(\psi) \right] = 0 \qquad (3\text{-}7)$$

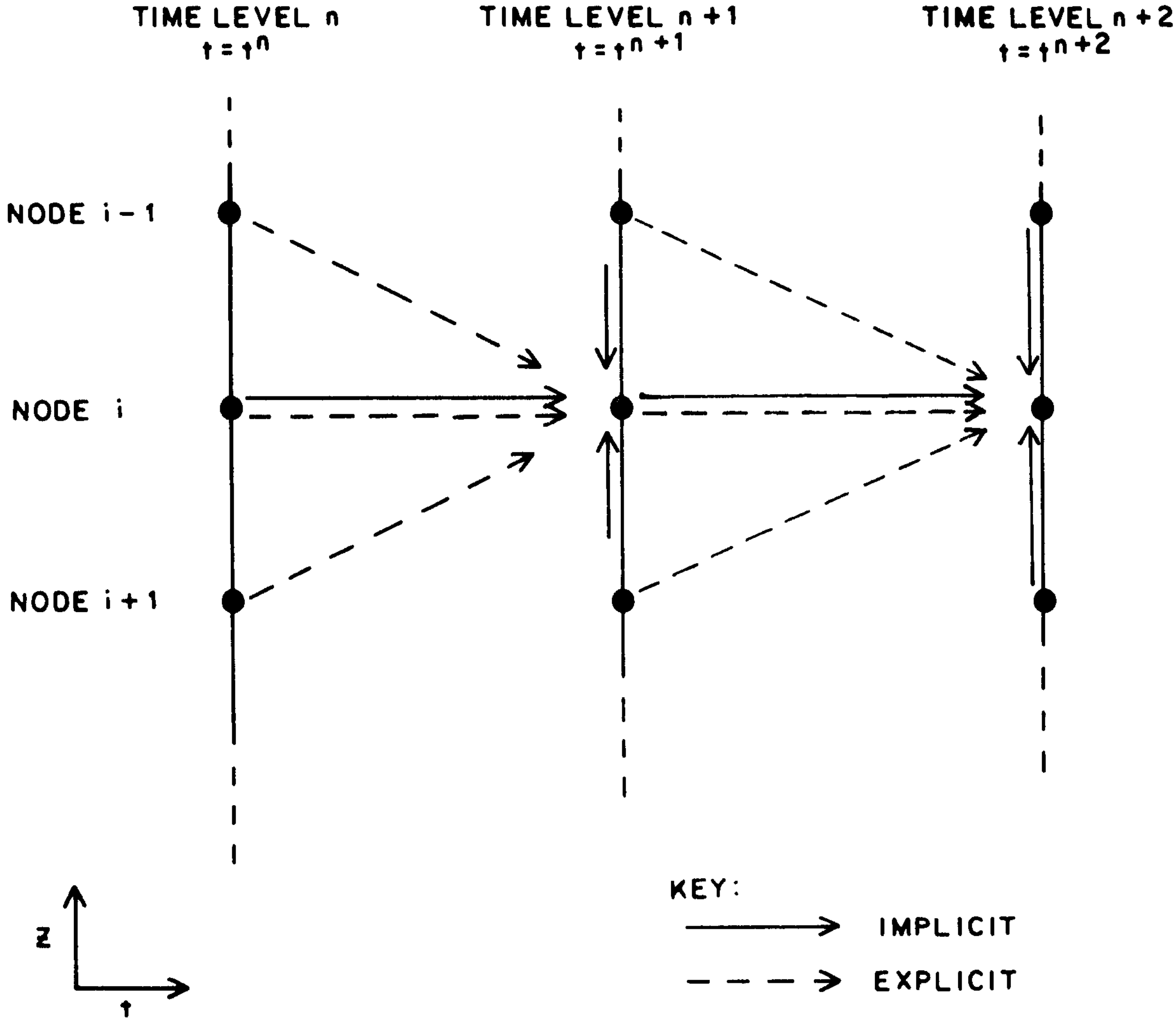

Figure 3-2. Finite difference method temporal discretization for node i at time level n + 1 showing explicit (dash) and implicit (solid) relationships.

in which $C(\psi) = \partial\theta/\partial\psi$ $[L^{-1}]$ is specific moisture capacity, where θ [-] is volumetric moisture content; ψ [L] is matric potential or capillary pressure head; $K(\psi)$ $[LT^{-1}]$ is vertical unsaturated hydraulic conductivity; z [L] is the vertical coordinate, negative downward; and t [T] is time. For notational purposes, the second term in (3-7) which is flux divergence, is denoted by:

$$V(\psi) = \frac{\partial}{\partial z}\left[K(\psi)\frac{\partial\psi}{\partial z}\right] + \frac{\partial K(\psi)}{\partial z} \tag{3-8}$$

and by (3-7)

$$V(\psi) = C(\psi)\frac{\partial\psi}{\partial t} \tag{3-9}$$

thus V $[T^{-1}]$ can be thought of as the rate of storage change at any point.

Standard centered finite difference expressions can be used to evaluate V in (3-8). The spatial domain is discretized into a mesh centered grid (see Figure 3-1) with node points on the boundaries between soil elements. Substituting the FDM representations for the two flux terms (see (3-1) and (3-3)) into (3-8), we have:

$$V_i(\psi) = \left[K_{i+1/2}\frac{\psi_{i+1} - \psi_i}{\Delta z_{i+1/2}} - K_{i-1/2}\frac{\psi_i - \psi_{i-1}}{\Delta z_{i-1/2}}\right]\frac{1}{\Delta z_i} + \frac{K_{i+1/2} - K_{i-1/2}}{\Delta z_i} \tag{3-10}$$

where subscript "i" designates node number and

$$\begin{aligned} \Delta z_{i+1/2} &= z_{i+1} - z_i \\ \Delta z_{i-1/2} &= z_i - z_{i-1} \\ \Delta z_i &= 1/2\,(z_{i+1} - z_{i-1}) \end{aligned} \tag{3-11}$$

are the lengths of the two elements on each side of node i and the length associated with node i, respectively. The element conductivities are evaluated as geometric averages:

$$\begin{aligned} K_{i+1/2} &= [K(\psi_{i+1}) \times K(\psi_i)]^{1/2} \\ K_{i-1/2} &= [K(\psi_i) \times K(\psi_{i-1})]^{1/2} \end{aligned} \tag{3-12}$$

A weighted temporal difference expression is obtained by substituting (3-9) into (3-4):

$$C^{(n+\alpha)}\left(\frac{\psi^{n+1} - \psi^n}{\Delta t}\right) = \alpha\, V(\psi^{n+1}) + (1-\alpha)\, V(\psi^n) \tag{3-13}$$

in which $0 \leq \alpha \leq 1$ is the weighting parameter; and

$$C^{n+\alpha} = \alpha C^{n+1} + (1-\alpha)\, C^{n} \tag{3-14}$$

These expressions apply at each node and subscripts have been dropped for convenience.

Equation (3-13) is a FDM expression of the governing equation (3-7) at a node. Since both C^{n+1} and K^{n+1} depend on the solution, ψ^{n+1}, equation (3-13) is nonlinear in ψ. Thus, after the initial prediction of ψ^{n+1} (equation 3-13), an iterative procedure is used to resolve ψ^{n+1}. The solution at a new iteration can be expressed as the solution from the last iteration plus a correction computed at the new iteration

$$\psi^{k+1} = \psi^{k} + \Delta\psi^{k+1} \tag{3-15}$$

in which all terms are at node i and time step n+1 and superscript k is iteration level. The left hand side (LHS) of (3-13) becomes:

$$C^{n+\alpha}\left(\frac{\psi^{n+1} - \psi^{n}}{\Delta t}\right) = C^{k,n+\alpha}\left(\frac{\psi^{n+1} - \psi^{n}}{\Delta t}\right) + C^{k,n+\alpha}\left(\frac{\Delta\psi^{k+1}}{\Delta t}\right) \tag{3-16}$$

The second term on the right hand side (RHS) of (3-13) is known from the last time step. The first RHS term is evaluated by substituting (3-15) into (3-10):

$$V_i\left(\psi^{k+1}\right) = \left[K^{k}_{i+1/2}\left(\frac{\psi^{k}_{i+1} + \Delta\psi^{k+1}_{i+1} - \psi^{k}_{i} - \Delta\psi^{k+1}_{i}}{\Delta z_{i+1/2}}\right) - K^{k}_{i-1/2}\left(\frac{\psi^{k}_{i} + \Delta\psi^{k+1}_{i} - \psi^{k}_{i-1} - \Delta\psi^{k+1}_{i-1}}{\Delta z_{i-1/2}}\right)\right]\frac{1}{\Delta z_i} + \frac{K^{k}_{i+1/2} - K^{k}_{i-1/2}}{\Delta z_i} \tag{3-17}$$

or

$$V_i\left(\psi^{k+1}\right) = \left[K^{k}_{i+1/2}\left(\frac{\Delta\psi^{k+1}_{i+1} - \Delta\psi^{k+1}_{i}}{\Delta z_{i+1/2}}\right) - K^{k}_{i-1/2}\left(\frac{\Delta\psi^{k+1}_{i} - \Delta\psi^{k+1}_{i-1}}{\Delta z_{i-1/2}}\right)\right]\frac{1}{\Delta z_i} + V_i\left(\psi^{k}\right) \tag{3-18}$$

in which $V(\psi^{k})$ is evaluated from the last iteration, and, again, superscript "k" or "k+1" represents iterative values at time step n+1.

The final form of the FDM governing equation is obtained by substituting (3-16) and (3-18) into (3-13) and grouping ψ^{k+1} terms on the LHS:

$$C_i^{k,n+\alpha}\left(\frac{\Delta\psi_i^{k+1}}{\Delta t}\right) - \left[K_{i+1/2}^k\left(\frac{\Delta\psi_{i+1}^{k+1} - \Delta\psi_i^{k+1}}{\Delta z_{i+1/2}}\right) - K_{i-1/2}^k\left(\frac{\Delta\psi_i^{k+1} - \Delta\psi_{i-1}^{k+1}}{\Delta z_{i-1/2}}\right)\right]\frac{1}{\Delta z_i} = -C_i^{k,n+\alpha}\left(\frac{\psi_i^k - \psi_i^n}{\Delta t}\right) + \alpha V_i(\psi^k) + (1-\alpha)\, V_i(\psi^n) \qquad (3\text{-}19)$$

All terms on the RHS of (3-19) are known from the last time step, "n", or the last iteration, "k" at the new time step "n+1".

Equation (3-19) is repeated for each node of unknown matric potential (i.e. excluding boundary nodes of fixed ψ). Thus, application of FDM to the governing flow equation results in a system of equations to be solved each time step for the unknown nodal values of matric potential. The resulting matrix equation is solved directly for $\Delta\psi^{k+1}$ using the Thomas algorithm for tridiagonal matrices (see Pinder and Gray, 1977). Equation (3-19) is solved iteratively for $\Delta\psi$ at each time step, updating ψ as shown in (3-15), ultimately resolving the equations until the soil properties for that time step approach constant values, and $\Delta\psi \to 0$. Convergence at a given time step is determined by a set criterion for the maximum $\Delta\psi$ at any node between successive iterations.

SOILINER will continue to iterate until the set criterion is achieved or the maximum number of iterations specified per time step is exceeded, and a forced exit from the iterative procedure occurs. In some cases convergence can be achieved by simply increasing the number of maximum iterations, however forced exits generally indicate divergence (i.e., $\Delta\psi \to \infty$). Conversely, convergence may not occur if the set criterion is too stringent. For example, the largest calculated $\Delta\psi$ at any node between successive iterations may oscillate around a value of 0.01 cm but not converge if the error criterion were set at 0.001 cm. Thus, convergence is a relative term associated with the desired level of accuracy and the effort required to optimize a combination of parameters for the solution strategy. Part II of this document provides guidelines for the development of an appropriate set of parameters for both transient and steady state solutions.

Figure 3-3 provides a flow chart for the overall procedure of a transient solution strategy. When convergence occurs at a given time step, flux and velocity calculations are made (Section 3.3), a particle is advected through the liner as a means of determining breakthrough (Section 3.4), and a new time step is determined. The procedure continues until steady state is achieved, at which time the particle is tracked until the point of breakthrough.

3.2.2 Steady State, Finite Difference Equation--

A finite difference approximation of (2-1b) for steady state conditions is developed identically to (3-19). However, the storage term $C(\psi)$ and α are set equal to 0.0 and 1.0 respectively in (3-19), resulting in:

$$-\frac{1}{\Delta z_i}\left[K_{i+1/2}^k\left(\frac{\Delta\psi_{i+1}^{k+1} - \Delta\psi_i^{k+1}}{\Delta z_{i+1/2}}\right) - K_{i-1/2}^k\left(\frac{\Delta\psi_i^{k+1} - \Delta\psi_{i-1/2}^{k+1}}{\Delta z_{i-1/2}}\right)\right] = V_i(\psi^k) \qquad (3\text{-}20)$$

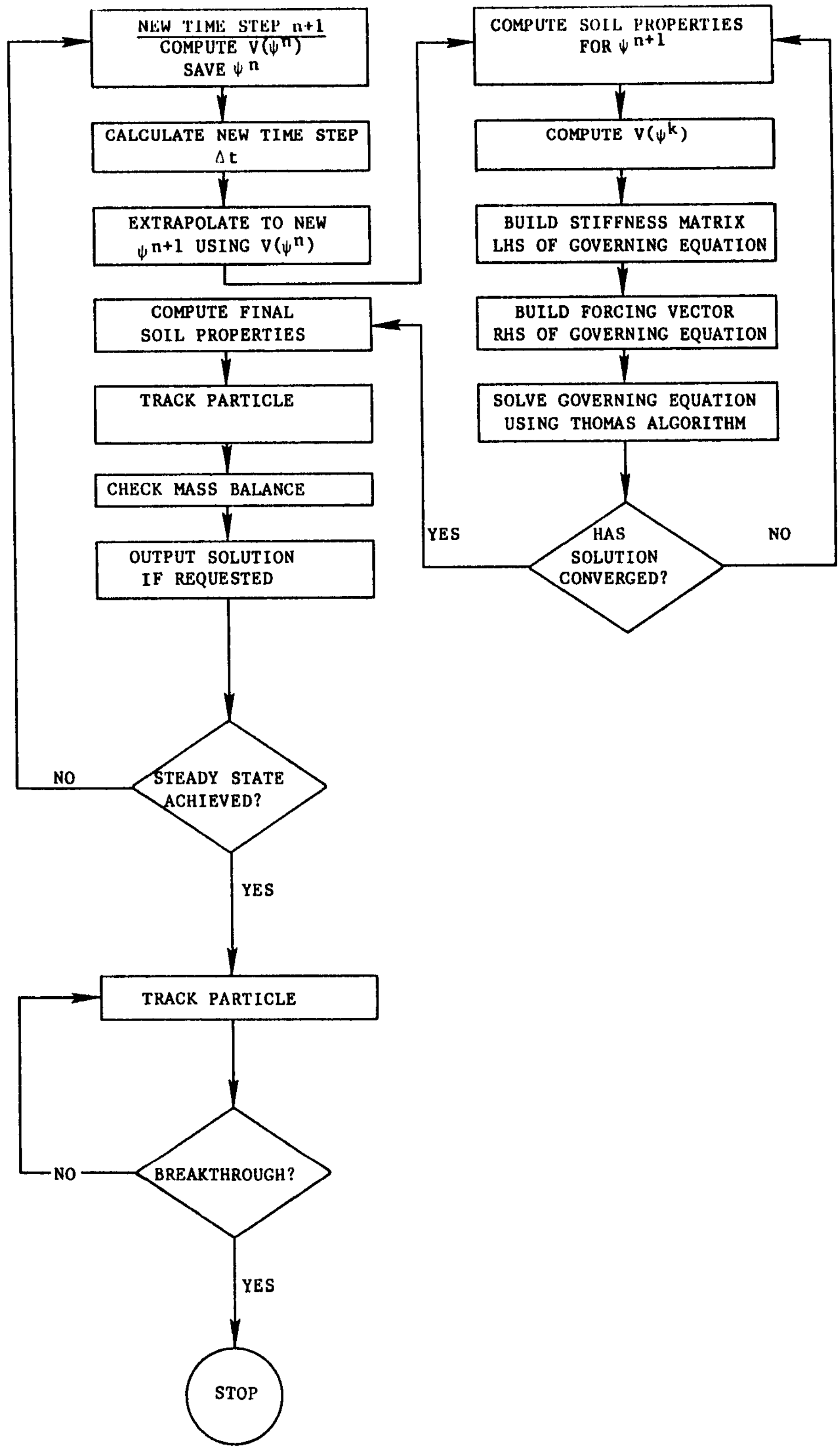

Figure 3-3. SOILINER solution procedure flow chart.

Since the steady state solution is achieved iteratively, initial conditions must be specified as represented by $V_i(\psi^k)$ in (3-20) for the first iteration k = 1. The system of equations to be solved at each iteration for the nodal values of matric potential (ψ), also employs the Thomas algorithm. Convergence is achieved utilizing a similar algorithm as that described for the transient solution. However, due to the highly nonlinear properties of the governing equation, the solution algorithm is more susceptible to divergence for the steady state solution than the transient formulation when modeling layered flow systems. Instability is due in part to the extreme change in soil properties during the iterative procedure between initial conditions and the steady state solution. Oscillations occur in association with numerical overshoot of the predicted values for $\Delta\psi_i$ at each node.

To prevent numerical overshoot, a successive relaxation algorithm is incorporated into the steady state solution:

$$\psi^{k+1} = \psi^k + \omega \Delta\psi^{k+1} \tag{3-21}$$

where ω is the successive relaxation parameter. Under-relaxation is achieved by setting ω to a value less than 1.0, and typically works best at ω approximately 0.5. Despite the use of ω, in some cases it is still necessary to use a less stringent convergence criteria than that which works for the transient solution.

3.3 Flux and Velocity Calculations

Once the pressure distribution has been determined at any given time step, or at steady state, flux and velocities are calculated across each element in the flow domain. Element fluxes and velocities were chosen for two reasons: (1) soil properties are assigned by element, and (2) a potential gradient across each element can be determined by the calculated ψ values at the two nodes which define a given element.

The flux term developed from a generalization of Darcy's law (2-2) is approximated numerically as follows:

$$q_e = -K_e(\psi)\left[\frac{\psi_{i+1} - \psi_i}{\Delta z_e} + 1.0\right] \tag{3-22}$$

where q_e is the element flux $[LT^{-1}]$, $K_e(\psi)$ is the interblock, geometric-mean conductivity $[LT^{-1}]$, ψ_{i+1} and ψ_i are the potentials at the bottom and top nodes of a given element respectively [L], and Δz_e is the element thickness [L]. Note that a negative value for q_e indicates downward flux, whereas a positive value indicates movement up, into the liner. Equation (3-22) can be written as:

$$q_e = -K_e(\psi)\left[\frac{\psi_{i+1} - \psi_i}{\Delta z_e}\right] - K_e(\psi) \tag{3-23}$$

where the first term on the RHS of (3-23) represents flux due to the matric potential gradient and the second term represents flux due to gravitational potential. Upon reaching steady state, all element fluxes should be equal thus satisfying (2-1b).

A conservative approach, with respect to liner design life, is provided in SOILINER for element velocity calculations:

$$\bar{v}_e = q_e/\theta_e \qquad (3\text{-}24)$$

where $\bar{v}_e$ is the interstitial pore velocity for a given element, q_e is the element flux, and θ_e is the average moisture content for the nodes defining the element of concern. Calculations involving θ_e, as opposed to porosity (n), result in higher element velocities since the cross-sectional area of a partially saturated pore capable of transmitting fluid is less than the actual pore size (i.e., $\theta \leq n$). The higher velocity requires a thicker liner to contain leachate for a specified design life.

3.4 Method of Determining Breakthrough

Two previous methods investigated as a means of indicating breakthrough of lagoon constituents involved setting a criteria based on the change in θ at the liner base (Goode and Smith, 1984; Johnson and Wood, 1984). However, problems associated with these approaches limit their applicability. The primary difficulty is conceptualizing the relationship between changes in θ and the migration of lagoon constituents. A change in θ at the liner base can be attributed to the displacement of the initial moisture from upper pores by infiltrating liquid, not the actual leachate. Finally, it has been shown that under most conditions a steady state ψ distribution (and thus θ distribution) is achieved sooner than a particle can be advected across the liner.

For the reasons stated above, a particle-tracking algorithm has been included in the SOILINER model. This algorithm bases particle movement on advection only and does not take into account the effects of dispersion. However, element velocities used to advect the particle are a function of space and time (i.e. $v_e = f(z,t)$) and thus reflect the dynamics of decreasing velocities over time near the wetting front.

The particle is initially positioned at the liner surface and a velocity for the first element is determined at $t_o = 0$. A new ψ distribution is then calculated at $t_1 = t_o + \Delta t$ (as discussed in Section 3.2) from which a second element velocity is calculated. Since the time-integration scheme employs Simpson's rule, another time step is completed before particle movement. At the time of particle movement, SOILINER then numerically integrates the velocity function to determine the distance traveled over the time period integrated. Finally, the total distance traveled over time is updated and the particle is "located" with respect to the grid geometry such that new element data is available for velocity calculations. Breakthrough occurs when the particle passes a specified node point, generally set at the liner/underlying-soil interface.

4. VERIFICATION

In order to verify the numerical technique used by SOILINER, the model's results were compared to analytical solutions for three physically realistic problems. The first two simulations are for steady state unsaturated flow and the third is for infiltration under ponding into an unsaturated clay.

Two grids were used for SOILINER's simulations. For each grid, node 1 is at the column top, z = 0, and the last node is at the column bottom, z = -50 cm. The first grid divided this 50 cm column into 50 1 cm elements (51 node points). The second grid divided the soil column as follows: the first 5 cm was divided into 10 elements; the next 25 cm into 25 elements; and the final 20 cm into 10 elements for a total of 45 elements and 46 nodes.

4.1 No-Flow Steady State

The governing equation for steady state unsaturated flow is:

$$\frac{\partial}{\partial z}\left(K(\psi)\ \frac{\partial \psi}{\partial z}\right) + \frac{\partial K(\psi)}{\partial z} = 0$$

or

$$\frac{\partial}{\partial z}\left\{K(\psi)\left[\frac{\partial \psi}{\partial z} + 1\right]\right\} = 0 \qquad (4\text{-}1)$$

which states that the flux is constant in space. The pressure boundary conditions to (4-1) determine the direction and rate of flow.

At steady state no-flow, the piezometric gradient is zero:

$$\frac{\partial \phi}{\partial z} = \frac{\partial(\psi + z)}{\partial z} = 0 \qquad (4\text{-}2)$$

Thus, the capillary pressure head gradient is equal to -1:

$$\frac{\partial \psi}{\partial z} = -\frac{\partial z}{\partial z} = -1 \qquad (4\text{-}3)$$

The boundary conditions for this situation are

$$\psi = 0 \qquad z = z_w \qquad (4\text{-}4)$$

at the water table at the bottom of the soil column and

$$\psi = z_w - z_t \qquad z = z_t \qquad (4\text{-}5)$$

at z_t, the soil column top. Between the column bottom and top, capillary pressure head is inversely proportional to elevation:

$$\psi = z_w - z \qquad z_w \leq z \leq z_t \qquad (4\text{-}6)$$

Equation (4-6) is an analytical solution to (4-1) with boundary conditions (4-4) and (4-5).

For the SOILINER simulations, the top node (z_t = 0cm) capillary pressure head is ψ = -50 cm and the bottom node (z_w =-50cm) capillary pressure is ψ = 0 cm. Simulations with both constant node spacing and variable node spacing produce the exact analytical solution to five significant digits.

4.2 Steady State Evaporation Flow Upward

Gardner [1958] developed analytical solutions for steady state unsaturated flow when the unsaturated hydraulic conductivity follows the following form:

$$K(\psi) = \frac{a}{(-\psi)^n + b} \tag{4-7}$$

where a and b are constants, and n is 1, 1/2, 2, 3, or 4. For n = 4, the analytical solution to (4-1) is given implicitly by:

$$z = \frac{1}{r}\left[\frac{1}{4\rho^3\sqrt{2}} \ln\left(\frac{\psi^2 - \rho\psi\sqrt{2} + \rho^2}{\psi^2 + \rho\psi\sqrt{2} + \rho^2}\right) + \frac{1}{2\rho^3\sqrt{2}} \tan^{-1}\left(\frac{\rho\psi\sqrt{2}}{\psi^2 - \rho^2}\right)\right] + W \tag{4-8}$$

in which

$r = q/a$

$\rho^4 = ß/r$

$ß = rb + 1$

where q is the discharge rate, constant in space and time, and W is a constant of integration. For a water table at an elevation of z_w = -50 cm and flow upward, W = -50 cm.

Using (4-7) for the functional soil hydraulic conductivity, SOILINER was run with boundary conditions of

$$\psi = 0 \text{ at } z_w = -50 \text{ cm}$$

and (4-9)

$$\psi = -200 \text{ cm at } z = 0$$

The flux computed by SOILINER was then used to compute Gardner's solution (4-8) using the computer program listed in Appendix B.

Figure 4-1 shows the agreement between SOILINER simulations with a regular grid (Test 2A) and a graded grid (Test 2C) and Gardner's analytic solution.

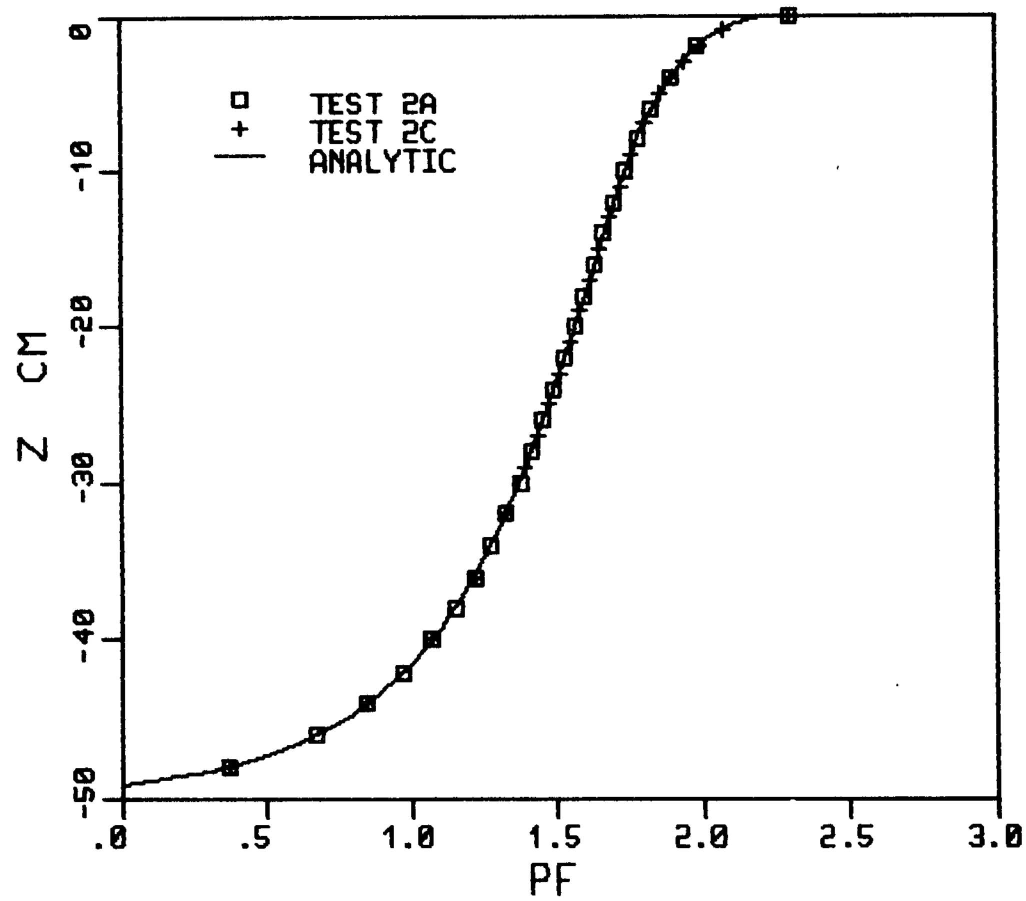

Figure 4-1. Comparisons of solutions for steady state vertical flow upward from a water table: solid curve is analytical solution of Gardner (1958); symbols are SOILINER numerical solutions at every other node. Test 2A (□) is a regular spaced grid with 50 nodes, Test 2C (+) is a variable spaced grid with 46 nodes. (pF = log $(-\psi)$, ψ in cm.)

4.3 Transient Infiltration

Philip [1958, 1969] developed a quasi-analytical solution to transient infiltration into unsaturated soil and applied this technique to simulate infiltration into the Yolo light clay with a porosity of 0.495 and a saturated hydraulic conductivity of 1.23 x 10^{-5} cm/s. The functional relationships for the unsaturated hydraulic conductivity and moisture characteristic were taken from Haverkamp et al. [1977] and are shown in Milly [1982]. The initial boundary conditions for this problem are

$$\psi = -600 \text{ cm} \qquad \text{all } z,\ t \leq 0$$

$$\psi = 25 \text{ cm} \qquad z = 0,\ t > 0$$

$$\psi = -600 \text{ cm} \qquad z = -\infty,\ t > 0$$

The soil is initially at a constant moisture content and capillary pressure over its entire thickness. For the SOILINER simulations, the initial time step was 0.1 sec and subsequent time steps were automatically calculated to keep the maximum pressure change between time steps at any node to about 10 cm.

Figure 4-2 shows a comparison of SOILINER using a regular grid with Philip's quasi-analytic solution. The agreement for this highly nonlinear problem is very reasonable. Figure 4-3 shows the similar accuracy of SOILINER with a graded grid. The graded grid has fewer nodes, but because of smaller node spacings at the top of the column, the initial moisture profile is closer to Philip's results, demonstrating the value of variable grid spacing.

SOILINER is shown to accurately simulate the flow of moisture in a vertical unsaturated column using both regular and graded grids. The infiltration test is very similar to the application of SOILINER in liner design. Successful completion of these verification runs using other computer models would similarily indicate their utility for soil liner design.

4.4 Particle Tracking

To verify the particle tracking algorithm, a fully saturated clay liner was simulated using SOILINER for comparison with a calculated breakthrough time based on Darcy's Law (equation 2-2). The following boundary and initial conditions were imposed on a 90 cm clay liner having a conductivity of 1.0x10^{-7} cm/sec and porosity of 0.495: (1) impoundment depth of 100.0 cm, (2) pressure at the liner base of 0.0 cm, and (3) a steady state pressure distribution within the liner at time t = 0. As required for steady state conditions, the change in flux over space is zero (i.e., flux everywhere in the liner is constant - see equation 2-1b). Since the entire clay liner is saturated, moisture content is also constant. Thus, regardless of the particle location in the liner, its velocity will remain unchanged. Figure 4-4 reveals this constant velocity up to the point of breakthrough at year 6.69.

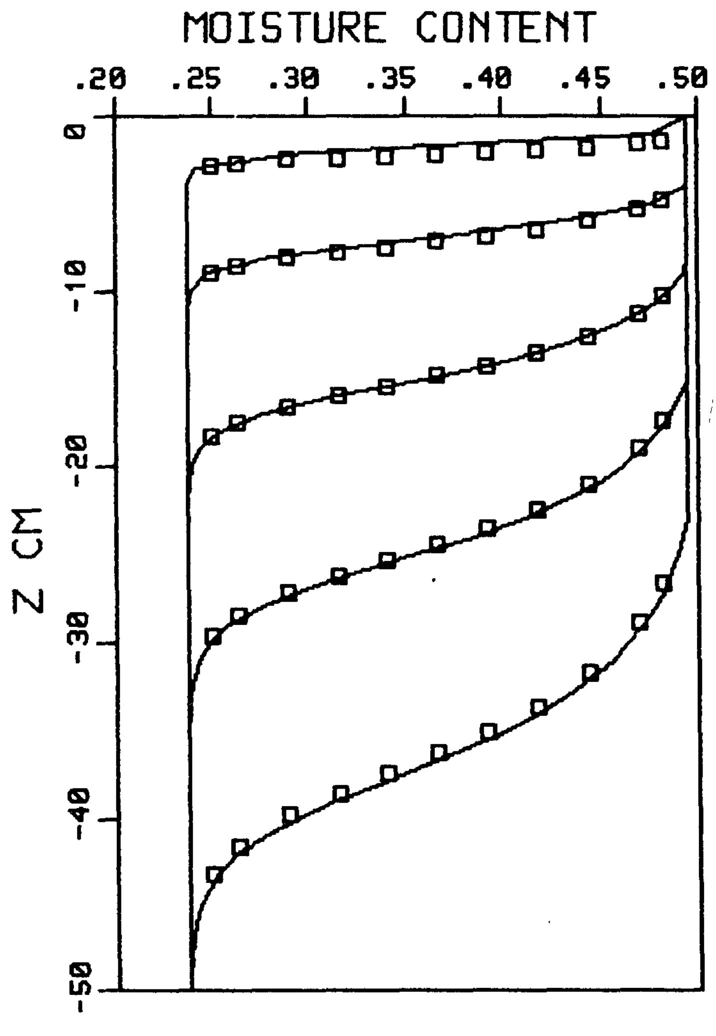

Figure 4-2. Comparison of Philip's quasi-analytic solution (□) and SOILINER regular grid solution (solid line) for infiltration into Yolo light clay under ponding. Curves are at 10^3, 10^4, 10^5, and $2x10^5$ sec.

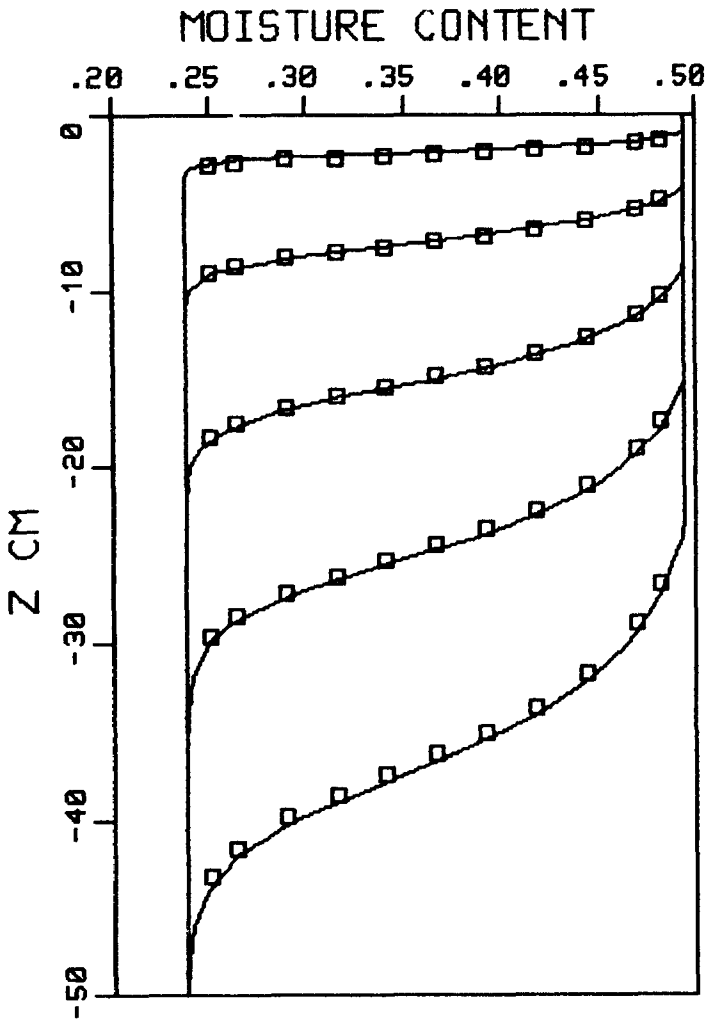

Figure 4-3. Comparison of Philip's quasi-analytic solution (□) and SOILINER graded grid (46 nodes) solution (solid line) for infiltration into Yolo light clay under ponding. Curves are at 10^3, 10^4, 10^5, and $2x10^5$ sec.

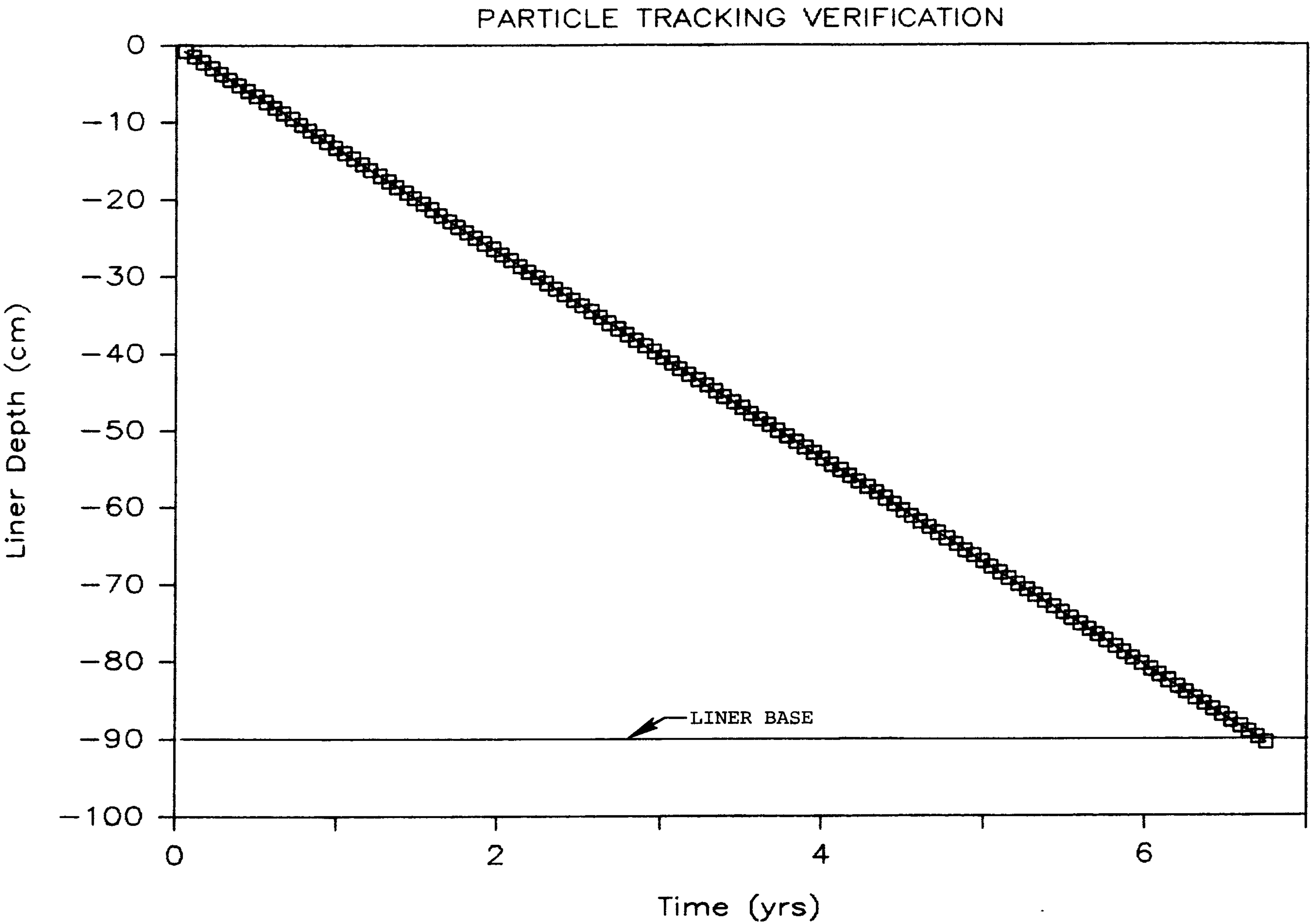

Figure 4-4. Change in particle depth with time across a fully saturated clay liner under steady state conditions.

For comparison, an estimated breakthrough time under the same initial and boundary conditions (across the entire liner thickness) can be obtained from equations 3-23 and 3-24:

$$q = 1.00\text{x}10^{-7} \text{ cm/sec} * \left[\frac{0-100.0}{-90.0} + 1 \right]$$

$$= 2.11\text{x}10^{-7} \text{ cm/sec}$$

$$v = 2.11\text{x}10^{-7} \text{ cm/sec} \div 0.495$$

$$= 4.26\text{x}10^{-7} \text{ cm/sec}$$

$$t = 90 \text{ cm} \div 4.26\text{x}10^{-7} \text{ cm/sec}$$

$$= 2.11\text{x}10^{8} \text{ sec} \quad 3.1536\text{x}10^{7} \text{ sec/yr}$$

$$= 6.69 \text{ years}$$

This calculation compares accurately with the breakthrough time of 6.69 years obtained from SOILINER.

PART II

SOILINER USER'S MANUAL

1. INTRODUCTION

1.1 History of the SOILINER Computer Code

The SOILINER computer model was originally developed at GCA under contract to EPA's Office of Solid Waste, Land Disposal Branch. The first version was completed in September, 1983. In April, 1984 the SOILINER report was released by EPA as a Technical Resource Document (EPA/530-SW-84-001; Goode and Smith, 1984) for public comment. This document emphasized the theoretical development of the finite-difference technique utilized in SOILINER and the ability to accurately simulate an infiltration event.

In November 1984 GCA investigated the times to breakthrough and steady state, with associated fluxes, for a variety of clay liner scenarios using SOILINER (Johnson and Wood, 1984). At that time, SOILINER was not specifically coded for the determinations of breakthrough and steady state. However, data obtained from the calculated moisture contents provided by SOILINER served as the basis for determining these times. The results of that study indicated that a moisture-based criterion for establishing breakthrough was not appropriate (see Section 3.4 of Part I) and a less subjective method should be developed.

The fundamental basis for simulating an infiltration event remains unchanged in the newest version of SOILINER. However, the original model has been modified to determine the times at which steady state and breakthrough occur. Breakthrough is determined from a particle tracking algorithm which is discussed further in Section 2.

1.2 Scope of the Manual

The remainder of this document is devoted to the development of a step-by-step procedure for the application of SOILINER. Section 2 discusses in detail the features and limitations of SOILINER such that it becomes apparent which flow scenarios can be accurately simulated. Section 3 provides a general description of the data necessary as input to the model. This section also discusses the development of characteristic curves for moisture content and unsaturated hydraulic conductivity, both a function of matric potential. It is important to note that these two relationships form the basis of an accurate simulation. Although a number of characteristic curves, covering a wide range of soil types, have already been coded into the model, site specific data can be easily incorporated into SOILINER.

Section 4 presents the findings from a thorough testing of the SOILINER model for a variety of solution strategies. Testing includes the investigation of modifications to the original code and new programming features. A number of simulations have been made, from which specific guidelines were established describing appropriate values for the model parameters for a variety of clay-liner scenarios.

1.3 Computer Requirements

SOILINER was originally written in FORTRAN IV for an IBM 3033 mainframe. The current single-precision version is dimensioned for a maximum of 200 node points. The average run-time for a 100± node flow domain over a considerable time period (3 to 5 years) takes between 30 and 60 seconds of CPU-time.

A modified version of SOILINER has been run successfully on an IBM-PC (256 K), and requires approximately 128 K-bytes of RAM. Even with an Intel 8087 math co-processor, average run-times are considerably longer.

Both versions require 4 input files and 5 output files. Input files include the following:

(1) Grid File - configures node coordinates.

(2) Properties File - configures element soil properties by layer, including the saturated properties of the soil from which the characteristic curves were developed.

(3) Initial Conditions file - configures initial ψ distribution.

(4) Control File - contains parameters which define the chosen solution strategy including the number of nodes, time stepping parameters, and boundary conditions.

Output files include:

(1) General Output - this file includes an echo of input data and pertinent output by time step including moisture, pressure, and conductivity results.

(2) Matric Potential, and (3) Moisture Content - these two files repeat data from the General Output File but are reproduced separately for graphical analysis.

(4) Flux Output - this file contains element flux data by time step, and can be used to demonstrate flux divergence (note that negative flux values indicate downward movement, out of the liner whereas positive values indicate movement up, into the liner).

(5) Velocity Output - this file contains data relevant to particle tracking and can be used to represent the dynamics of infiltration and the time to breakthrough.

The use of separate files facilitates input data manipulation and the graphical analysis of output. Individual parameters and their format are discussed further in Appendices C and D.

2. MODELING CAPABILITIES

2.1 Model Assumptions

SOILINER was specifically developed to simulate vertical, unsaturated flow beneath ponded liquid. The assumptions implicit in the governing equation (2-1) are:

(1) the air phase is always at a spatially constant atmospheric pressure;

(2) the fluid has a constant density and does not freeze;

(3) flow is not affected by temperature gradients or solute concentration gradients; and

(4) the medium does not deform.

The last assumption must be carefully considered, especially for liners having a high percentage of clay which may be subject to leachate interaction. Under extreme conditions, dissolution and piping (the actual loss of clay particles to underlying strata) may occur due to the chemical nature of an impounded liquid. Other constituents have been shown to alter the structure of clay barriers. In any case, liner deformation invalidates the characteristic curve for the soil under consideration. However, most hazardous waste leachates are water-based. While strongly concentrated leachates may increase permeability, no examples were found in the literature where very dilute aqueous solutions caused substantial permeability increases (Anderson and Jones, 1983).

One final consideration is specific to the numerical representation of the governing equation (2-1). Most of the characteristic curves available in the SOILINER model were developed utilizing data obtained from the desorption cycle. Consequently, SOILINER does not account for hysterisis.

2.2 Solution Strategies

SOILINER is capable of running transient and/or steady state solutions depending on the combination of parameters established in the control file. If a steady state solution only is desired, the ISTEDY integer flag must be set equal to 1. For a transient simulation ISTEDY is set equal to 0, and a MAXNT is established such that all desired output times are reached for the given combination of time stepping parameters (see Section 4). If the transient option is chosen, the steady state algorithm is first solved as a means of determining the time at which steady state is achieved during transient simulations. Additional parameters for both the steady state and transient solution strategies are covered in Section 4 and Appendices C and D.

2.3 Definition of Clay Liner Domain

The size and content of the clay liner domain depends on the limits imposed on the system such that boundary conditions can be specified. The upper boundary should always be specified at the clay-liner upper surface and will generally have ψ set equal to the impoundment depth. Ideally the lower boundary would be established at the water table where $\psi = 0$. This domain may include a large portion of the underlying site soil, and would require discretization of both the liner and native soil. However, it is possible to model only the clay liner if the lower boundary condition can be reasonably estimated. When a sharp discontinuity in hydraulic conductivity exists between the liner and native soil (i.e. 1.0×10^{-7} to 1.0×10^{-2} cm/sec), the lower boundary can be set to reflect the height above a water table. For comparison, Figure 2-1 depicts the changes in ψ over the period of infiltration for a 90 cm clay liner ($K = 1.0 \times 10^{-7}$ cm/sec) underlain by 300 cm of an unsaturated, sandy soil ($K = 9.4 \times 10^{-3}$ cm/sec). Figure 2-2 depicts the ψ distributions for a similar domain where the only difference is a decreased conductivity for the underlying, native soil ($K = 1.0 \times 10^{-5}$ cm/sec). As shown in Figure 2-1, the initial ψ of -300 cm at the clay/sand interface increased only 9 cm to -291 cm, whereas the ψ increased from -300 cm to -137 cm if the sand were replaced with a silt (see Figure 2-2).

SOILINER is also capable of modeling layered flow domains. To demonstrate this ability, a two liner system was simulated using SOILINER. The hypothetical liner was constructed as follows:

Layer 1: 61 cm (2') clay; $K = 1.0 \times 10^{-7}$ cm/sec, n = 0.495

Layer 2: 30 cm (1') sand; $K = 9.44 \times 10^{-3}$ cm/sec, n = 0.287

Layer 3: 91 cm (3') clay; $K = 1.0 \times 10^{-7}$ cm/sec, n = 0.495

Native Soil: 300 cm to water table (PSI = 0.0); $K = 9.44 \times 10^{-7}$ cm/sec, n = 0.287

where the impoundment depth was set equal to 100 cm. To simulate worst case conditions, a leachate collection system was not included in the upper sand layer.

The change in pressure with time shown in Figure 2-3 reveals the dynamic interaction between layers in this liner configuration. Figure 2-4 depicts the corresponding changes in moisture content. Note that a drop in ψ

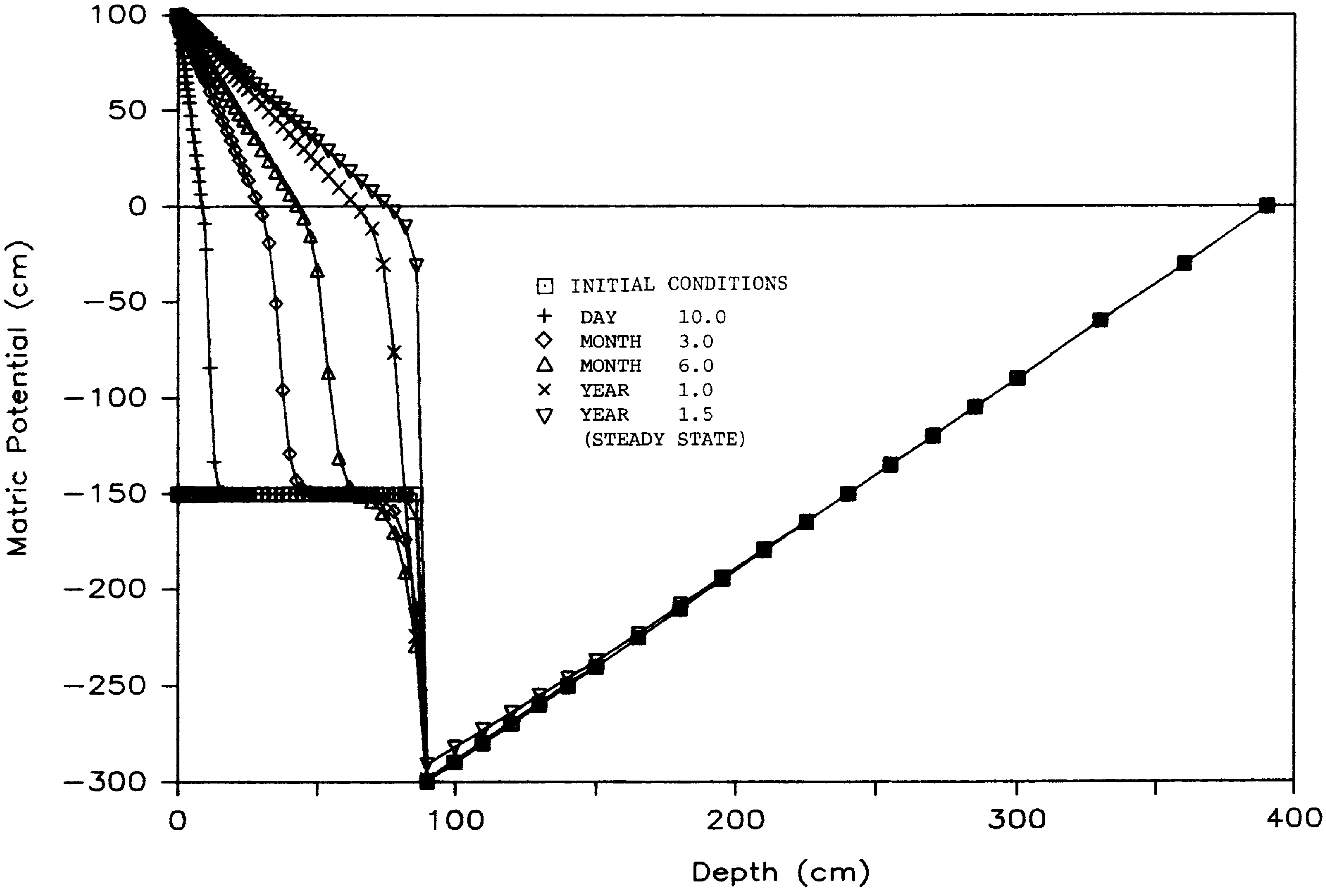

Figure 2-1. Changes in pressure distribution for a clay liner underlain by a high conductivity native soil.

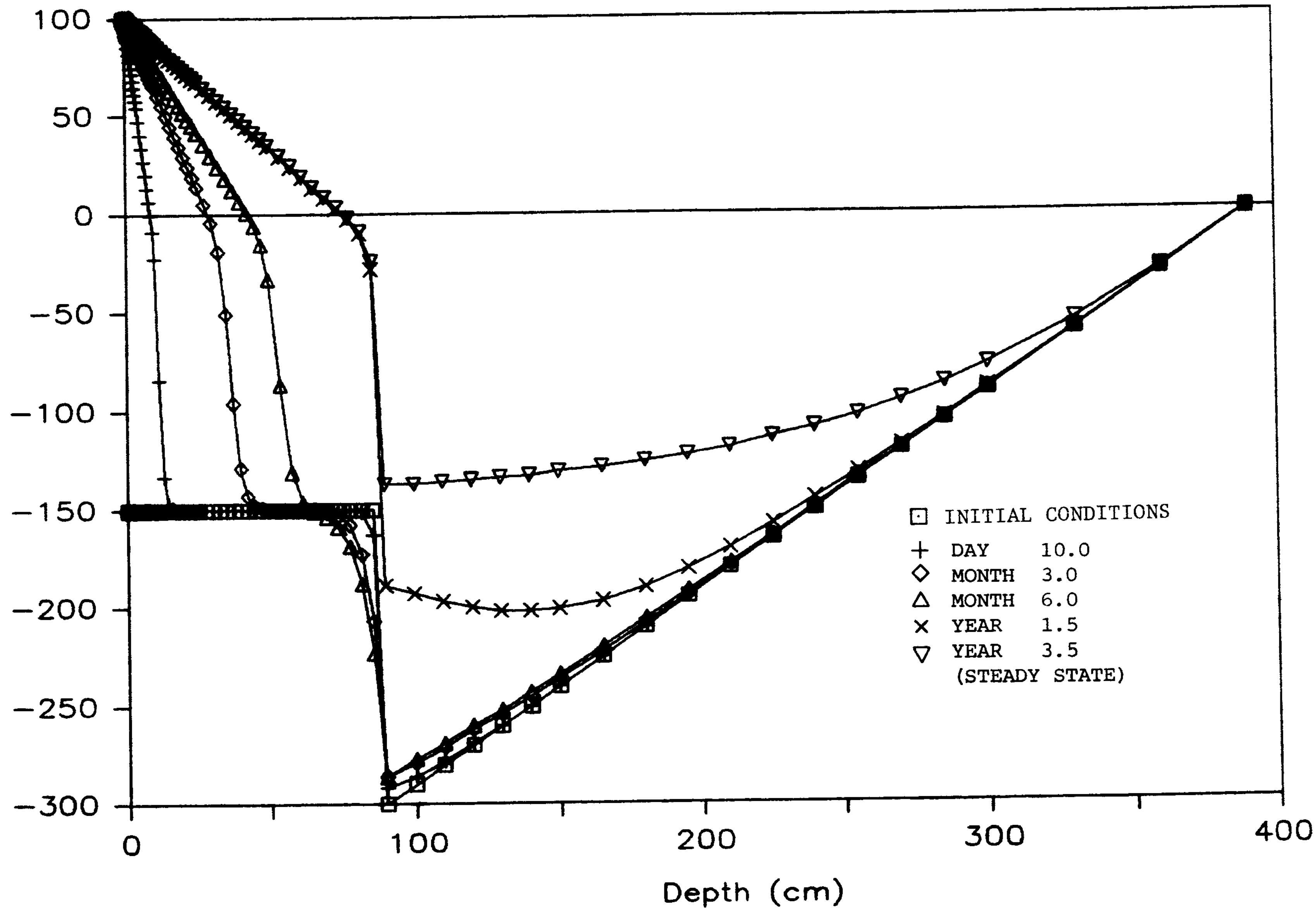

Figure 2-2. Changes in pressure distribution for a clay liner underlain by a low conductivity native soil.

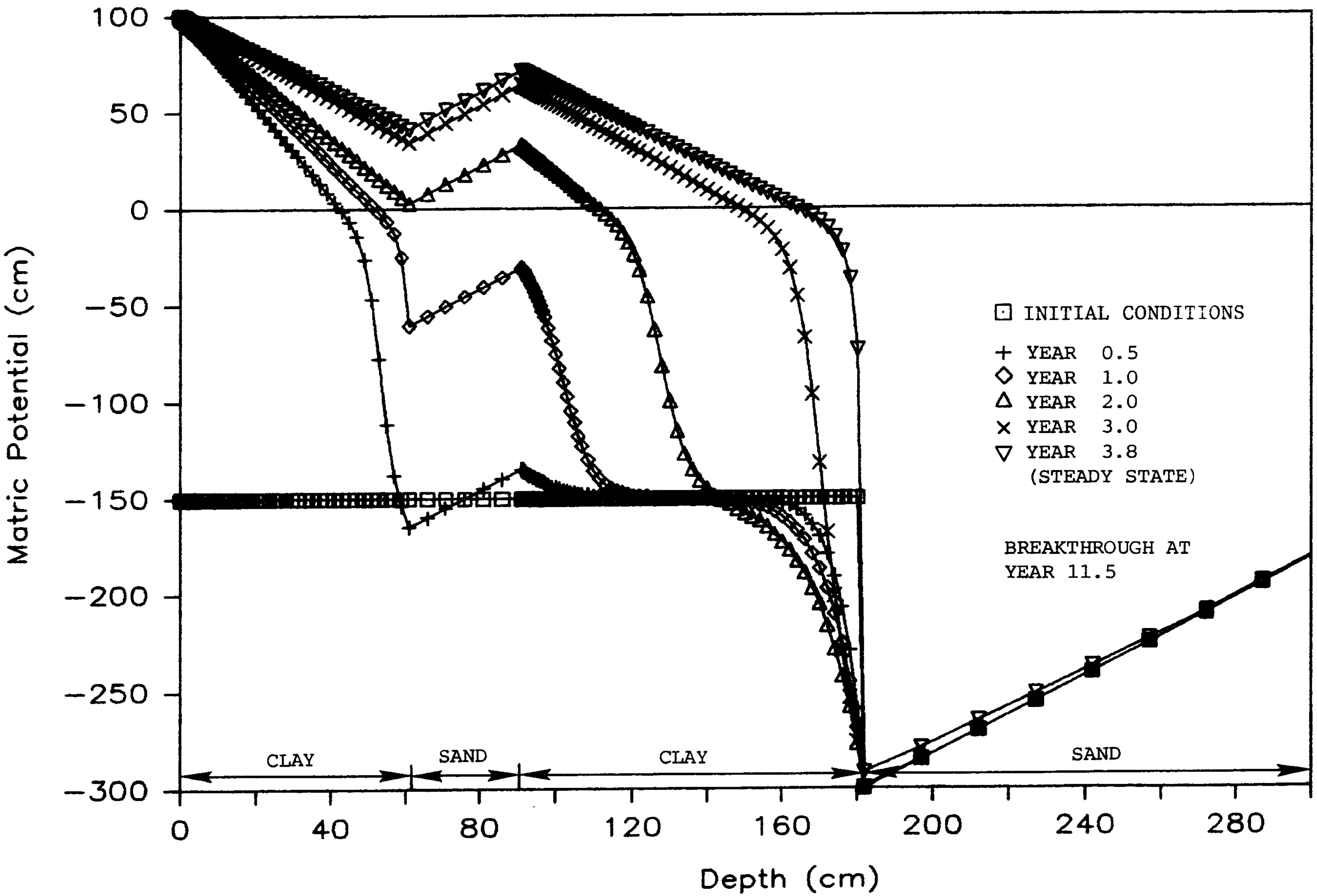

Figure 2-3. Changes in pressure with time in the upper 300 cm of a 482 cm clay-sand-clay liner configuration.

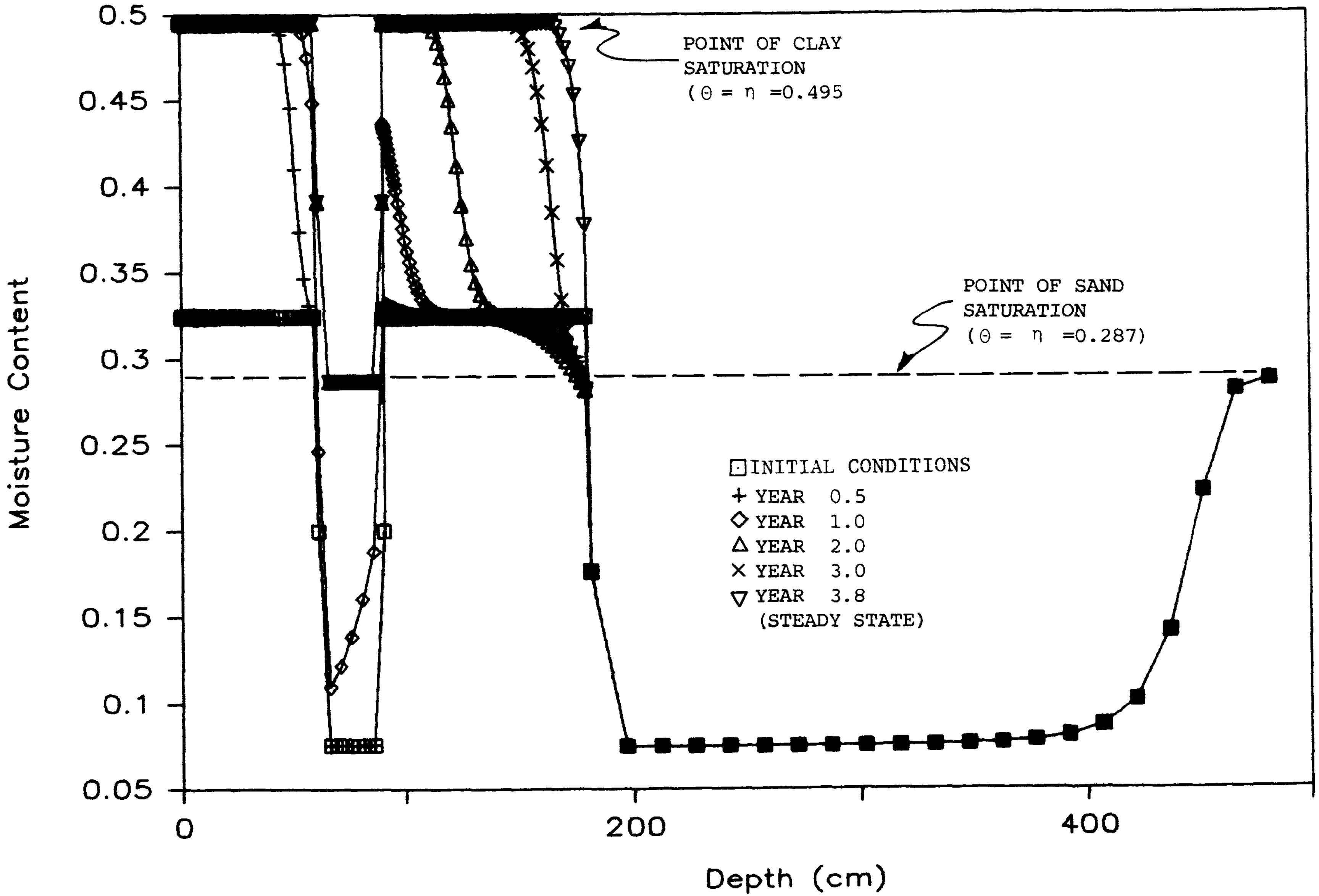

Figure 2-4. Changes in moisture content with time corresponding to pressure curves shown in Figure 2-3.

indicates moisture loss, whereas an increase in ψ indicates moisture gain. Many salient features of the infiltration event can be drawn from these graphics:

(1) The upper portion of the first sand layer loses moisture at a rate greater than replenishment from the overlying clay (see year 0.5 where ψ in the sand drops below initial conditions); this net loss is offset by moisture gain in the clay immediately below the sand.

(2) For the first three time steps moisture is lost from the bottom clay liner base, as indicated by the ψ drop which propagates into the liner.

(3) Overall, pressure becomes less negative as pores reach their steady state moisture content; note that pores initially drained at the liner base are rewetted from infiltrating liquid sometime after year 2.

(4) Although ψ increases with depth in both sand layers, the total head (ψ+ z) always decreases with depth resulting in downward flux.

(5) At steady state, the entire two-liner system (181 cm) is saturated to a depth of 164 cm, at which point ψ values drop below 0.0 with increasing depth.

For this liner configuration with an initial pressure of -150 cm, steady state was achieved in 3.8 years and particle breakthrough occurred at year 11.5.

2.4 Determining Breakthrough

SOILINER utilizes a particle tracking algorithm to determine the time required for a particle to be advected across a specified liner thickness. The effects of dispersion are not accounted for, and thus, breakthrough represents the average linear velocity of migrating contaminants. The liner thickness of concern is established by setting NCLAYN equal to the bottom node of the liner. The particle is initially positioned at the liner surface. Subsequent movement is then determined from velocity data generated by SOILINER. Figure 2-5 depicts the dynamics of particle movement across a 60 cm clay liner. The initial velocities are high as shown by the steeply sloping curve near the liner surface (i.e., Δz great with respect to Δt). With time, velocities decrease as the wetting front penetrates the liner.

2.5 Variable Initial and Boundary Conditions

Use of the numerical technique not only enables one to vary the soil properties, but also allows variable initial conditions to be specified by the

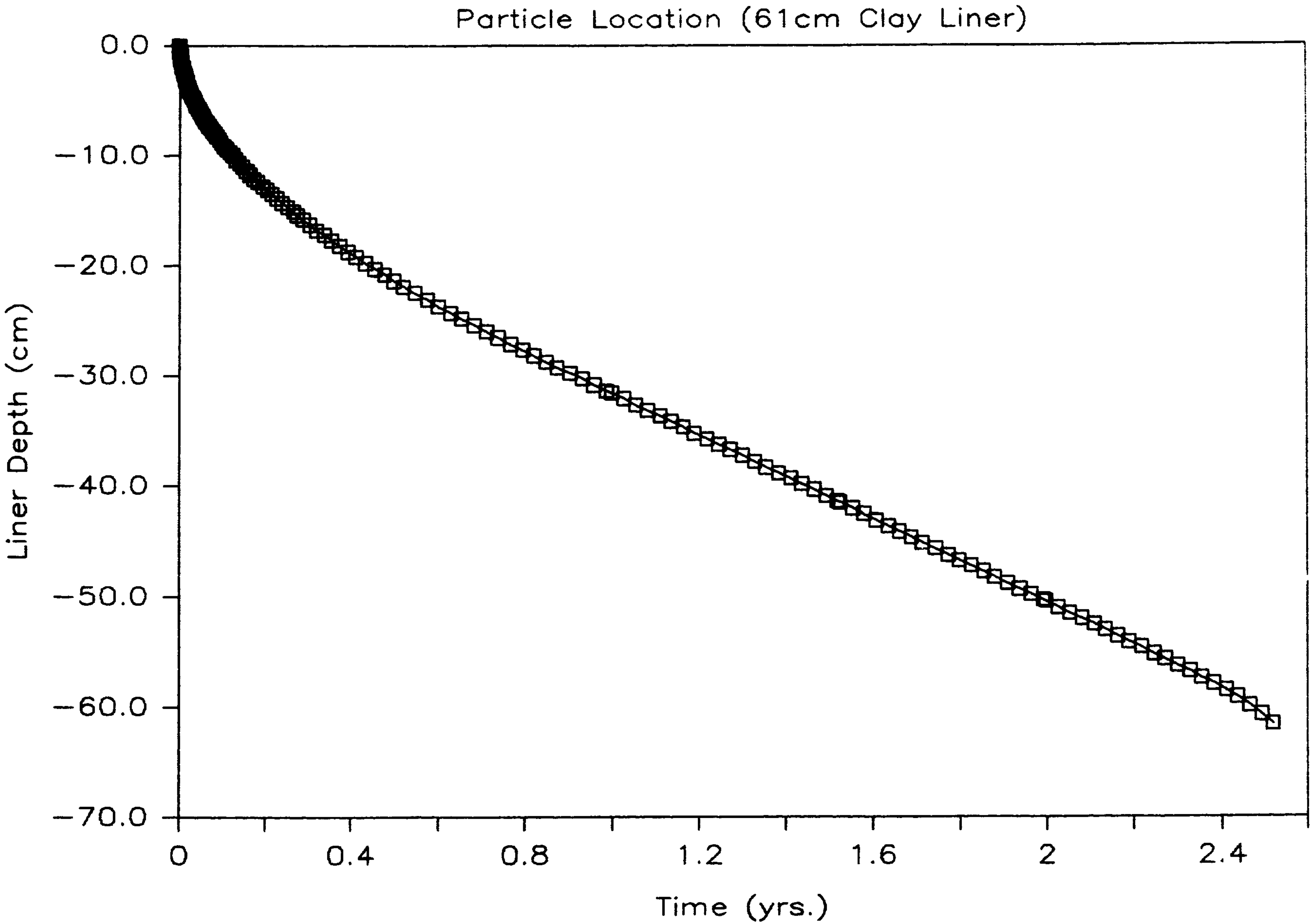

Figure 2-5. Changes in particle depth depicting the dynamic characteristics of the infiltration event, specifically decreasing wetting front velocity with respect to time.

user. It may be reasonable to assume a constant pressure for a carefully constructed, homogeneous clay liner in which the moisture content for each lift is monitored and adjusted. However, multilayered liners are likely to exhibit widely varying initial pressures and associated moisture contents. Also, the pressure distribution in native soils prior to liner construction will vary, generally becoming more negative with increasing height above the water table (see Section 2, Part I). By breaking the continuous flow domain into a limited number of discrete node points for the finite-difference technique, it is possible to specify different initial pressures at various depths in the liner domain. This inherent ability also facilitates investigation of the effects on infiltration due to time-varying boundary conditions.

At any specified point in time (t_1) it is possible to predict the ψ pressure distribution for a given set of initial and boundary conditions. At that time it may also be desirable to determine the effect of an increase or decrease in impoundment depth (i.e. upper boundary condition). To achieve this objective with SOILINER, the user would simply employ the existing, calculated pressure distribution at time t_1 as initial conditions for subsequent simulation under the new impoundment depth. Once the new boundary conditions are specified, a desired set of special output times can be obtained which will reflect the changes imposed on the system.

Figure 2-6 depicts the initial ψ distribution and four special output times for a clay liner simulation where the impoundment depth was changed from 100 cm to 200 cm prior to particle breakthrough. For the first three output times (up to year 1.5) the impoundment depth was constant at 100 cm. By year 1.5 steady state was achieved and the particle had penetrated the liner to a depth of 36 cm. At this time the impoundment depth was increased to 200 cm. Within 2 months, a new steady state pressure distribution was reached (year 1.7).

The impact of increasing impoundment depth on particle movement is depicted in Figure 2-7. If the head (H) had remained constant at 100 cm, as shown by the dotted line, breakthrough would have occurred at approximately year 5. However, the change in head (ΔH) at year 1.5, increased the particle velocity due to an increased gradient, with particle breakthrough occurring one year earlier.

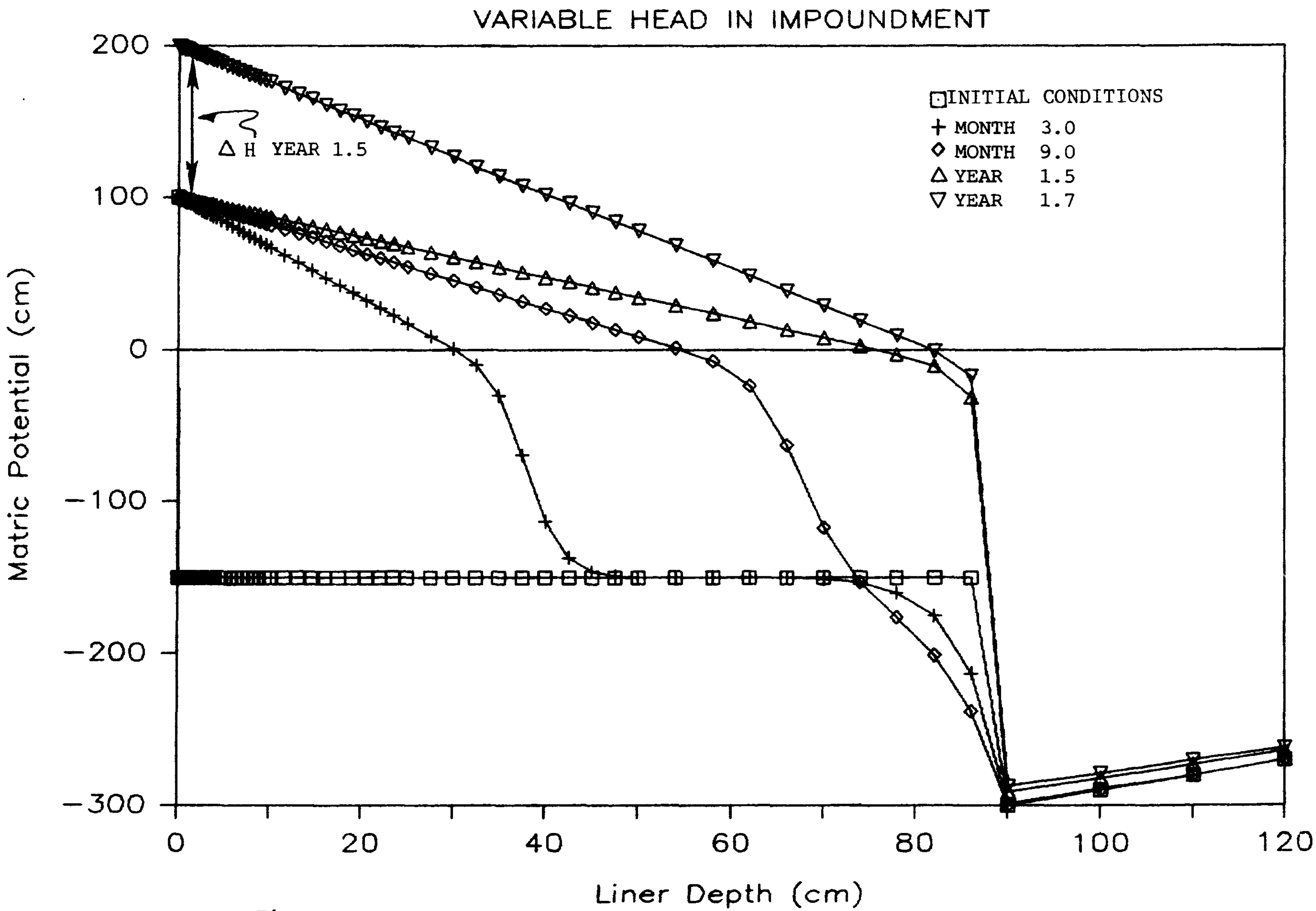

Figure 2-6. Change in steady state pressure distribution due to increased impoundment depth at year 1.5.

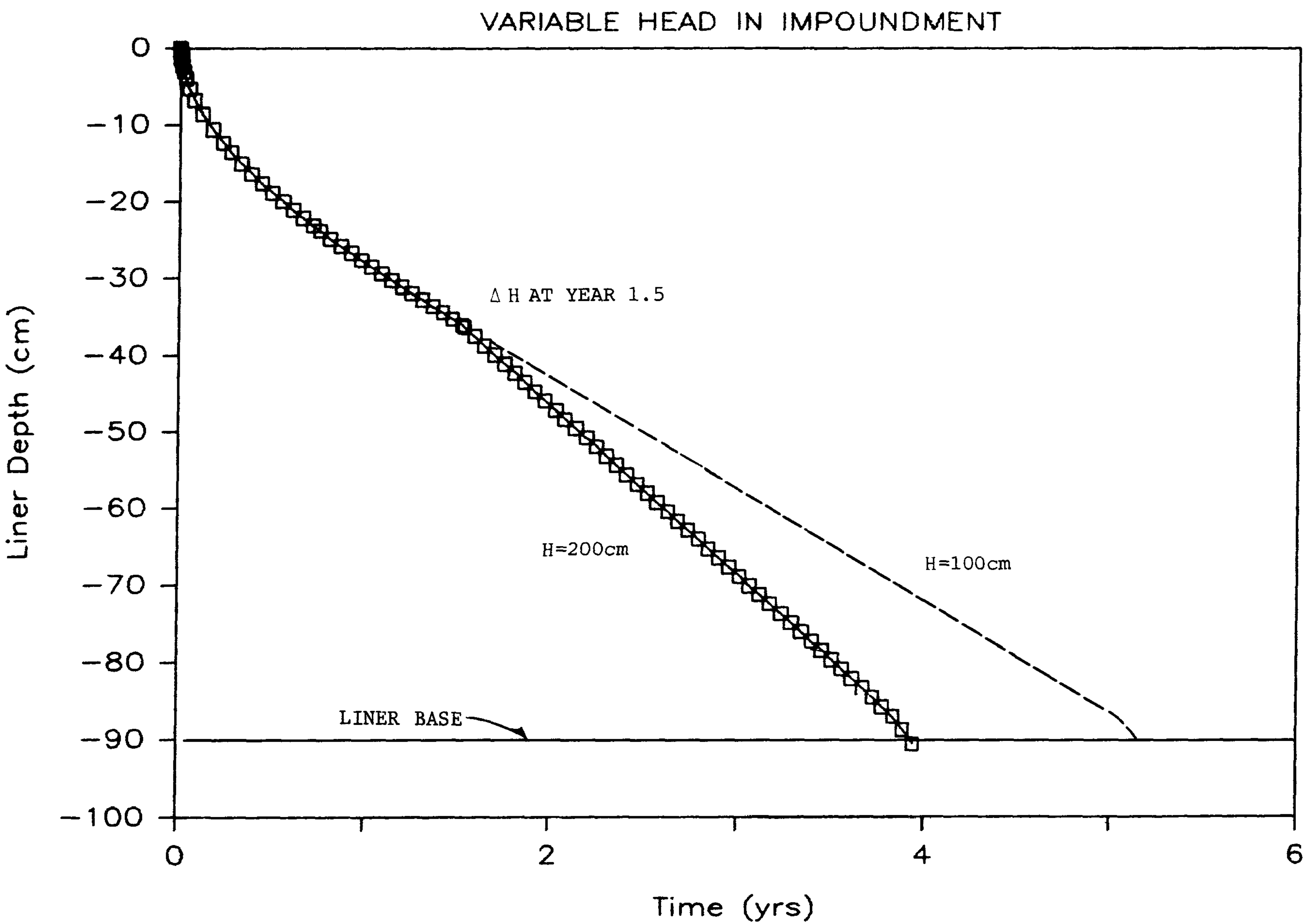

Figure 2-7. Effect of increased impoundment depth on the time to breakthrough.

3. DATA REQUIREMENTS

Data necessary to run SOILINER are related to either (1) the solution strategy, or (2) liner properties. Most of the parameters found in SOILINER are required solely for the development of a solution strategy (i.e. time stepping, grid discretization, etc.) to simulate a given set of conditions. Although very little field data is required, the method in which soil properties are obtained (and their accuracy) is critical and serves as the fundamental basis for the SOILINER model. Methods of determining soil moisture and conductivity as a function of matric potential (i.e. negative pressure) will be described in detail after a general discussion of the required input.

3.1 Description of Input Data

As mentioned previously, there are four input data files. Following is a listing, by file, of the input data.

Grid File:

- NUM - number of consecutive nodes with separation DELTA.
- DELTA - separation between adjacent nodes (cm).

Properties File:

- NUM - number of elements in soil layer.
- ISOIL - one-dimensional array holding soil-type code numbers ranging from 0 to the number of soils for which characteristic curves have been coded into the model (see Table 3-1, p. II-25).
- SATK - saturated conductivity for the associated soil type.
- POR - porosity of the associated soil type.
- PSICRT - critical pressure (see Section 3.3, Part II).

Initial Conditions File:

- NUM - number of consecutive nodes with initial pressure, PSI.
- PSI - initial pressure, PSI (cm).

Control File:

- TITLE - character variable used to describe each particular run.

LPRINT - Boolean flag for output format; T = long, F = short.

LSTEDY - Boolean flag; T = steady state solution, F = transient.

ENDTIM - desired total period of simulation (yrs).

DT - initial time step size (secs).

CHPARM - change parameter utilized to increment DT between successive time steps.

DTMAX - maximum allowable time step (secs).

MAXNT - maximum number of time steps.

ALPHA - time weighting parameter set between 0.0 and 1.0 for transient solutions.

ERRMAX - criteria established for the maximum change in ψ between successive iterations at any given time step (cm).

MAXIT - maximum allowable number of iterations at any given time step.

NOUT - number of special output times.

TOUT - one-dimensional array holding NOUT number of special output times (NOTE: if NOUT ≤ 0, TOUT must not be present; if NOUT = -1, all time steps are printed).

NUMNP - total number of nodes.

NCLAYN - number of liner nodes (utilized in determining breakthrough).

SRPARM - successive relaxation parameter ranging from 0.0 to 1.0.

H - depth of liquid impoundment (upper ψ boundary condition).

PSIBOT - pressure at the base of the flow domain (lower ψ boundary condition).

Review of the necessary input data reveals that the only field- or lab-oriented measurements are the initial pressures and the soil properties SATK and POR. What is not readily apparent, nor required as direct input from the user, are the characteristic curves for the specified soil types. The parameters H, PSIBOT, initial PSI, SATK, and POR are readily available (either measured or estimated) in comparison to the effort required to develop a characteristic curve. Subsequent reference to parameters listed in the above files will utilize their FORTRAN variable names when appropriate. Appendix C provides a more complete list of variables found in SOILINER.

3.2 Measurement of the Soil Moisture Characteristic Curves

In measuring the fundamental relationship between soil moisture and suction (matric potential x specific weight of water) it is better, in principle, to take in-situ measurements rather than measurements of disturbed samples (e.g., dried, screened, and artificially packed sample) in the laboratory. At low suction (0-1 bars) the soil moisture retention is strongly influenced by soil structure and pore size distribution (Hillel, 1980). The most preferable method in measuring the soil moisture characteristic is by making simultaneous measurements of wetness, with a neutron moisture meter, and suction, with a tensiometer. Unfortunately, this approach has often been frustrated by soil heterogeneity and by uncertainties over hysteretic phenomena as they occur in the field. The following paragraphs briefly describe these instruments for use in measuring soil moisture and suction in the field. The reader is referred to Hillel (1980) for more detail.

The tensiometer is the most widely accepted practical device for measuring matric potential in the field. As shown in Figure 3-1, the tensiometer consists of a porous cup connected through a tube to a manometer, where all parts are filled with water. The porous cup, which is generally made of ceramic material, is placed in the soil where the suction measurement is to be made. When initially placed in the soil, the water contained inside the tensiometer is generally at atmospheric pressure. The soil water surrounding the cup is most likely below atmospheric pressure. This causes a suction which draws a certain amount of water from the tensiometer, thus causing a drop in its hydrostatic pressure. This pressure is registered on the manometer, which can be a simple water or mercury filled U tube, a vacuum gauge, or an electrical transducer.

A tensiometer is placed in the soil for a long period of time to follow changes in the pressure of soil water. As moisture is depleted from the soil by evapotranspiration or drainage, or replenished by percolation or irrigation, corresponding readings on the tensiometer can be obtained.

During measurements of suction, simultaneous readings of soil moisture content must be obtained at the same locations in order to relate soil moisture with suction. An efficient and reliable instrument for measuring soil moisture is the neutron moisture meter. As shown in Figure 3-2, this instrument consists of a probe which is lowered into the soil, attached to a scalar or ratemeter. The probe contains a source of fast neutrons (obtain an average speed of 1,600 km/sec) and a detector of slow neutrons. The scalar monitors the flux of slow neutrons scattered by the soil.

In general, the probe emits fast neutrons into the surrounding soil where they encounter and collide with the atomic nuclei of hydrogen. When they collide, the fast neutrons lose some of their kenetic energy, thus becoming slow neutrons. Neutrons that have been slowed to a critical speed are said to be thermalized. As the thermalized neutrons repeatedly collide, and move about randomly, a number of them return to the probe where they are counted by the slow neutron detector. The amount of thermalized neutrons is proportional to the concentration of soil moisture.

3.3 Functional Relationship Formula

To formulate the two part function

$$\psi = \psi_s W^{-4} \text{ for } 0 \leq W \leq W_i \tag{3-1}$$

and

$$\psi = -m(W-n)(W-1) \text{ for } W_i \leq W \leq 1 \tag{3-2}$$

where m and n are:

$$m = \frac{\psi_i}{(1-W_i)^2} - \frac{\psi_i b}{W_i(1-W_i)} \tag{3-3}$$

$$n = 2W_i - \frac{\psi_i b}{mW_i} - 1 \tag{3-4}$$

in which the parameters ψ, ψ_s, W, b, W_i, and ψ_i are shown in Figure 3-3, soil moisture content, θ, and matric potential, ψ, must first be obtained using field techniques described in the previous section or from the literature (note that ψ_i and θ_s are PSICRT and POR, respectively).

To calculate soil wetness, W, the unsaturated moisture content, θ, must be divided by the saturated moisture content, θ_s or in this study, the total porosity. The saturated moisture content must be determined by first oven drying an undisturbed sample from the field, and then weighing it. It is then saturated with some liquid and weighed again. Finally, the saturated sample is immersed in the same liquid, and the weight of the displaced liquid is noted. The weight of the liquid required to saturate the sample divided by the weight of the liquid displaced is the porosity as a decimal.

Both the matric potential at saturation, ψ_s, and the exponent b are empirical and are estimated using a power curve regression analysis performed on the log transforms of the obtained data (ψ and W, where $W = \theta/\theta_s$). The value ψ_s is the log of the y-intercept, and b is the slope of the line.

To determine the values of the parameters m and n, the location of the inflection point, (W_i, ψ_i) in Figure 3-3 must be determined. This is conducted by first plotting the soil moisture matric potential data and

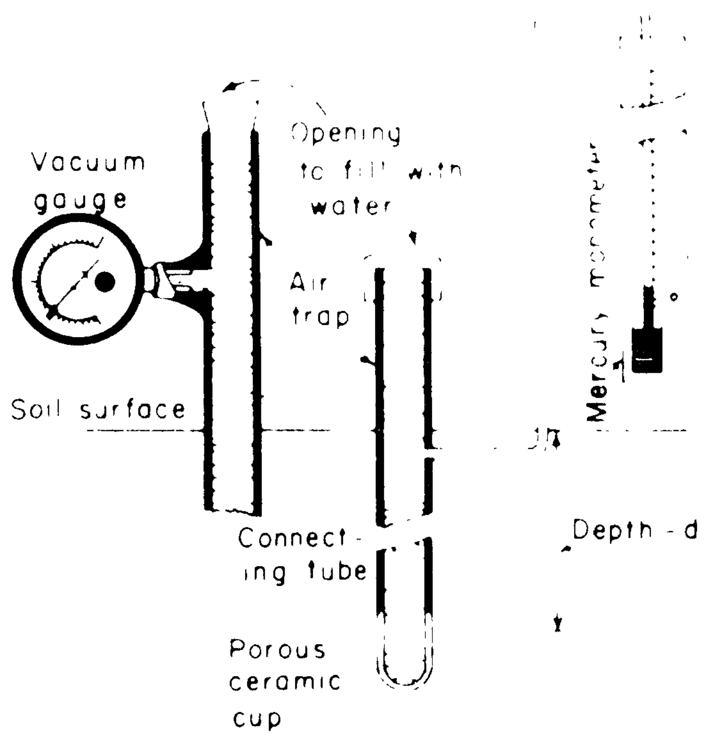

Figure 3-1. Illustration of the essential parts of a tensiometer (from Hillel, 1980).

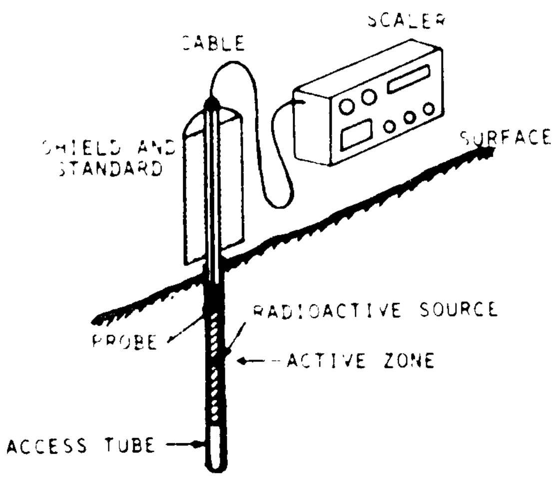

Figure 3-2. Illustration of the essential components of a portable neutron soil-moisture meter (after Hillel, 1980).

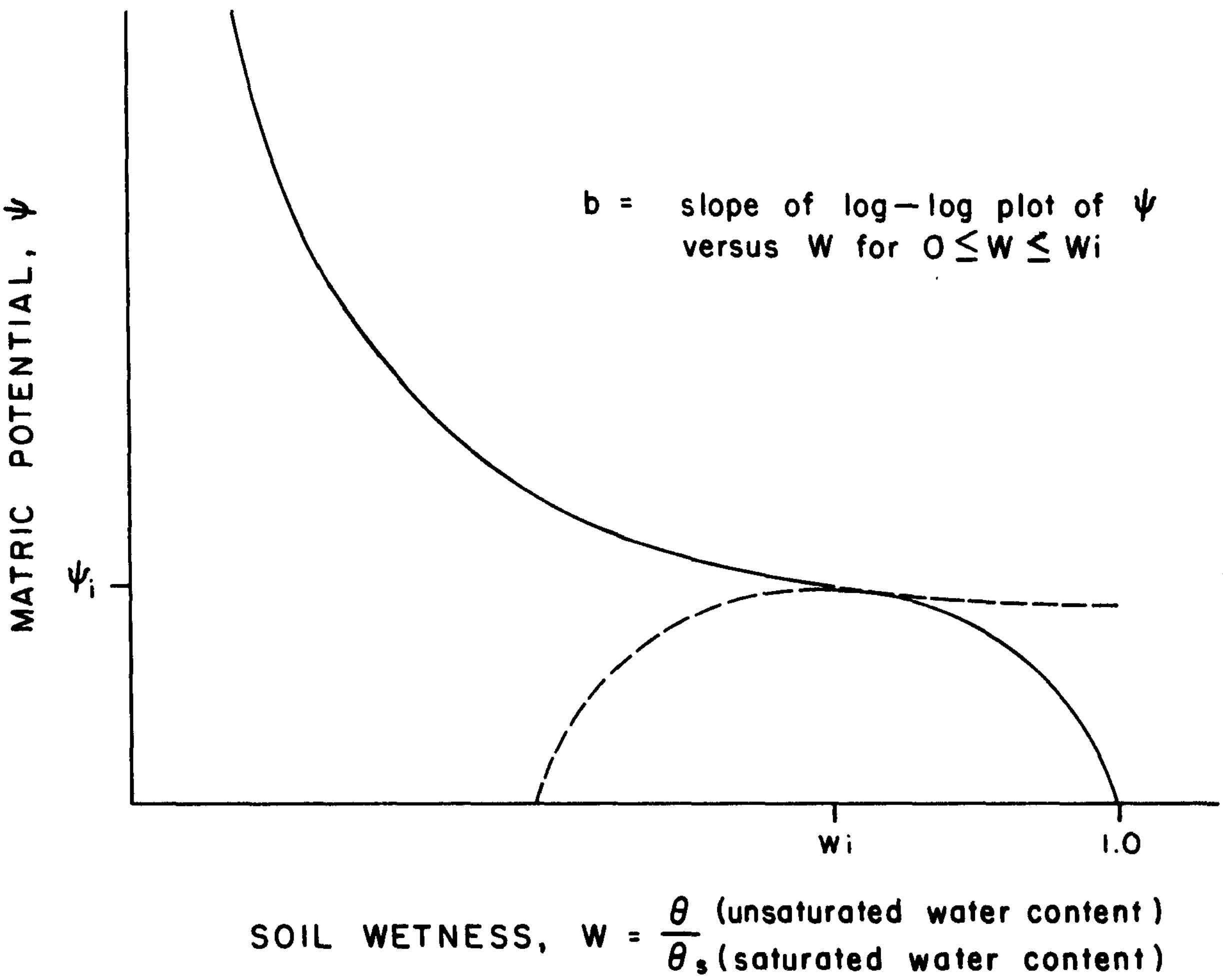

Figure 3-3. Qualitative illustration of moisture characteristic curve (the broken line segments are disregarded).

observing the point where $d\psi/dW$ changes from an increasing to a decreasing function as W decreases. It should be noted that equation 3-2 represents the W versus ψ curve where $W_i \leq W \leq 1$ only if $m > 0$, which requires that $W_i > b/b(b+1)$. Once the inflection point has been located, the values ψ_i, W_i, and b are used to calculate m and n using equations 3-3 and 3-4. In turn, ψ_s, b, m, and n are inserted into equations 3-1 and 3-2 to obtain the two part function described by Clapp and Hornberger (1978) relating soil moisture and matric potential.

Rearranging equation 3-2 using the quadradic formula gives

$$W = \frac{-m(n+1) - \sqrt{[m(n+1)]^2 - 4(-m)(-mn-\psi)}}{2(-m)} \qquad (3\text{-}5)$$

which is a convenient arrangement for solving W in the FORTRAN computer code, given m, n, and ψ are known.

The power curve function

$$k = W^{2b+3} \qquad (3\text{-}6)$$

is used in the SOILINER model to relate soil moisture, θ ($W = \theta/\theta_s$), and the unsaturated hydraulic conductivity, K, where

$$k = K/K_s$$

in which k is the relative hydraulic conductivity, K_s is the saturated hydraulic conductivity (SATK), and b is the same value as the parameter b in equation 3-1. To formulate this function for a specific soil, K_s must be determined.

Measurements of a saturated hydraulic conductivity can be made in the laboratory using falling or constant head permeameters, or in the field by means of a bail or slug test. In using permeameters, a sample of the soil is subjected to water under a known or falling head, and the flow through the sample in a known time is measured. Such tests have limited value due to the difficulty of placing samples representative of their natural state in the permeameter. Also, flow in solution cavities cannot be duplicated in a permeameter.

Bail or slug tests are carried out in the field using a single piezometer. Both tests are conducted by causing an instantaneous change in the water level within the piezometer. This change is achieved by either adding a known volume of water into the piezometer (slug test) or, removal of a known volume of water from the piezometer (bail test). The recovery of the water level is then observed.

A number of methods have been derived for determining the saturated hydraulic conductivity from the recovery data. The simplest interpretation of the recovery data is that of Hvorslev (1951). This method is for a point piezometer. For bail or slug tests run in piezometers that are open over the entire thickness of an aquifer, the methods of Cooper et al. (1967) and Papadopoulos et al. (1973) should be used. The main limitation of slug or bail tests is that the well point or screen should not be clogged or corroded. In addition, it may be difficult finding an in-situ, saturated soil that has the same characteristics as the liner soil being modeled.

Once the values of ψ_s, K_s, θ_s, b, W_i, and ψ_i have been determined, equations 3-1, 3-5, and 3-6 are inserted into subroutine SPROP in the SOILINER model (Appendix G). A soil type number is assigned to the particular soil so that it can be distinguished from other soil types which currently exist in the model.

3.4 Characteristic Curves Available in SOILINER

The SOILINER model uses functional expressions for determining unsaturated hydraulic conductivity, moisture content, and specific moisture capacity (which is the derivative of moisture content with respect to matric potential). For each element within the soil column, the soil type code (1, 2, 3, etc.) must be selected and used as input in the Soil Properties File. This allows stratification of different soil types within a column (i.e., clay, silt, sand, clay). The FORTRAN listing shows that 13 soil types are used, but the number of soil types is limited only by the number of functional relationships inserted into the subroutine SPROP. Table 3-1 shows the soil types used in the model and their associated values for saturated hydraulic conductivity, saturated moisture content (equal to the porosity in this study), and saturated matric potential. Figures 3-4a and 3-4b show the soil moisture characteristic curves for soil types 1 through 11 listed in Table 3-1.

The soil types numbered 1 through 11, and their associated empirical equations in Appendix G, are from Clapp and Hornberger (1978). Of these soils, the silty clay has the lowest saturated conductivity value of 1×10^{-4} cm/s. Typically, a clay liner consists of clay with a saturated hydraulic conductivity of 1×10^{-7} cm/s (40 CFR Part 264). Based on an extensive literature search, GCA was unable to locate any characteristic data representative of a clay with such a low saturated conductivity. Therefore, a hypothetical clay soil was derived by using the Yolo light clay utilized by Haverkamp et al. (1977) with a modified saturated conductivity of 1×10^{-7} cm/s (soil type 12). The empirical equations derived by Haverkamp et al. (1977) to relate matric potential with soil moisture and unsaturated hydraulic conductivity are also used in the SOILINER model (Appendix G) to represent this hypothetical clay. Finally, soil type 0 represents a high conductivity sand also developed by Havercamp et. al. (1977).

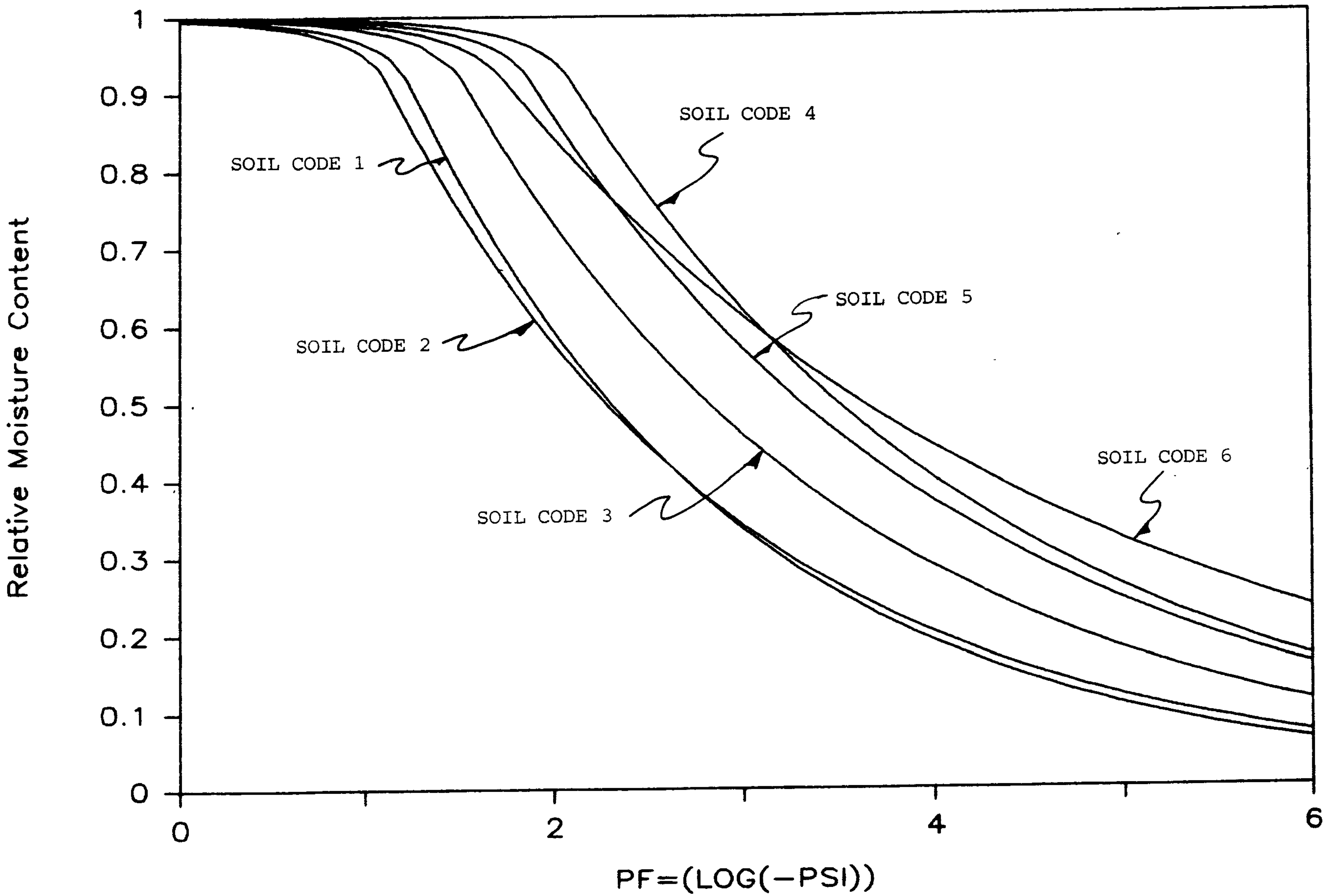

Figure 3-4a. Characteristic moisture curves for soil types ranging from sand to sandy clay loam.

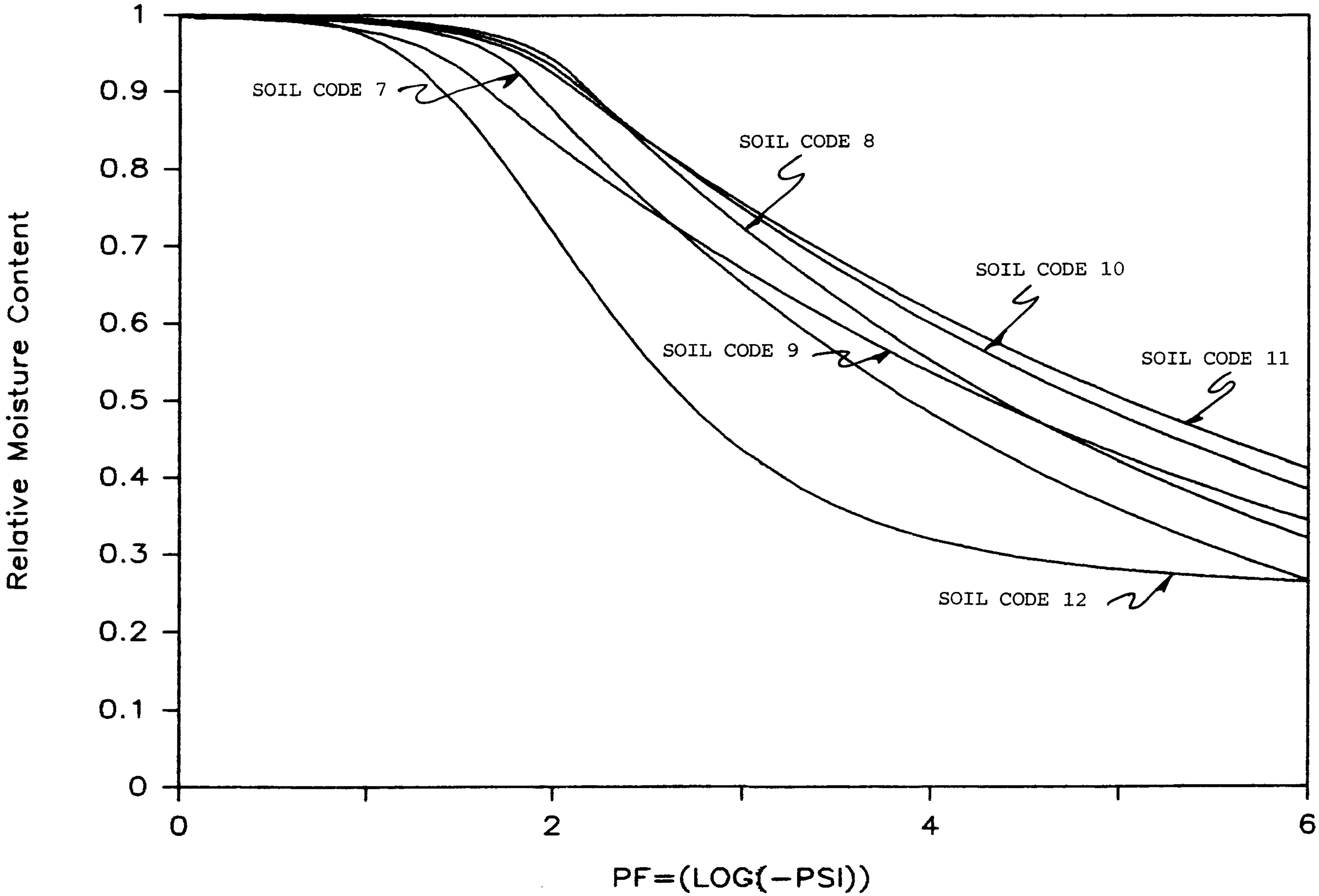

Figure 3-4b. Characteristic moisture curves for soil types ranging from silty clay loam to a compacted clay.

TABLE 3-1. REPRESENTATIVE VALUES OF HYDRAULIC PARAMETERS FOR THE SOIL TEXTURE CLASSES CODED INTO SOILINER

Soil code number	Soil texture	Mean clay fraction	SATK (K_s) (cm/sec)	POR (θ_s)	PSICRT (ψ_i) (cm/sec)
0	Sand	--	9.44×10^{-3}	0.287	-1.00
1	Sand	0.03	1.760×10^{-2}	0.395	-16.96
2	Loamy sand	0.06	1.563×10^{-2}	0.410	-12.97
3	Sandy loam	0.09	3.466×10^{-3}	0.435	-32.08
4	Silt loam	0.14	7.200×10^{-4}	0.485	-122.28
5	Loam	0.19	6.950×10^{-4}	0.451	-74.92
6	Sandy clay loam	0.28	6.300×10^{-4}	0.420	-54.12
7	Silty clay loam	0.34	1.700×10^{-4}	0.477	-67.94
8	Clay loam	0.34	2.450×10^{-4}	0.476	-128.19
9	Sandy clay	0.43	2.166×10^{-4}	0.426	-36.42
10	Silty clay	0.49	1.033×10^{-4}	0.492	-116.63
11	Clay	0.63	1.283×10^{-4}	0.482	-104.78
12	Heavy clay**		1.000×10^{-7}	0.495	-1.00

* From Li et al. (1976)

** Hypothetical clay liner soil from Havercamp et. al. (1977)

4. PROCEDURE FOR APPLYING SOILINER

Upon completion of the initial data gathering phase, it should be possible to synthesize a conceptual model of the liner. A conceptual model is an abstract description of the flow domain including the: (1) liner configuration, (2) nature and position of the boundary conditions, and (3) initial pressure distribution. Once the conceptual model has been established for a desired simulation, SOILINER can be applied.

4.1 Model Set-up

Following is a brief, step-by-step procedure in logical sequence to establish a SOILINER simulation:

1. Locate boundaries of the flow domain based on the liner thickness and depth to the water table. Boundaries must be positioned such that values of ψ (H and PSIBOT) can be specified.

2. Design a finite difference grid (discretization). Appendix D - Grid File provides an example discretization scheme.

3. Choose an appropriate characteristic curve for each soil type and assign the corresponding saturated properties (see Section 3.3) to each element of the flow domain as shown in Appendix D - Properties File.

4. Construct the initial pressure distribution based on the design specifications of the liner material during installation (i.e., optimum moisture content and the corresponding matric potential) and the existing field conditions for the native soil as described in Appendix D - Initial Conditions File. Typically, the moisture profile in the native soil is assumed to be in static equilibrium above a fixed water table which is deemed to represent an average depth below the liner base (see Section 2, Part I).

5. Develop a solution strategy as shown in Appendix D - Control File.

Steps 1, 3, and 4 are fairly straightforward and have been discussed in detail previously. More attention will be given to methods of discretization (Section 4.2) and the development of solution strategies (Sections 4.3 and 4.4). The discussions on steady state and transient solution strategies are based on an in-depth sensitivity analysis of parameters within SOILINER.

4.2 Designing a Finite Difference Grid

Once the dimensions of a conceptual model have been established, it is necessary to discretize the flow domain. Discretization results in a finite number of node points which constitute a one-dimensional grid. The governing equation employed by SOILINER is solved in terms of these node points (see Section 3, Part I). Thus model accuracy is influenced by the discretization scheme, and the following must be considered during grid development: (1) locations of abrupt changes in the initial PSI distribution, usually occurring at the liner surface, (2) sharp discontinuities in soil properties, and (3) model efficiency - although a large number of nodes will generally increase simulation accuracy, computer storage requirements and associated run times will increase.

The simplest grid divides the liner and site soil into equally sized subdomains. These elements may be fairly large if there are only small variations of moisture content in the profile, such as occurs under conditions of equilibrium. However, if large gradients of moisture (e.g., a wetting front) occur, the size of the elements should be smaller. In order to economize on the total number of elements, it is important in such cases to use a variably-sized set of elements in order to concentrate nodes in the region of greatest variations, as discussed below.

For the transient infiltration problem, the initial distribution of matric potential varies gradually everywhere, except at the top of the liner. As infiltration proceeds, this sharp front moves into the soil and is gradually smoothed as it moves down the soil column. By the time the moisture front reaches the bottom of the liner, it is sufficiently dispersed to relax the subdomain size requirement. This variation suggests a grid with small subdomains at the top of the liner and a gradation of subdomain size moving down the column. Figure 4-1 shows two graded grids of this general design. In the first, the grid consists of blocks of segments which become larger in steps. The second sample has subdomains which vary in size continuously from the liner top to the water table. In this second grid, the bottom subdomain is about 10 times as large as the top one.

It is difficult to specify a general criterion for the size of grid spacing. The subdomain size and gradation are usually determined during the initial simulations. Many models are very sensitive to this discretization and care should be taken. A good test of the effect of subdomain size on the solution is to compare the change in simulated matric potential obtained using smaller and smaller grid spacings. When the change between two simulations is unaffected by grid spacing, the larger size can probably be used.

4.3 Steady State Solution Strategy

Regardless of the solution strategy chosen, SOILINER will always calculate the steady state PSI distribution. For transient simulations, the steady state solution serves as a means of determining the time to steady state during an infiltration event. However, it is possible to request only the steady state solution (by setting LSTEDY = T) for the purpose of first developing this strategy. This feature avoids lengthy run times associated with transient simulations before properly developing a steady state solution.

In comparison with the transient formulation, there are relatively few parameters in the Control File which are required for a steady state simulation - SRPARM, MAXIT, and ERRMAX. Of these three, only the successive relaxation parameter (SRPARM) is specific to the steady state algorithm.

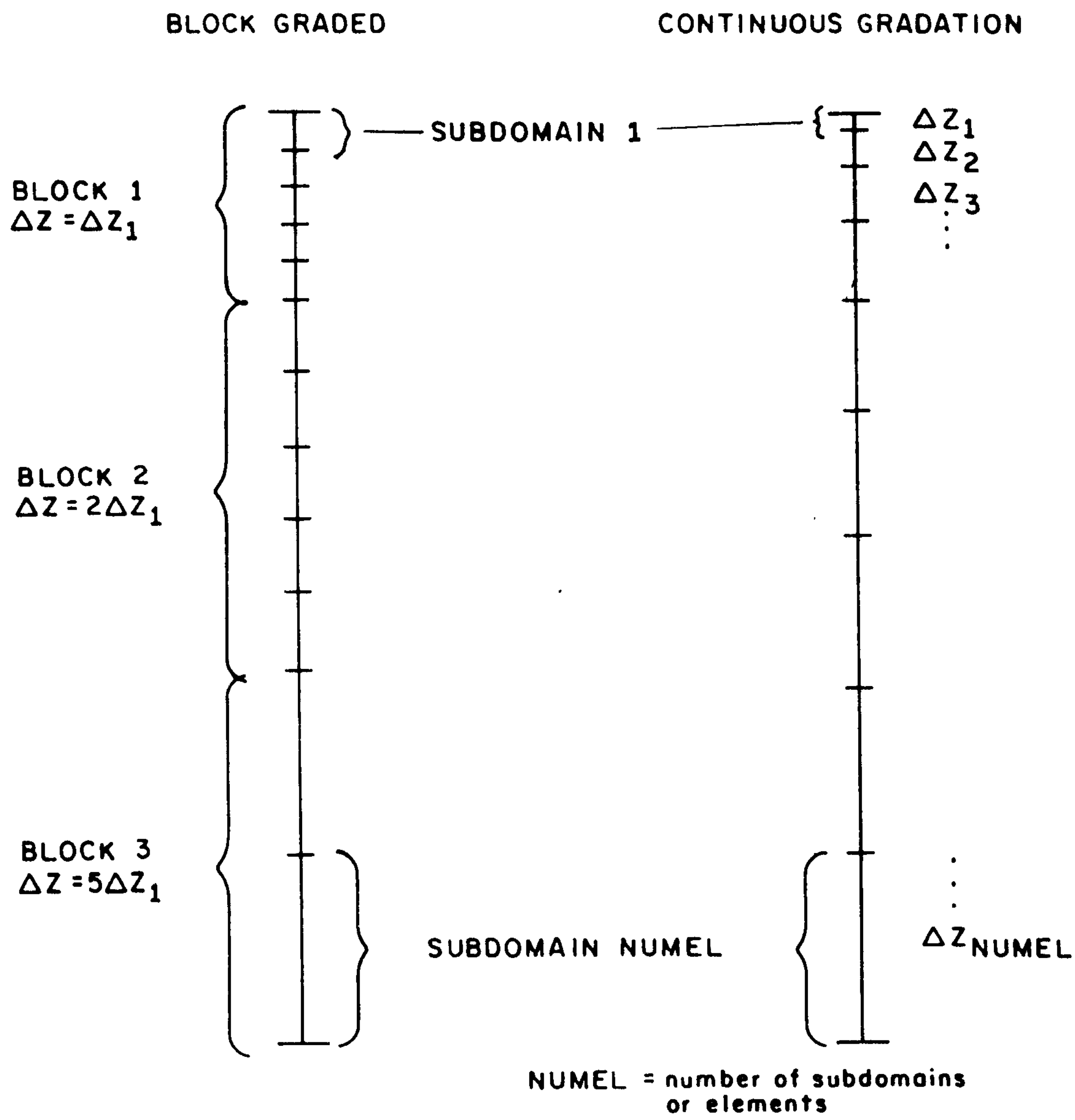

Figure 4-1. Two vertical grids with variable subdomain sizes.

As discussed in Part I, the steady state solution is achieved iteratively. For each iteration the calculated DSPI at each node is forced to satisfy the specified boundary conditions based on the current soil properties. The pressure distribution is then updated as follows:

$$PSI = PSI + (SRPARM * DPSI) \tag{4-1}$$

Soil properties are adjusted to reflect the new PSI distribution and DPSI is again calculated. The iterative procedure continues until the change in soil properties no longer significantly affects the calculated PSI distribution (i.e., DPSI $\leq$ ERRMAX).

For layered systems, values of SRPARM $\geq$ 1.0 lead to model instability. Figure 4-2 depicts the affect of SRPARM = 1.0 on the change in PSI at nodes 25, 55, and 85 from a 500 cm liner configuration (180 cm clay, 320 cm sand). Without relaxation (SRPARM = 1.0), the magnitude of the calculated DPSI between iterations is far greater than a reasonable ERRMAX at each node. Similar oscillations occur at all other nodes. To reduce numerical overshoot and the propagation of subsequent oscillation, it is necessary to specify a value of SRPARM that is less than 1.0.

When SRPARM < 1.0, only a percentage of the calculated DPSI is used to resolve PSI during the iterative procedure. Thus, the magnitude of initial overshoot is reduced and subsequent oscillations may be sufficiently damped to allow convergence. Figure 4-3 reveals the affect of SRPARM < 1.0 on convergence at node 85. As shown, SRPARM = 0.80 was not sufficient to achieve convergence. However, values of 0.6, 0.4, and 0.2, all converged on the same steady state PSI distribution. Termination of the iterative procedure at node 85 for these three values indicates that all nodes in the flow domain also met the error criterion.

Although values for SRPARM of 0.2, 0.4, and 0.6, all lead to convergence, the number of iterations did vary (27, 16, and 24, respectively). Tables 4-1 and 4-2 provide a comparison on the efficiency of various SRPARM scenarios for a 90 cm and 180 cm clay liner configuration, respectively. For both scenarios, MAXIT was set at 100 and ERRMAX was established at 0.01 cm. Results indicate that: (1) SRPARM generally works best at values close to 0.5, (2) decreasing head values require more iterations, (3) increasing MAXIT for values of SRPARM > 0.5 does not guarantee convergence (these scenarios frequently diverge if SRPARM > 0.5), and (4) as the number of nodes increases, more iterations are required before ERRMAX is satisfied simultaneously at all nodes. In general, for all cases where SRPARM $\leq$ 0.5, the number of iterations required for convergence decreases dramatically as the ERRMAX criterion is relaxed to values greater than 0.01 cm.

4.4 Transient Solution Strategy

The first step in establishing a transient solution strategy is to estimate the time of leachate breakthrough using any of the available analytical techniques. Three techniques reviewed in the Technical Resource Document (EPA/530-SW-84-001) include the: (1) transit time equation, (2) the

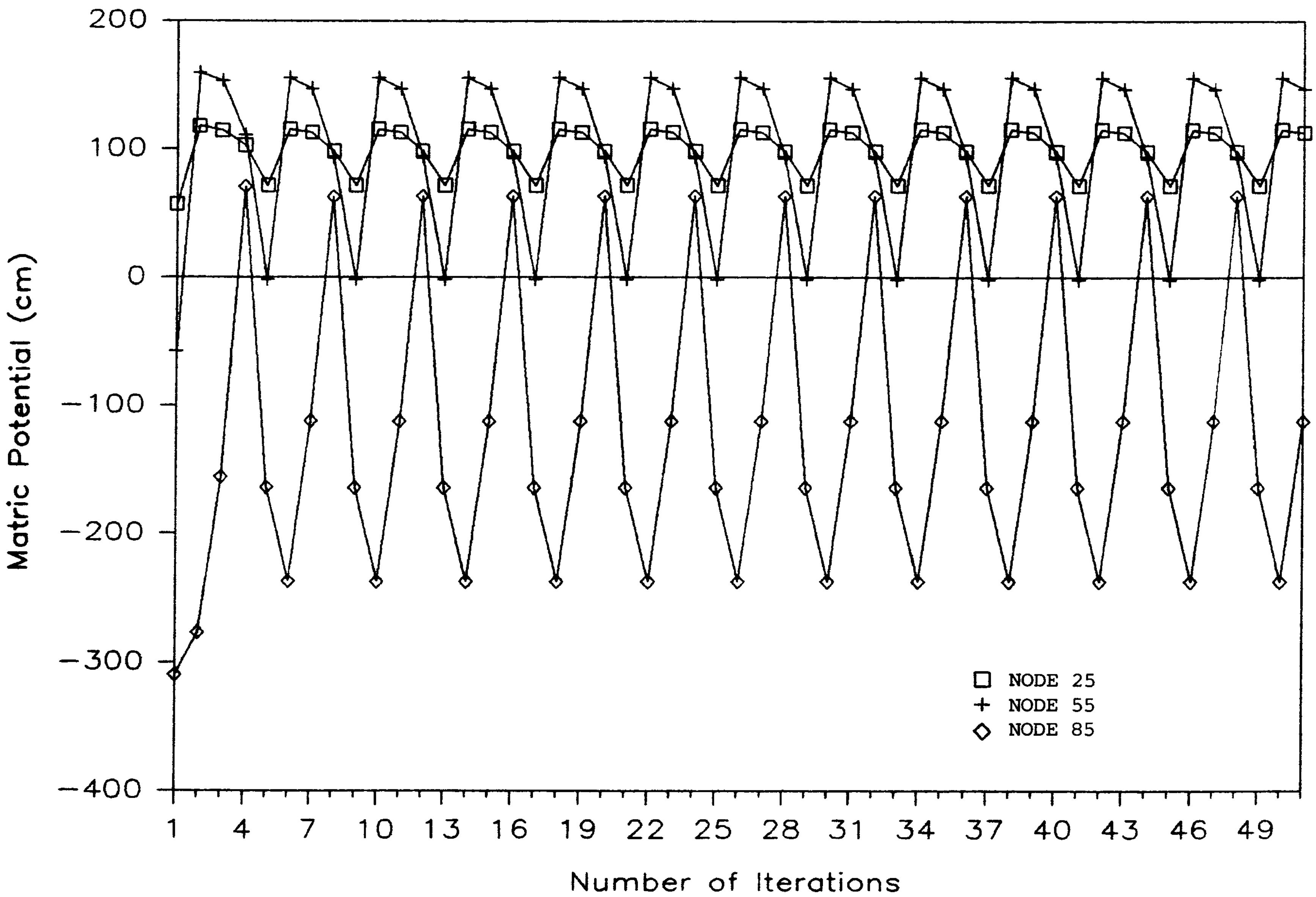

Figure 4-2. Steady state algorithm leads to numercial oscillation as shown at three selected nodes from a clay/sand liner configuration (SRPARM = 1.0).

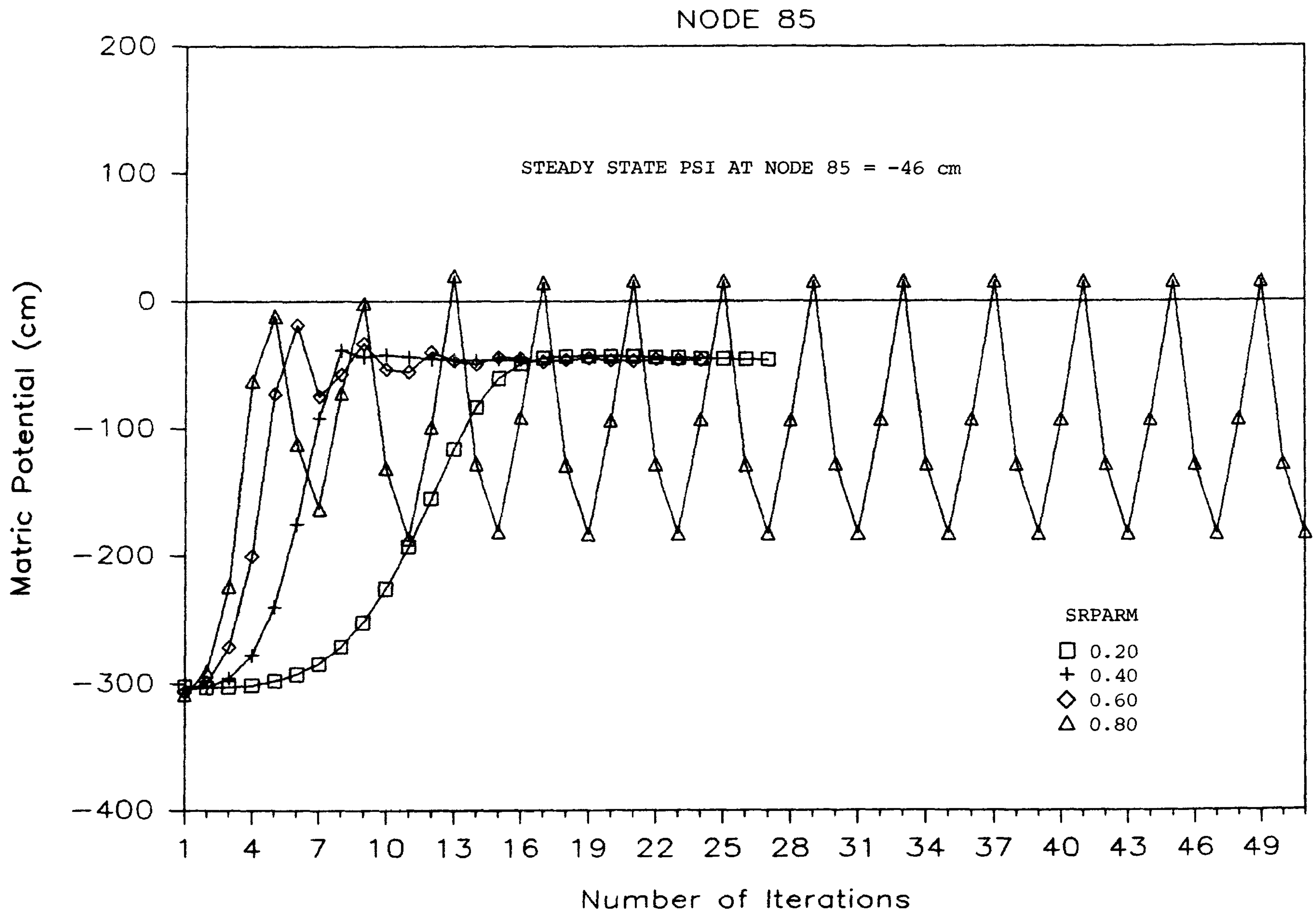

Figure 4-3. Impact of SRPARM on convergence during steady state algorithm at node 85.

TABLE 4-1. EFFECT OF SRPARM ON THE NUMBER OF ITERATIONS TO CONVERGENCE (90 cm LINER)[a]

HEAD (cm)	SRPARM 0.2	0.4	0.5	0.6	0.8
100.00	40	21	17	20	101[b]
50.00	42	26	26	32	101[b]
30.50	47	32	35	52	101[b]
2.54	77	101[b]	101[b]	101[b]	101[b]

[a] ERRMAX = 0.01 cm, MAXIT = 100
[b] MAXIT exceeded before convergence

TABLE 4-2. EFFECT OF SRPARM ON THE NUMBER OF ITERATIONS TO CONVERGENCE (180 cm LINER)[a]

HEAD (cm)	SRPARM 0.2	0.4	0.5	0.6	0.8
100.00	44	23	25	43	101[b]
50.00	43	34	42	101[b]	101[b]
30.50	50	45	77	101[b]	101[b]
2.54	83	101[b]	101[b]	101[b]	101[b]

[a] ERRMAX = 0.01 cm, MAXIT = 100
[b] MAXIT exceeded before convergence

Green-Ampt wetting front model, and (3) the transient, linearized solution. A first approximation of breakthrough serves as an aid in determining the total duration (ENDTIM) of a transient simulation. Once an ENDTIM is established, special output times can be selected to evaluate varying positions of the wetting front with time. In many cases, however, steady state is achieved before the specified ENDTIM. SOILINER terminates the time stepping procedure and only tracks the particle based on the steady state PSI distribution. Consequently, the last printed output time occurs at steady state, in which case special output times greater than the time at which steady state is achieved, and ENDTIM will not be printed. The remaining time stepping parameters will be discussed after a brief description of the infiltration event.

During the infiltration event, matric potential, PSI, conductivity, $K(\psi)$, and moisture content, $\theta(\psi)$, are continuously changing until steady state is achieved. This simultaneous change in soil properties as a function of PSI effects the way in which PSI is calculated over time. To handle this nonlinearity, SOILINER employs an iterative approach based on the finite difference method. First, the new PSI distribution is predicted at a calculated time level t_1 based on the previous PSI distribution and associated soil properties. The soil properties are then updated at t_1 to reflect the newly calculated PSI distribution. Based on the new soil properties, SOILINER attempts to adjust each nodal value of pressure, PSI, by a calculated change in pressure, DPSI.

Each time SOILINER updates the soil properties and readjusts PSI by the value DSPI (see 3-15, Part I), an iteration is completed at the time level t_1. SOILINER continues to iterate until the maximum DPSI at all nodes becomes less than the user-specified, error for convergence, ERRMAX. To prevent uncontrolled iteration for a solution which is not converging at t_1, a maximum number of iterations per time level (MAXIT) is also specified. If in attempting to reduce DPSI to a value smaller than ERRMAX, MAXIT is exceeded, a forced exit from the time level occurs and a warning is issued. In either case, whether ERRMAX is met or MAXIT is exceeded, the iterative procedure terminates and a new time step is determined.

In order to step forward in time, SOILINER requires input data which is used to calculate the size of each time step, DT. One of these input parameters is DT itself, which describes the initial time step increment. After the first time step using the initial, user-specified DT, SOILINER calculates each subsequent time step as:

$$DT = \frac{DT * CHPARM}{CHMAX} \qquad (4-2)$$

where CHPARM is a time step change parameter, input by the modeler, and CHMAX represents the maximum change in matric potential at any node within the solution domain, and is calculated between successive time steps in the program. Therefore, after the first time step, SOILINER forces a small time step if CHMAX is large, and a large time step if CHMAX is small. This algorithm is a characteristic of the model designed to provide an internal method of assuring model efficiency during the initial phase of infiltration. As time progresses towards steady state, changes in the PSI distribution between successive time steps is generally small. As a result, calculated time steps may get excessively large. To prevent this occurrence, SOILINER incorporates another temporal input parameter (DTMAX), which regulates the maximum size of a calculated time step. If the calculated time step is greater than DTMAX, SOILINER forces DT to equal DTMAX. This parameter prevents the time steps from getting so large that the predictions of PSI become inaccurate, which may lead to difficulties in iteration convergence and unusually high run times.

SOILINER continues to step through time until steady state or the final simulation time step (ENDTIM) is reached. Each new time step is generated as DPSI falls below ERRMAX, or MAXIT is exceeded. However, as another method of avoiding uncontrolled CPU time, SOILINER requires that the user input a maximum number of time steps per simulation (MAXNT). Model execution will be terminated if MAXNT is exceeded.

The transient solution technique employed by SOILINER also incorporates a temporal weighting parameter, ALPHA, which is used to weigh the approximations of the state variable, PSI, at a point in time, t^{n+1}, between the known values of PSI at t^n, and the unknown values of PSI at t^{n+1}. ALPHA can vary between 0.0 and 1.0. When ALPHA = 0.0, equation 3.19 (Part I) is solved explicitly based on the values of PSI at t^n, which are known. When ALPHA = 1.0, equation 3.19 (Part I) is solved fully implicitly because the values of PSI at t^{n+1} are unknown. Values of ALPHA between 0.0 and 1.0 weight the approximations accordingly. A value of 0.5 for ALPHA assumes that the best approximation of PSI at t^{n+1} is based equally on the known values of PSI at t^n and the unknown values of PSI at t^{n+1}.

4.4.1 Method of Temporal Sensitivity Analysis--

Several transient simulations were conducted on an IBM 3033 mainframe in which DT, DTMAX, CHPARM, and ALPHA were varied in order to evaluate SOILINER'S sensitivity to these parameters. All remaining input data were held constant including the initial and boundary conditions, soil properties, and grid discretization. Each transient simulation was conducted such that the final output time (ENDTIM) was specified to assure that the simulation reached steady state conditions. Then a steady state solution was obtained for these same conditions for comparison with the transient simulations to determine how variations in DT, DTMAX, CHPARM, and ALPHA affected model stability, accuracy and efficiency.

Each run's stability, or ability to converge, was evaluated based on a review of the output. The accuracy of each simulation was determined by quantitative comparisons of the matric potential at ENDTIM with the steady state solution. Good comparisons were considered as those which showed no greater than 1.0 cm of difference in matric potential between any two corresponding nodes. Although some simulations were allowed extended CPU times to observes the results, generally one minute was arbitrarily chosen as the maximum amount of CPU time allocated for a given simulation. Results of the sensitivity analyses substantiate that one minute is an economic and realistic amount of time when compared with the values of ERRMAX and MAXIT used for these analyses. The efficiency of each simulation was evaluated based primarily on the total number of iterations (NKITER) for a given simulation. If job summaries are not available to obtain CPU times, the variable NKITER can be utilized as an indicator of CPU time.

The scenario employed for all the sensitivity analyses is outlined below and illustrated in Figure 2-1 of Part I. This scenario was used to minimize the total CPU time required for the large number of simulations performed in this sensitivity analysis.

1) Total solution domain, z_w (Figure 2-1, Part I), of -361.00 cm, corresponding to 76 nodes (or 75 elements).

2) Clay soil liner depth, d, of 61.00 cm, with a hydraulic conductivity of 1.0×10^{-7} cm/sec, porosity of 0.4950, and 56 nodes of variable grid size.

3) Natural sand fill of depth 300.00 cm, with a hydraulic conductivity of 9.444×10^{-3} cm/sec, porosity of 0.2870, and 20 nodes of constant 15.0 cm grid size.

4) A constant pressure boundary condition of 100.00 cm of impounded liquid, h_i, above the clay liner.

5) A constant pressure boundary condition of 0.0 cm for the water table at the base of the sand fill (-361.00 cm).

6) Constant initial matric potential in the clay liner of -300.00 cm and variable initial matric potential in the sand fill to establish static equilibrium.

7) Temporal parameters of DTMAX = 8.64×10^5 sec, MAXNT = 2000, MAXIT = 50, and ERRMAX = 0.01 cm.

These values will provide some direction for users, however, the modeler should recognize that each scenario will probably require at least some variation from the above in order to meet specific requirements for convergence, accuracy, and efficiency.

4.4.2 Choosing a Time Step--

The considerations in selecting time step sizes are analogous to those in grid design. One efficient numerical scheme uses very small time steps at the beginning of infiltration, when the matric potential gradients are highest, and gradually increases the time step size as the moisture front advances into the soil liner and disperses. Again, the initial time step size and the rate of gradation are usually determined during initial simulations of the problem at hand. Given that initial time steps may be as low as one second, or less, a variable time step size is imperative to maintain cost effectiveness during long-term simulation.

Based on the algorithm used to calculate DT (equation 4-2), it is clear that, after the first time step, DT can be regulated to an extent by CHPARM. However, there are several additional input parameters of equal importance which need to be considered in selecting a value of DT. These parameters include: DTMAX, ERRMAX, MAXIT, and MAXNT. Various combinations of these parameters will lead to different results for a given scenario, with varying levels of accuracy and efficiency. In some instances, the model may not converge to a solution. There is no unique combination of these parameters for a given problem which will yield the best set of results. Rather, the modeler must determine what level of accuracy is required and weigh this against the cost (in CPU time) of obtaining those results.

Based on the scenario used for the sensitivity analyses, SOILINER appeared very sensitive to values of CHPARM and DTMAX, and rather insensitive to values of DT ranging from approximately 10 to 500. The most efficient and accurate results obtained (for the given scenario) during the sensitivity analyses occurred with $50 \leq$ CHPARM ≤ 100 and DT within the range stated above. However, it should be noted that high values for either DT or CHPARM should be moderated by relatively low values for the other. As can be seen in Figure 4-4, SOILINER was not sensitive to the initial DT when an appropriate value of CHPARM was utilized. Changes in DT of more than an order of magnitude do not appear to affect CPU time significantly (Table 4-3). However, as DT was increased above 100 for the simulations shown in Table 4-3, MAXIT warnings occurred and the solution became progressively less accurate. In addition, values of DT that were excessively large when combined with high values of CHPARM caused inaccurate predictions of PSI for the first time step. The user is cautioned against setting excessively high DT values since the results from each time step are used as the basis for future predictions. If a large initial time step is used, it should be noted that model convergence does not necessarily guarantee accuracy. Furthermore, convergence may not be achieved, as depicted in Figure 4-5.

The user is also cautioned against small values of DT based on the following logic. Results indicated that if the calculated time step was forced to remain small using CHPARM, the number of iterations required for convergence per time step was low. Since there was little change in the predicted value of PSI and the soil properties at each new time step, DPSI reached a value smaller than ERRMAX in relatively few iterations. However, MAXNT was set at a large value because the simulation required many time steps to reach even the first specified output time. If the specified simulation time was very large relative to the calculated values of DT, the CPU time, and therefore the cost of the simulation was high (see Table 4-4).

If the simulation progress through time normally with no execution warnings (i.e., MAXIT exceeded), DT will generally increase as steady state is approached. Successively larger DT values arise as the change in PSI between time steps decreases (See equation 4-2). Even though changes in PSI can get relatively small, it has been shown that DT can become excessive, leading to model instability. To prevent unusually large, calculated DT values, the variable DTMAX is specified. If DT exceeds DTMAX, DT is set equal to DTMAX within SOILINER. However, it is still possible to specify a DTMAX that is too large.

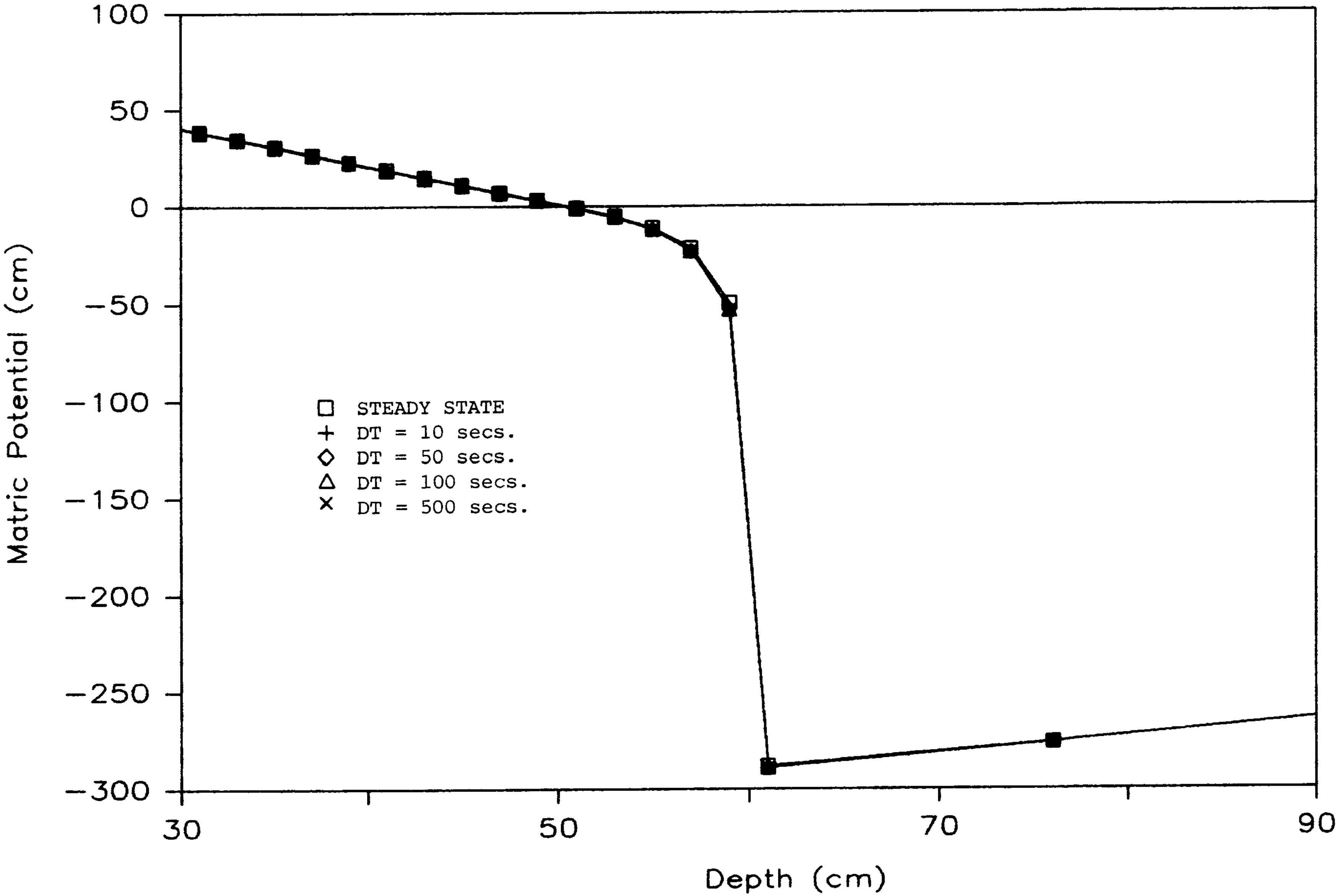

Figure 4-4. SOILINER sensitivity to initial DT values (pressure distribution at ENDTIM of transient simulation where CHPARM = 75 and ALPHA = 1.0).

TABLE 4-3. CPU TIME AS A FUNCTION OF DT FOR CONSTANT CHPARM=100*

DT	CPU Time/Simulation (Sec)
10	24.09
100	23.73
1,000	23.74
10,000	20.29
86,400	18.65

* All remaining input parameters constant for each simulation

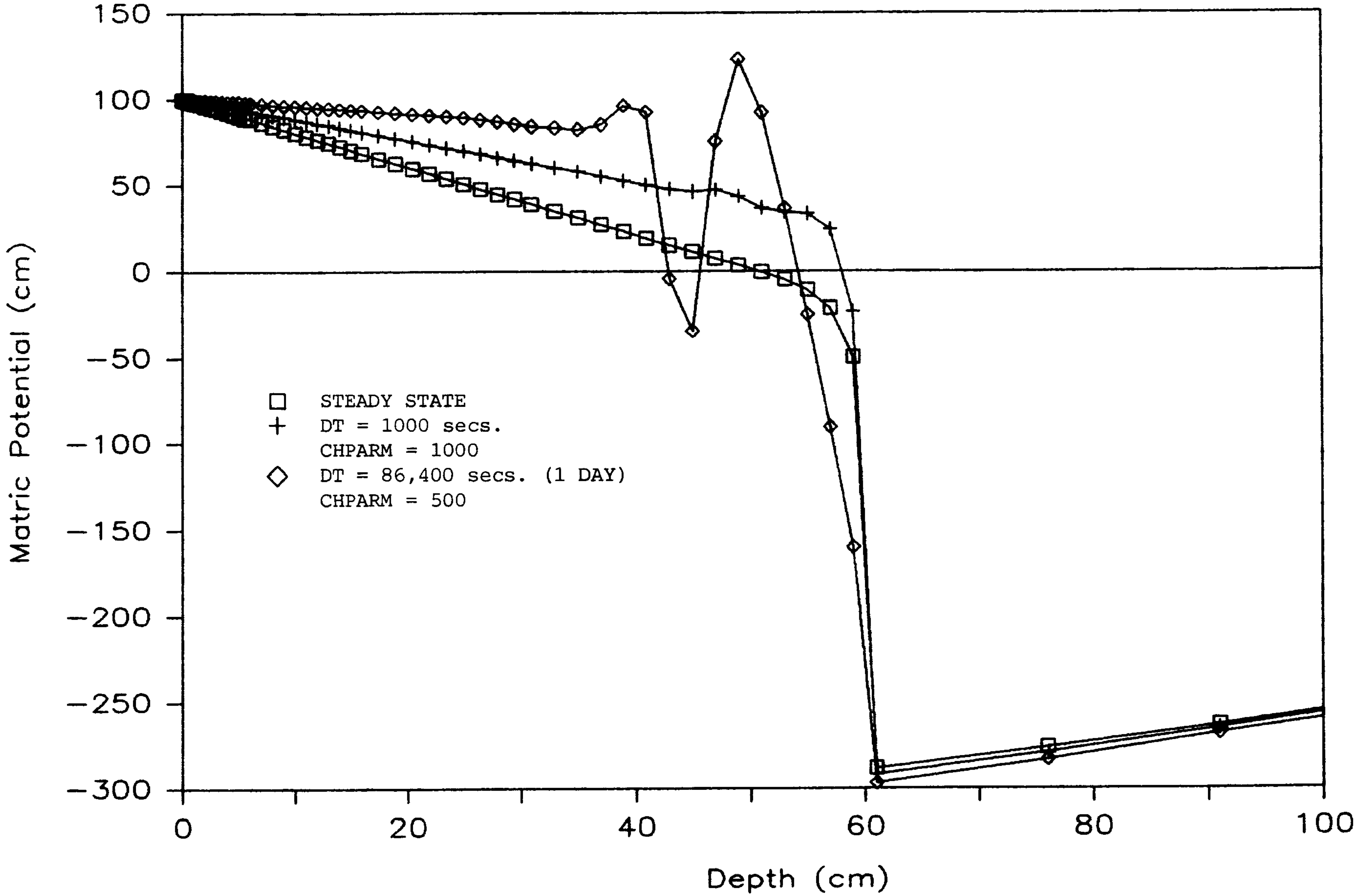

Figure 4-5. Divergence associated with large DT and CHPARM values.

TABLE 4-4. SAMPLE CPU TIME NECESSARY FOR SIMULATIONS WITH SMALL DT AND CHPARM VALUES*

ALPHA	DT	CHPARM	CPU TIME/SIMULATION (SEC)
0.75	10	10	62.28
1.00	10	10	65.43
1.00	10	1.0	53.38[a]

* All remaining input parameters constant for each simulation

a Forced exit at this time due to MAXNT exceeded. This run terminated at simulation time 4.5 hours (ENDTIM was set to 1 year).

Figure 4-6 depicts changes in PSI distributions overtime for a two-foot liner. Note that special output time 5 (2 years) represents a time step in which MAXIT was also exceeded (overall, MAXIT was exceeded 8 times). This problem was easily rectified by reducing DTMAX from 100 to 10 days, whereby all MAXIT warnings were eliminated. Although the number of time steps increased from 164 to 232, NKITER decreased from 1860 to 1634, a reflection of improved model efficiency.

4.4.3 Choosing a Temporal Weighting Parameter--

The explicit solution (ALPHA = 0.0) for PSI at a future time step is based on the known values of PSI at the present time step. This solution technique should only be used with small time steps, which reduces model efficiency. Also, even with small time steps the explicit solution is conditionally stable (i.e. convergence may not occur). Conversely, the implicit solution (ALPHA = 1.0) is unconditionally stable (i.e. it always converges), but since the solution for PSI at a future time step is based on the unknown derivatives of PSI at that future time, the accuracy of the results is still a function of the time step size, although not as sensitive as the explicit solution.

In order to evaluate the effect various values of ALPHA had on the solution to a given scenario, several simulations were conducted in which ALPHA was set at 0.0, 0.25, 0.50, 0.75, and 1.00. For each value of ALPHA, the temporal parameters were varied in an attempt to achieve model convergence within 59 seconds of CPU time while noting accuracy of the results.

Several simulations were conducted for values of ALPHA = 0.75 and 1.00. It was found that these simulations were unconditionally stable, however, the user is cautioned to review the output carefully because accuracy was dependent on the time stepping parameters. For both 0.75 and 1.00, each simulation was more easily adjusted (than was possible for lower values of ALPHA) so that no maximum iteration warnings were issued. These adjustments were made primarily with CHPARM, DT, and DTMAX. As indicated by Table 4-5, it appears that transient solutions are most accurately and efficiently simulated when the fully implicit solution (ALPHA = 1.00) is employed. It should be noted that for each run shown with MAXIT warnings in Table 4-5, the final output time (ENDTIM) was in good comparison with the steady state solution. Thus, MAXIT warnings do not necessarily indicate poor results, although an ideal simulation would not give any. The modeler is advised that as ALPHA is increased, the solution technique increasingly estimates the state variable, PSI, at time level t^{n+1} based on the predicted derivatives of PSI at that

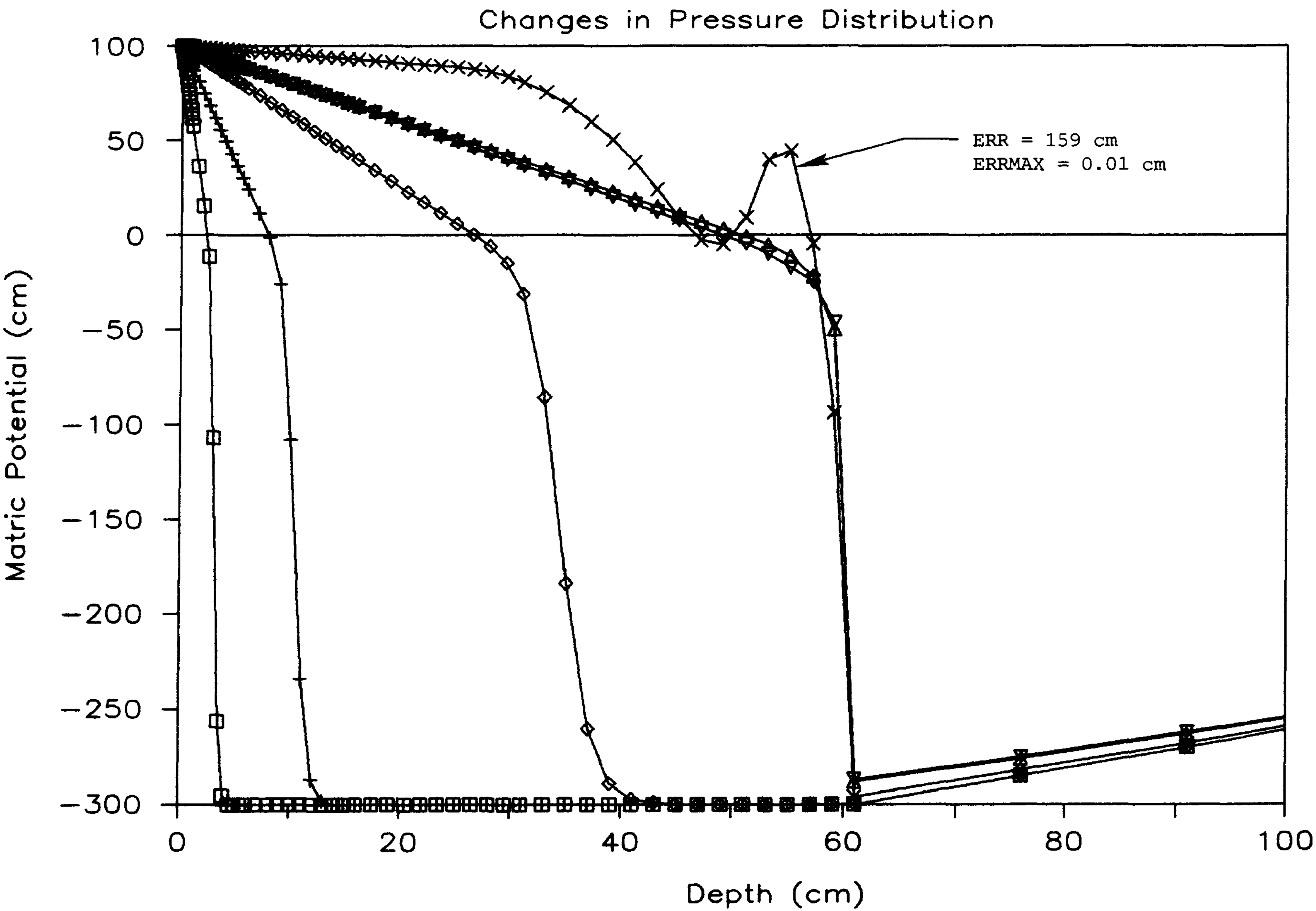

Figure 4-6. Results of MAXIT warning due to excessive DTMAX, where the magnitude of error between successive iterations (with respect to ERRMAX) can be determined from ERR.

TABLE 4-5. SOLUTION CHARACTERISTICS AS A FUNCTION OF TEMPORAL PARAMETERS AND MAXIT

ALPHA	DT	CHPARM	MAXIT	CPU TIME	NUMBER OF MAXIT EXCEEDED WARNINGS
0.75	10	10	50	62.28	2
0.75	10	75	50	24.90	1
0.75	10	50	50	26.82	0
0.75	10	100	50	23.56	1
0.75	1000	100	225	26.39	1
0.75	1000	100	175	25.33	1
0.75	1000	100	100	23.91	1
1.0	10	10	50	65.40	1
1.0	10	50	50	30.78	1
1.0	10	75	50	28.14	0
1.0	10	100	50	28.28	4
1.0	50	75	50	28.00	0
1.0	100	75	50	28.09	0
1.0	500	75	50	27.39	0

time. However, for a scenario other than the one used in this sensitivity analysis, it is possible that solution techniques which weigh the approximations of PSI somewhat more on the known values of PSI at time level t^n (e.g., ALPHA < 0.75) may be more efficient and accurate.

Model convergence with accurate results could not be obtained for any value of ALPHA $\leq$ 0.50. Problems encountered in these simulations were primarily register overflows or underflows, or insufficient CPU time (i.e., greater than 59 seconds - Table 4-6). Those simulations that did not converge showed highly inaccurate results. Some simulations revealed matric potentials within the liner that exceeded the boundary conditions. It appears that both the Crank-Nicolson and explicit methods of solution to the unsaturated, transient flow equation should not be used with SOILINER. This limitation exists because explicit solutions are conditionally stable and need to be used with small time steps. Furthermore, although Crank-Nicolson solutions are unconditionally stable, their accuracy is also a function of the time step size. Since SOILINER uses an internal method of determining each time step, the user cannot assure that each calculated DT value will remain within the range required for a particular solution technique. Explicit and Crank-Nicolson solutions typically showed convergence and/or inaccuracy problems and it is recommended that these solution strategies be avoided.

4.4.4 Setting a Tolerance for Iteration Convergence--

The CPU requirements should be borne in mind when the tolerance for iteration convergence, ERRMAX, is set. Values of ERRMAX that are very low will typically require high values for MAXIT and lengthy simulation times. In some cases, if ERRMAX is too low (i.e., less than 0.01 cm), convergence may never occur regardless of the specified MAXIT. Conversely, if ERRMAX is too high, run time will be minimized (by relaxing the requirement on iteration convergence) at the expense of model accuracy. Typical values for ERRMAX range from 1.0 to 0.01 cm.

Normally, the modeler should decide what level of accuracy is acceptable and set ERRMAX to this value. One should then attempt a simulation using a conservative value of MAXIT and adjust DT and CHPARM at this MAXIT value while noting CPU time and model accuracy. In this manner, the user can systematically identify the optimum temporal input parameters. It is the modeler's decision to increase MAXIT (and therefore the cost of a simulation) based on the results of prior simulations.

4.4.5 Choosing the Maximum Number of Iterations Per Time Step--

The maximum number of iterations per time step, MAXIT, is used as one means of limiting the amount of CPU time for a given scenario. Depending on the scenario and input parameters, the number of iterations required per time step for convergence may vary greatly. Under most conditions, MAXIT was found not to have exceeded 50. However, the modeler should review the output from a given scenario to determine the benefit of increasing MAXIT directly if convergence appeared to be occurring when a forced exit from the time step occurred. If the solution appeared to be diverging (as shown in Figure 4-6), other parameters must be changed.

TABLE 4-6. VALUES OF DT AND CHPARM PRODUCING REGISTER UNDERFLOWS/OVERFLOWS OR ABENDS AS A FUNCTION OF ALPHA

ALPHA	DT	CHPARM	COMMENT
0.0	0.1	0.1	A[1]
0.0	0.1	10.0	U/O[2]
0.0	0.1	100.0	U/O
0.0	10.0	10.0	U/O
0.0	100.0	100.0	U/O
0.0	1,000.0	1,000.0	U/O
0.0	5,000.0	5,000.0	U/O
0.25	0.1	0.1	A 2 min. CPU allocated
0.25	1.0	1.0	A 2 min. CPU allocated
0.25	10.0	10.0	A 2 min. CPU allocated
0.25	10.0	1,000.0	U/O 2 min. CPU allocated
0.50	0.1	1.0	U/O
0.50	0.1	50.0	U/O
0.50	0.1	1,000.0	A
0.50	1.0	1.0	A
0.50	1.0	50.0	U/O
0.50	10.0	1.0	U/O
0.50	10.0	10.0	U/O
0.50	10.0	1,000.0	A
0.50	100.0	50.0	U/O
0.50	1,000.0	50.0	U/O
0.50	10,000.0	50.0	U/O

[1]A Abend (i.e., Program terminated due to run-time errors or insufficient CPU time.)

[2]U/O Underflow or Overflow (i.e., Numbers were too small or large, respectively, for the computer to handle.)

All remaining input parameters constant for each simulation

All simulations allocated 50 CPU seconds unless otherwise noted.

In some cases, the impact from exceeding MAXIT may not be as obvious as that shown in Figure 4-6. For the last time step shown in Figure 4-6 the calculated ERR at MAXIT was 159.5 cm, far greater than the specified ERRMAX of 0.01 cm. In other cases, it is possible that the criterion established for convergence was barely missed (i.e., ERR = 0.02 cm with an ERRMAX set at 0.01 cm) when the MAXIT warning was issued. For this situation, the calculated PSI distribution associated with the MAXIT warnings may be reasonable. Since MAXIT warnings are frequently issued for a time step that does not coincide with a special output time, it is difficult to visualize the magnitude of the ERR associated with the warning. When MAXIT is exceeded, erroneous values for PSI and the associated soil properties are used as the basis for subsequent predictions through the forcing vector F, and stiffness matrix STIFF (see Part I and Appendix F). Consequently, an extremely poor prediction at a given time step (Figure 4-6) may be propagated through time even though subsequent time steps may converge with no MAXIT warnings. As an aid in determining the impact of MAXIT warnings on the overall simulation, SOILINER provides the calculated ERR each time a MAXIT warning is issued. Ideally, the parameters will be adjusted accordingly such that no warnings are issued for a desired simulation.

4.4.6 Sensitivity of the Particle Tracking Algorithm--

Since the particle tracking algorithm is based on the predicted PSI distribution at each time step, it is very sensitive to prediction accuracy. Figure 4-7 demonstrates, by comparison with Figure 2-5, the impact of model instability on particle movement. The results depicted in both graphics were obtained from the same simulation, with one exception. The variable, DTMAX, was changed from 10 days (Figure 2-5) to 100 days as shown by the increased spacing between data points towards the latter half of the simulation time in Figure 4-7. The results in this figure were derived from the predicted PSI distributions of Figure 4-6. For the time steps numbered in Figure 4-7, MAXIT was exceeded and erroneous PSI results were obtained and in some instances, caused the particle to move upward into the liner. By readjusting parameters (i.e., DTMAX reduced to 10 days) it was possible to eliminate all warning messages resulting in an accurate simulation of particle movement (Figure 2-5).

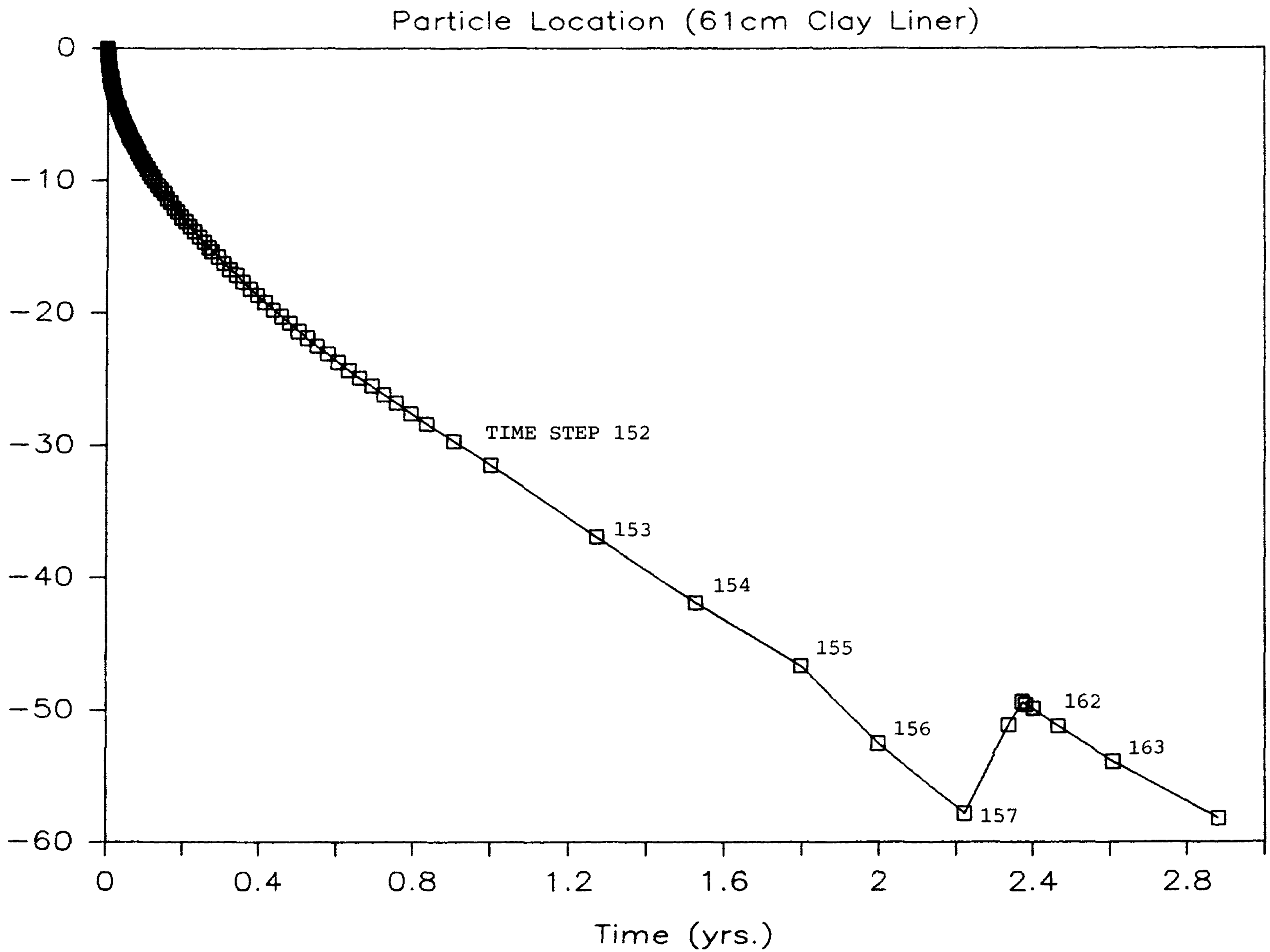

Figure 4-7. Instability of particle movement due to inaccurate PSI distributions associated with divergence at the time steps shown above.

REFERENCES

Anderson, D.C., and S.G. Jones, Clay barrier-leachate interaction, presented at National Conference on Management of Uncontrolled Hazardous Waste Sites, Washington, DC, October 31-November 2, 1983.

Bear, R.J., Hydraulics of Groundwater, McGraw-Hill, New York, 1979.

Brooks, R.H., and A.T. Corey, Hydraulic properties of porous media, Hydrology Paper No. 3, Colorado State University, Fort Collins, March 1964.

Brutsaert, W.F., A function iteration technique for solving the Richards equation applied to two-dimensional infiltration problems, Water Resources Research 6(7), 1583-1596, 1971.

Campbell, G.S., A simple method for determining unsaturated conductivity from moisture retention data, Soil Sci., 117, 311-314, 1974.

Clapp, R.B., and G.M. Hornberger, Empirical equations for some soil hydraulic properties, Water resources Research, 14, 601-604, August 1978.

Cooley, R.L., A finite difference method for unsteady flow in variably saturated porous media: application to a single pumping well, Water Resources Research, 7(6), 1607-1625, December 1971.

Cooper, H.H., Jr., J.D. Bredehoeft, and I.S. Papadopoulos, Response of a finite-diameter well to an instantaneous charge of water, Water Resources Research, 3, 263-269, 1967.

Dane, J.H., and P.J. Wierenga, Effect of hysteresis on the prediction of infiltration, redistribution, and drainage of water in a layered soil, Journal of Hydrology, 25, 229-242, 1975.

Elzeftawy, A., and K. Cartwright, Evaluating the saturated and unsaturated hydraulic conductivity of soils, in T.F. Zimmie and C.O. Riggs, eds., Permeability and Groundwater Contaminant Transport, American Society for Testing and Materials, 168-181, 1981.

Elzeftawy, A., and B.J. Dempsey, Unsaturated transient and steady state flow of moisture in subgrade soil, Transportation Research Rec. 612, 56-61, 1976.

Freeze, R.A., The mechanism of natural ground-water recharge and discharge: 1. One-dimensional, vertical, unsteady, unsaturated flow above a recharging or discharging ground-water flow system, Water Resources Research, 5(1), 153-171, 1969.

Freeze, R.A., and A. Cherry, Groundwater, Prentice-Hall, Englewood Cliffs, NJ, 1979.

Gardner, W.R., Some steady-state solutions of the unsaturated moisture flow equation with application to evaporation from a water table, Soil Science, 8, 228-232, 1958.

Gillham, R.W, A. Klute, and D.F. Heerman, Hydraulic properties of a porous medium measurement and empirical representation, Soil Science Society of America Journal, 40, 203-207, March-April 1976.

Goode, D.J., and P.A. Smith, Procedures for modeling flow through clay liners to determine required liner thickness, draft technical resource document for public comment, EPA/530-SW-84-001, 1984.

Green, D.W., H. Dabiri, and C.F. Weinaug, Numerical modeling of unsaturated groundwater flow and comparison of the model to a field experiment, Water Resources Research, 6, 862-874, June 1970.

Haverkamp, R., M. Vauclin, J. Toma, P.J. Wierenga, and G. Vachaud, A comparison of numerical simulation models for one-dimensional infiltration, Soil Science Society of America Journal, 41, 285-294, 1977.

Hillel, D.I., Soil and Water - Physical Principles and Processes, Academic Press, New York, 1971.

Hillel, D.I., Fundamentals of Soil Physics, Academic Press, N.Y., 1980.

Hvorslev, M.J., Time lag and soil permeability in groundwater observations. U.S. Army Corps of Engineers Waterways Exp. Sta. Bull. 36, Vicksburg, Miss, 1951.

Jackson, R.D., R.J. Reginato, and C.H.M. Van Bavel, Comparison of measured and calculated hydraulic conductivities of unsaturated soils, Water Resources Research, 1(3), 375-380, 1965.

Johnson, R.A., and E.S. Wood, Unsaturated flow through clay liners, letter report in fulfillment of EPA Contract No. 68-01-6871, GCA/Technology Division, Bedford, MA, 1984.

King, L.G., Description of soil characteristics for partially saturated flow, Soil Science Society of America Proceedings, 29(4), 359-362, 1965.

Kunze, R.J., and D.R. Nielsen, Finite-difference solutions of the infiltration equation, Soil Science, 134(2), 81-88, August 1982.

Li, E.A., V.D. Shanholtz, and E.W. Carson, Estimating saturated hydraulic conductivity and capillary potential at the wetting front, Dept. of Agr. Eng., VA Polytech. Inst. and State Univ., Blacksburg, 1976.

McQueen, I.S., and R.F. Miller, Approximating soil moisture characteristics from limited data: empirical evidence and tentative model, Water Resources Research, 10(3), 521-527, June 1974.

Milly, P.C.D., and P.S. Eagleson, The coupled transport of water and heat in a vertical soil column under atmospheric excitation, R.M. Parsons Laboratory for Water Resources and Hydrodynamics, M.I.T., Technical Report No. 258, July 1980.

Morel-Seytoux, H.J., Two-phase flows in porous media, 119-120, in V.T. Chin, ed., Advances in Hydroscience, Academic Press, N.Y., 1973.

Mualem, Y., A new model for predicting the hydraulic conductivity of unsaturated porous media, Water Resources Research, 12(3), 513-522, June 1976.

Mualem, Y., Hydraulic conductivity of unsaturated porous media: generalized microscopic approach, Water Resources Research 14(2), 325-334, April 1978.

Philip, J.R., The theory of infiltration: 6. Effect of water depth over soil, Soil Science, 85(5), 278-286, 1958.

Philip, J.R., Theory of infiltration, 215-297, in V.T. Chow, ed., Advances in Hydroscience, Vol 5, Academic Press, N.Y., 1969.

Pinder, G.F., and W.G. Gray, Finite Element Simulation in Surface and Subsurface Hydrology, Academic Press, N.Y., 1977.

Papadopoulos, I.S., J.D. Bredehoeft, and H.H. Cooper, On the analysis of slug test data, Water Resources Research, 9, 1087-1089, 1973.

Ragab, R., J. Feyen, and D. Hillel, Comparison of experimental and simulated infiltration profiles in sand, Soil Science, 133(1), 61-64, January 1982.

Reeder, J.W., D.L. Freyberg, J.B. Franzini, and I. Remson, Infiltration under rapidly varying surface water depths, Water Resources Research, 16(1), 97-114, February 1980.

Rogowski, A.S., Watershed physics: model of the soil moisture characteristic, Water Resources Research, 7(6), 1575-1582, December 1971.

APPENDIX A

LIST OF SYMBOLS USED IN TEXT

Symbol	FORTRAN Variable	Definition
α	ALPHA	Temporal weighting parameter
γ_ω		Specific weight of water
$\theta(\psi)$	RMOIST	Moisture content (equals n at saturation point)
θ_e		Average element moisture content
θ_r		Non-reducible moisture content
θ_s		Saturated water content
θ_{15}		Moisture content at 1.5×10^4 cm pressure
ϕ		Total head ($\phi = \psi + z$)
ψ	PSI	Pressure head (matric potential when $\psi < 0$)
ψ_b		Air entry pressure
ψ_i	PSICRT	Critical pressure along characteristic moisture curve
ψ_s		Matric potential at the liner/site soil interface, or the matric potential at saturation
ψ_{15}		1.5×10^4 cm or 15 bars pressure
$\Delta\psi$	DPSI	Change in pressure between successive iterations
$C(\psi)$	C, COLD	Specific moisture capacity ($\partial\theta/\partial\psi$)
h_1	H	Depth of liquid impoundment
k	RELK	Relative hydraulic conductivity (K/K_s)
K		Hydraulic conductivity
K_e		Element, hydraulic conductivity
K_s	SATK	Saturated, hydraulic conductivity
$K(\psi)$	RKL	Unsaturated, hydraulic conductivity
n	POR	Porosity

Symbol	FORTRAN Variable	Definition
q		Darcy flux
q_e	FLUX	Darcy element flux
S_e		Effective saturation
t	TIME	Time
Δt	DT	Change in time (length of time step)
$V(\psi)$	VNEW, VOLD	Velocity vector (RHS of equation 3-9, Part I)
$\bar{v}_e$	VELO	Element interstitial velocity
W		Soil wetness
z	Z	Depth below datum at liner surface
Δz	DZ	Distance between two adjacent nodes
z_w		Depth below datum of the water table

APPENDIX B

COMPUTER PROGRAM FOR GARDNER'S ANALYTICAL SOLUTION

09/26/83   08:40:42   GCAGD.GARDNER.FORT.DATA

```
C*******
C
C      GARDNER.FORT   STEADY STATE EVAPORATION IN UNSATURATED SOIL
C
C      SEE W.R. GARDNER, SOME STEADY-STATE SOLUTIONS OF THE UNSATURATED
C           MOISTURE FLOW EQUATION WITH APPLICATION TO EVAPORATION FROM
C           A WATER TABLE, SOIL SCIENCE, 85, 228-232, 1958.
C
C      STEADY STATE SOLUTION FOR SOIL WITH HYDRAULIC CONDUCTIVITY
C      FUNCTION AS FOLLOWS:
C
C     K =      A               S = - (PSI) SUCTION
C
C          ----------
C           4
C          S   + BETA
C
C      DAN GOODE       JUNE 1983
C
C****************
C
      IRD=5
      IPRT=6
      IRSLT=9
C      READ SOIL FUNCTION PARAMETERS
      READ(IRD,*) A,B,WT,FLUX
      WRITE(IPRT,2001) A,B,WT,FLUX
C      COMPUTE SOLUTION PARAMETERS
      SQR2=1.4142136
      ALPHA=FLUX/A
      BETA=ALPHA*B + 1.0
      RO=(BETA/ALPHA)**0.25
      RO2=RO*RO
      RO3=RO2*RO
      AINV=1.0/ALPHA
      TERM1=1.0/(4.*RO3*SQR2)
      TERM2=1./(2.*RO3*SQR2)
      WRITE(IPRT,2004) ALPHA,BETA,RO,RO2,RO3,AINV,TERM1,TERM2
C
C      READ STEPPING FOR SOLUTION
      READ(IRD,*) PF,DPF,PFEND,MAX
      WRITE(IPRT,2002) PF,DPF,PFEND,MAX
C
      PF=PF-DPF
C
      DO 10 I=1,MAX
      PF=PF+DPF
      IF(PF.GT.PFEND) GOTO 999
C
      S=10.0**PF
      PSI=-S
      S2=S*S
      TERM3=RO*S*SQR2
      ARG1=(S2+TERM3+RO2)/(S2-TERM3+RO2)
      ARG2=TERM3/(RO2-S2)
C
      PARM=ATAN(ARG2)
      IF (ARG2.LT.0) PARM=PARM + 3.1415927
      Z=(ALOG(ARG1)*TERM1+PARM*TERM2)*AINV-WT
C
      WRITE(IPRT,2003) I,Z,PSI,PF,S,S2,TERM3,ARG1,ARG2
```

```
      WRITE(IRSLT,2003) I,Z,PSI,PF
   10 CONTINUE
C
  999 STOP
 2001 FORMAT(//' GARDNER OUTPUT'//
     *' SOIL FUNCTION PARAMETERS'//
     *' A',15(1H.),'A=',1PE10.2/
     *' B',15(1H.),'B=',1PE10.2/
     *' DEPTH TO WATER TABLE',5(1H.),'WT=',0PF10.2/
     *' FLUX UPWARD',7(1H.),'FLUX=',1PE10.2)
 2002 FORMAT(//' SOLUTION STEPPING PARAMETERS'//
     *' BEGINNING SUCTION',10(1H.),'PF=',0PF10.2/
     *' PF INCREMENT',10(1H.),'DPF=',0PF10.3/
     *' FINAL PF',12(1H.),'PFEND=',0PF10.2/
     *' MAXIMUM NUMBER OF POINTS',5(1H.),'MAX=',I10///
     *'         I          Z        PSI        PF          S          S2       TERM
     *3          ARG1       ARG2',/)
 2003 FORMAT(I10,0PF10.3,1PE10.2,0PF10.3,6(1PE10.2))
 2004 FORMAT(//' INTERMEDIATE TERMS'//
     *' ALPHA=',5(1H.),1PE10.4/
     *' BETA=',5(1H.),1PE10.4/
     *' RO=',5(1H.),1PE10.4/
     *' RO2=',5(1H.),1PE10.4/
     *' RO3=',5(1H.),1PE10.4/
     *' AINV=',5(1H.),1PE10.4/
     *' TERM1=',5(1H.),1PE10.4/
     *' TERM2=',5(1H.),1PE10.4///)
      END
```

APPENDIX C

PARTIAL LIST OF FORTRAN VARIABLES IN SOILINER

ALPHAS - Temporary storage of the weighting parameter, ALPHA.

AM1 - 1.0 - ALPHA.

AVEL - Average particle velocity of three successive time steps.

C - Array containing the average node values of the specific moisture capacity, based on data from two adjacent elements (see CL).

CDIST - Distance traveled by the particle from its last position to the point of breakthrough.

CHANGE - Absolute value of the change in pressure at a given node between successive time steps.

CL - Each interior, mesh centered node is broken into two nodes, one associated with each adjacent element; the CL array contains values of the specific moisture capacity (i.e., the derivative of moisture with respect to pressure) for this expanded set of nodes.

DELZ - Absolute value of the distance between two adjacent nodes.

DPSI - Array holding nodal values of the change in pressure between two successive iterations.

DSTOR - Total moisture storage (cm) during time DT for all elements in the flow domain.

DTV - Time required for the particle to travel from its previous position to the point of breakthrough.

DZ - Distance (cm) between two adjacent nodes.

DZN - Distance between midpoint of two adjacent elements.

EFLUX - Mass balance absolute error (cm/sec) considering fluxes and storage rate at a given time step.

EREL - Relative mass balance storage error.

ERR - The maximum, absolute value at a node for the change in pressure between successive iterations.

EVOL - Mass balance absolute error (cm) considering input, output, and storage per unit surface area during time step DT.

ETOT - Mass balance absolute error (cm) considering cummulative input, ouput, and storage per unit surface area.

F	-	Forcing vector of the matrix equation solved by the Thomas algorithm (contains all knowns on the right hand side of equation 3-19, Part I).
FLUX1	-	Flux into top element (cm/sec) of the flow domain at the current time step.
FLUX10	-	Flux1 from previous time step.
FLUX2	-	Flux leaving the flow domain (cm/sec) at the current time step.
FLUX20	-	Flux2 from previous time step.
IARG	-	Variable passed as an integer argument to subroutines.
IOUT	-	Integer flag indicating special output time to be printed.
ITER	-	Number of k-iterations for a given time step.
IWARN	-	Total number of times maximum iteration warning is given.
JSOIL	-	Variable containing soil code number; used to determine appropriate characteristic curve for a given element.
KODE	-	Integer flag used to indicate that information associated with a special output time is to be printed.
KODEBT	-	Integer flag indicating time step when breakthrough occurred.
KODESS	-	Integer flag indicating time step when steady state achieved.
NELEMV	-	Element of particle location.
NKITER	-	Total number of k-iterations (number of times matrix equation solved).
NMAX	-	Maximum number of nodes (set at 200).
NODEV1	-	Upper node of NELEMV.
NODEV2	-	Lower node of NELEMV.
NOUT1	-	Index used to determine which special output time is to be printed next.
NPM1	-	Number of nodes minus one.
NPM2	-	Number of nodes minus two.
NT	-	Total number of time steps.
NUMEL	-	Total number of elements.
NUMEL2	-	Two times the number of time elements.

NX - Number of data points for the velocity integration scheme using Simpson's Rule.

OLDC - Array containing nodal values of the specific moisture capacity from the previous time step.

OLDML - Array holding nodal values of moisture content from the previous time step, for the expanded node set (see CL).

OLDTD - Total distance traveled by the particle at the previous time step.

PSILN - Log Base 10 of the negative matric potential for a particular node (PSI must be less than 0).

PSINEG - Negative value of PSI.

PSIOLD - Array holding pressure distribution from previous time step.

RATEST - Rate of liquid storage during infiltration over the period DT.

RELK - Fraction used to multiply the saturated conductivity value for a given node to determine the relative conductivity of the node when it is unsaturated.

RK - The calculated, unsaturated conductivity at a given node.

RKL - Array holding nodal values of conductivity for the expanded node set (see CL).

RMOIST - Moisture content at a given node.

RMSTL - Array holding nodal values of moisture content for the expanded set of nodes (see CL).

SRPSI - Array holding the pressure distribution as calculated using the successive relaxation algorithm.

SSPSI - Array holding the final steady state pressure distribution.

STARK - Element hydraulic conductivity.

STOR - Cumulative moisture storage per unit surface are (cm) for all elements in the flow domain up to the current time.

SYEAR - Seconds per year.

TDIST - Total distance traveled by the particle.

TIME1 - Temporary storage for the variable TIME.

TIMEBT - Time at the point of particle breakthrough (cm).

TOTV1 - Cumulative volume flux per unit area (cm) through the top element of the flow domain.

TOTV2 - Cumulative volume flux per unit area (cm) through the bottom of the flow domain.

TOUT1 - Variable holding the current special output time (after each special output time, TOUT1 is assigned the next special output time).

VEL - Calculated interstitial velocity (subroutine OUTPUT).

VELO - Array holding interstitial velocity data necessary for integration of the velocity function.

VNEW - Array holding current values of the velocity vector (equation 3-19, Part I) which is updated each interation.

VOLD - Array holding valves of the velocity vector (equation 3-19, Part I) from the previous time step.

VOL1 - Volume flux per unit surface area (cm) through the top element over the time period DT.

VOL2 - Volume flux per time unit surface area (cm) through the bottom element over the time period DT.

VTIME - Array holding times required for integration of the velocity function.

WORK - Working array containing coefficients of the stiffness matrix used in the Thomas algorithm.

YDTV - DTV (years).

YTIME - Time in years.

ZDIST - Particle distance traveled over the time period of integration.

ZZ - Midpoint of an element.

APPENDIX D

EXAMPLE OF INPUT DATA SETS

This Appendix provides annotated copies of the four data sets required as input to the SOILINER model. These data pertain to a double liner system as follows:

Layer 1: 61 cm (2") clay; $K = 1.00 \times 10^{-7}$ cm/sec, $n = 0.495$

Layer 2: 30 cm (1') sand; $K = 1.76 \times 10^{-2}$ cm/sec, $n = 0.395$

Layer 3: 91 cm (3') clay; $K = 1.00 \times 10^{-7}$ cm/sec, $n = 0.495$

Layer 4: 300 cm to the water table; $K = 1.76 \times 10^{-2}$ cm/sec, $n = 0.395$

```
TEST: DOUBLE LINER; 61/30/91/300CM CLAY/SAND/CLAY/SAND; TRANSIENT
  T  F  4.0  100.  6.05E+5  3000  1.0  .10  50  80.
  4
  0.5  1.0  2.0  3.0
  152  0.40  132
  100.0   0.0
```

Control File

RECORD TYPE 1: FORMAT(80A)

TITLE - character string identifying run

RECORD TYPE 2: UNFORMATTED

LPRINT - boolean flag for output format (F = short; T = long)
LSTEDY - boolean flag (F = transient solution; T = steady state)
ENDTIM - estimated solution time (yrs)
DT   - initial time step size (sec)
DTMAX - maximum allowable time step size (sec)
MAXNT - maximum number of time steps
ALPHA - temporal weighting parameter (0.0 ≤ ALPHA ≤ 1.0)
ERRMAX - convergence criteria (cm)
MAXIT - maximum number of iterations at any given time step
CHPARM - change parameter applied to step size between steps

RECORD TYPE 3: UNFORMATTED

NOUT   - number of special output times (4 maximum)

RECORD TYPE 4: UNFORMATTED

** NOTE: If NOUT <= 0, this record must be excluded.

TOUT(1:NOUT) - special output times (yrs)

RECORD TYPE 5: UNFORMATTED

NUMNP - number of nodes (200 maximum)
SRPARM - successive relaxation parameter (0.0 ≤ SRPARM ≤ 1.0)
NCLAYN - number of liner nodes

RECORD TYPE 6: UNFORMATTED

H   - head in impoundment (cm)
PSIBOT - pressure at base of flow domain (cm)

```
10      0.10          Grid File
10      0.50
10      1.00          RECORD TYPE 1:  UNFORMATTED
10      1.50
15      2.00            NUM - number of consecutive nodes
 6      5.00                  (downwards) with separation DELTA
10      0.10          DELTA - separation between adjacent nodes
10      0.50
10      1.00
10      1.50
30      2.00
20     15.00
```

```
131      -150.
  1      -300.
  1      -285.
  1      -270.
  1      -255.
  1      -240.
  1      -225.
  1      -210.
  1      -195.
  1      -180.
  1      -165.
  1      -150.
  1      -135.
  1      -120.
  1      -105.
  1       -90.
  1       -75.
  1       -60.
  1       -45.
  1       -30.
  1       -15.
  1         0.
```

Initial Conditions File

RECORD TYPE 1: UNFORMATTED

NUM - number of consecutive nodes with pressure PSI

PSI - initial pressure (cm)

55	12	1.000E-07	0.4950	-1.000
6	1	1.760E-02	0.3950	-16.96
70	12	1.000E-07	0.4950	-1.000
20	1	1.760E-02	0.3950	-16.96

Soil Properties File

RECORD TYPE 1: UNFORMATTED

NUM - number of nodes in soil layer
ISOIL - soil code number
SATK - saturated conductivity for soil ISOIL
POR - porosity for soil ISOIL
PSICRT - critical pressure for soil ISOIL

APPENDIX E

EXAMPLE OF OUTPUT DATA

This Appendix provides example output for the input data shown in Appendix D. The first half of the output is primarily an echo of the input (LPRINT = .TRUE.). Echoed data is followed by a brief summary of the results from the steady state algorithm, including the number of iterations and the error at convergence. If convergence had not been achieved, a warning would have been issued.

A total of six output times is provided - initial conditions, the four specified output times, and the steady state solution. After each output time a mass balance summary is printed. Finally, both the times to steady state and breakthrough are given when they occur. For this example steady state was achieved at year 3.8 and breakthrough occured at year 12.0.

Only the General Output file is included in this Appendix. The four remaining output files contain data in a format suitable for graphical analysis. These files and their output data are summarized below:

Flux Output (FLX.OUT)

L - element number
ZZ - midpoint of the element (cm)
FLUX - element flux (cm/sec)
VEL - element interstitial velocity (cm/sec)

Pressure Output (PSI.PRN)

I - node number
Z - node depth (cm)
PSI - calculated pressure (cm)
PF - log base 10 of the nodal pressure

Moisture Output (MST.PRN)

I - node number
Z - node depth (cm)
RMOIST - moisture value at the node

Particle Depth Output (PDT.OUT)

NT - time step number
TIME - time (yrs)
TDIST - total distance travel by the particle since initiation of the simulation (cm)

```
*******************************************************************************

SOILINER OUTPUT

*******************************************************************************

 TEST: DOUBLE LINER; 61/30/91/300CM CLAY/SAND/CLAY/SAND; TRANSIENT

*******************************************************************************


TEMPORAL DISCRETIZATION PARAMETERS

STEADY STATE PARM............LSTEDY=         F
  IF LSTEDY EQ T, COMPUTE STEADY STATE ONLY
  OTHERWISE, COMPUTE TRANSIENT SOLUTION
SIMULATION TIME..................ENDTIM(YRS)=      4.00
TIME STEP.............................DT=  .100E+03
MAXIMUM ALLOWABLE TIME STEP..........DTMAX= .6050E+06
MAXIMUM NUMBER OF TIME STEPS............MAXNT=      3000
TEMPORAL WEIGHTING PARAMETER..........ALPHA=     1.00
MAXIMUM ERROR FOR CONVERGENCE.......ERRMAX=     .1000
MAXIMUM ITERATIONS PER TIME STEP......MAXIT=       50
TIME STEP CHANGE PARAMETER...............CHPARM=   80.0000

NUMBER OF SPECIAL OUTPUT TIMES..........NOUT=         4


SPECIAL OUTPUT TIMES(YRS)

       .50      1.00      2.00      3.00


SPATIAL DISCRETIZATION PARAMATERS

NUMBER OF NODE POINTS..........NUMNP=       152
NUMBER OF ELEMENTS (COMPUTED)........NUMEL=      151
SUCCESSIVE RELAXATION PARAMETER.......SRPARM=      .40
NUMBER OF LINER NODES.................NCLAYN=      132
(USED FOR BREAKTHROUGH DETERMINATION)


GRID DATA

       NODE    Z         DZN

ELEMENT    DZ
```

	1	.000	
1	-.100		
	2	-.100	-.100
2	-.100		
	3	-.200	-.100
3	-.100		
	4	-.300	-.100
4	-.100		
	5	-.400	-.100
5	-.100		
	6	-.500	-.100
6	-.100		
	7	-.600	-.100
7	-.100		
	8	-.700	-.100
8	-.100		
	9	-.800	-.100
9	-.100		
	10	-.900	-.100
10	-.100		
	11	-1.000	-.300
11	-.500		
	12	-1.500	-.500
12	-.500		
	13	-2.000	-.500
13	-.500		
	14	-2.500	-.500
14	-.500		
	15	-3.000	-.500
15	-.500		
	16	-3.500	-.500
16	-.500		
	17	-4.000	-.500
17	-.500		
	18	-4.500	-.500
18	-.500		
	19	-5.000	-.500
19	-.500		
	20	-5.500	-.500
20	-.500		
	21	-6.000	-.750
21	-1.000		
	22	-7.000	-1.000
22	-1.000		
	23	-8.000	-1.000
23	-1.000		
	24	-9.000	-1.000
24	-1.000		
	25	-10.000	-1.000
25	-1.000		
	26	-11.000	-1.000
26	-1.000		

	27	-12.000	-1.000
27	-1.000		
	28	-13.000	-1.000
28	-1.000		
	29	-14.000	-1.000
29	-1.000		
	30	-15.000	-1.000
30	-1.000		
	31	-16.000	-1.250
31	-1.500		
	32	-17.500	-1.500
32	-1.500		
	33	-19.000	-1.500
33	-1.500		
	34	-20.500	-1.500
34	-1.500		
	35	-22.000	-1.500
35	-1.500		
	36	-23.500	-1.500
36	-1.500		
	37	-25.000	-1.500
37	-1.500		
	38	-26.500	-1.500
38	-1.500		
	39	-28.000	-1.500
39	-1.500		
	40	-29.500	-1.500
40	-1.500		
	41	-31.000	-1.750
41	-2.000		
	42	-33.000	-2.000
42	-2.000		
	43	-35.000	-2.000
43	-2.000		
	44	-37.000	-2.000
44	-2.000		
	45	-39.000	-2.000
45	-2.000		
	46	-41.000	-2.000
46	-2.000		
	47	-43.000	-2.000
47	-2.000		
	48	-45.000	-2.000
48	-2.000		
	49	-47.000	-2.000
49	-2.000		
	50	-49.000	-2.000
50	-2.000		
	51	-51.000	-2.000
51	-2.000		
	52	-53.000	-2.000
52	-2.000		

```
         53   -55.000    -2.000
53     -2.000
         54   -57.000    -2.000
54     -2.000
         55   -59.000    -2.000
55     -2.000
         56   -61.000    -3.500
56     -5.000
         57   -66.000    -5.000
57     -5.000
         58   -71.000    -5.000
58     -5.000
         59   -76.000    -5.000
59     -5.000
         60   -81.000    -5.000
60     -5.000
         61   -86.000    -5.000
61     -5.000
         62   -91.000    -2.550
62       -.100
         63   -91.100      -.100
63       -.100
         64   -91.200      -.100
64       -.100
         65   -91.300      -.100
65       -.100
         66   -91.400      -.100
66       -.100
         67   -91.500      -.100
67       -.100
         68   -91.600      -.100
68       -.100
         69   -91.700      -.100
69       -.100
         70   -91.800      -.100
70       -.100
         71   -91.900      -.100
71       -.100
         72   -92.000      -.300
72       -.500
         73   -92.500      -.500
73       -.500
         74   -93.000      -.500
74       -.500
         75   -93.500      -.500
75       -.500
         76   -94.000      -.500
76       -.500
         77   -94.500      -.500
77       -.500
         78   -95.000      -.500
78       -.500
```

```
       79   -95.500      -.500
79       -.500
       80   -96.000      -.500
80       -.500
       81   -96.500      -.500
81       -.500
       82   -97.000      -.750
82      -1.000
       83   -98.000     -1.000
83      -1.000
       84   -99.000     -1.000
84      -1.000
       85  -100.000     -1.000
85      -1.000
       86  -101.000     -1.000
86      -1.000
       87  -102.000     -1.000
87      -1.000
       88  -103.000     -1.000
88      -1.000
       89  -104.000     -1.000
89      -1.000
       90  -105.000     -1.000
90      -1.000
       91  -106.000     -1.000
91      -1.000
       92  -107.000     -1.250
92      -1.500
       93  -108.500     -1.500
93      -1.500
       94  -110.000     -1.500
94      -1.500
       95  -111.500     -1.500
95      -1.500
       96  -113.000     -1.500
96      -1.500
       97  -114.500     -1.500
97      -1.500
       98  -116.000     -1.500
98      -1.500
       99  -117.500     -1.500
99      -1.500
      100  -119.000     -1.500
100     -1.500
      101  -120.500     -1.500
101     -1.500
      102  -122.000     -1.750
102     -2.000
      103  -124.000     -2.000
103     -2.000
      104  -126.000     -2.000
104     -2.000
```

```
       105  -128.000     -2.000
105      -2.000
       106  -130.000     -2.000
106      -2.000
       107  -132.000     -2.000
107      -2.000
       108  -134.000     -2.000
108      -2.000
       109  -136.000     -2.000
109      -2.000
       110  -138.000     -2.000
110      -2.000
       111  -140.000     -2.000
111      -2.000
       112  -142.000     -2.000
112      -2.000
       113  -144.000     -2.000
113      -2.000
       114  -146.000     -2.000
114      -2.000
       115  -148.000     -2.000
115      -2.000
       116  -150.000     -2.000
116      -2.000
       117  -152.000     -2.000
117      -2.000
       118  -154.000     -2.000
118      -2.000
       119  -156.000     -2.000
119      -2.000
       120  -158.000     -2.000
120      -2.000
       121  -160.000     -2.000
121      -2.000
       122  -162.000     -2.000
122      -2.000
       123  -164.000     -2.000
123      -2.000
       124  -166.000     -2.000
124      -2.000
       125  -168.000     -2.000
125      -2.000
       126  -170.000     -2.000
126      -2.000
       127  -172.000     -2.000
127      -2.000
       128  -174.000     -2.000
128      -2.000
       129  -176.000     -2.000
129      -2.000
       130  -178.000     -2.000
130      -2.000
```

```
      131  -180.000    -2.000
131     -2.000
      132  -182.000    -8.500
132   -15.000
      133  -197.000   -15.000
133   -15.000
      134  -212.000   -15.000
134   -15.000
      135  -227.000   -15.000
135   -15.000
      136  -242.000   -15.000
136   -15.000
      137  -257.000   -15.000
137   -15.000
      138  -272.000   -15.000
138   -15.000
      139  -287.000   -15.000
139   -15.000
      140  -302.000   -15.000
140   -15.000
      141  -317.000   -15.000
141   -15.000
      142  -332.000   -15.000
142   -15.000
      143  -347.000   -15.000
143   -15.000
      144  -362.000   -15.000
144   -15.000
      145  -377.000   -15.000
145   -15.000
      146  -392.000   -15.000
146   -15.000
      147  -407.000   -15.000
147   -15.000
      148  -422.000   -15.000
148   -15.000
      149  -437.000   -15.000
149   -15.000
      150  -452.000   -15.000
150   -15.000
      151  -467.000   -15.000
151   -15.000
      152  -482.000
```

SOIL PROPERTIES

ELEMENT	ISOIL	SATK	POR	PSICRT
1	12	1.000E-07	.4950	-1.000
2	12	1.000E-07	.4950	-1.000
3	12	1.000E-07	.4950	-1.000

4	12	1.000E-07	.4950	-1.000
5	12	1.000E-07	.4950	-1.000
6	12	1.000E-07	.4950	-1.000
7	12	1.000E-07	.4950	-1.000
8	12	1.000E-07	.4950	-1.000
9	12	1.000E-07	.4950	-1.000
10	12	1.000E-07	.4950	-1.000
11	12	1.000E-07	.4950	-1.000
12	12	1.000E-07	.4950	-1.000
13	12	1.000E-07	.4950	-1.000
14	12	1.000E-07	.4950	-1.000
15	12	1.000E-07	.4950	-1.000
16	12	1.000E-07	.4950	-1.000
17	12	1.000E-07	.4950	-1.000
18	12	1.000E-07	.4950	-1.000
19	12	1.000E-07	.4950	-1.000
20	12	1.000E-07	.4950	-1.000
21	12	1.000E-07	.4950	-1.000
22	12	1.000E-07	.4950	-1.000
23	12	1.000E-07	.4950	-1.000
24	12	1.000E-07	.4950	-1.000
25	12	1.000E-07	.4950	-1.000
26	12	1.000E-07	.4950	-1.000
27	12	1.000E-07	.4950	-1.000
28	12	1.000E-07	.4950	-1.000
29	12	1.000E-07	.4950	-1.000
30	12	1.000E-07	.4950	-1.000
31	12	1.000E-07	.4950	-1.000
32	12	1.000E-07	.4950	-1.000
33	12	1.000E-07	.4950	-1.000
34	12	1.000E-07	.4950	-1.000
35	12	1.000E-07	.4950	-1.000
36	12	1.000E-07	.4950	-1.000
37	12	1.000E-07	.4950	-1.000
38	12	1.000E-07	.4950	-1.000
39	12	1.000E-07	.4950	-1.000
40	12	1.000E-07	.4950	-1.000
41	12	1.000E-07	.4950	-1.000
42	12	1.000E-07	.4950	-1.000
43	12	1.000E-07	.4950	-1.000
44	12	1.000E-07	.4950	-1.000
45	12	1.000E-07	.4950	-1.000
46	12	1.000E-07	.4950	-1.000
47	12	1.000E-07	.4950	-1.000
48	12	1.000E-07	.4950	-1.000
49	12	1.000E-07	.4950	-1.000
50	12	1.000E-07	.4950	-1.000
51	12	1.000E-07	.4950	-1.000
52	12	1.000E-07	.4950	-1.000
53	12	1.000E-07	.4950	-1.000
54	12	1.000E-07	.4950	-1.000
55	12	1.000E-07	.4950	-1.000

56	1	1.760E-02	.3950	-16.960
57	1	1.760E-02	.3950	-16.960
58	1	1.760E-02	.3950	-16.960
59	1	1.760E-02	.3950	-16.960
60	1	1.760E-02	.3950	-16.960
61	1	1.760E-02	.3950	-16.960
62	12	1.000E-07	.4950	-1.000
63	12	1.000E-07	.4950	-1.000
64	12	1.000E-07	.4950	-1.000
65	12	1.000E-07	.4950	-1.000
66	12	1.000E-07	.4950	-1.000
67	12	1.000E-07	.4950	-1.000
68	12	1.000E-07	.4950	-1.000
69	12	1.000E-07	.4950	-1.000
70	12	1.000E-07	.4950	-1.000
71	12	1.000E-07	.4950	-1.000
72	12	1.000E-07	.4950	-1.000
73	12	1.000E-07	.4950	-1.000
74	12	1.000E-07	.4950	-1.000
75	12	1.000E-07	.4950	-1.000
76	12	1.000E-07	.4950	-1.000
77	12	1.000E-07	.4950	-1.000
78	12	1.000E-07	.4950	-1.000
79	12	1.000E-07	.4950	-1.000
80	12	1.000E-07	.4950	-1.000
81	12	1.000E-07	.4950	-1.000
82	12	1.000E-07	.4950	-1.000
83	12	1.000E-07	.4950	-1.000
84	12	1.000E-07	.4950	-1.000
85	12	1.000E-07	.4950	-1.000
86	12	1.000E-07	.4950	-1.000
87	12	1.000E-07	.4950	-1.000
88	12	1.000E-07	.4950	-1.000
89	12	1.000E-07	.4950	-1.000
90	12	1.000E-07	.4950	-1.000
91	12	1.000E-07	.4950	-1.000
92	12	1.000E-07	.4950	-1.000
93	12	1.000E-07	.4950	-1.000
94	12	1.000E-07	.4950	-1.000
95	12	1.000E-07	.4950	-1.000
96	12	1.000E-07	.4950	-1.000
97	12	1.000E-07	.4950	-1.000
98	12	1.000E-07	.4950	-1.000
99	12	1.000E-07	.4950	-1.000
100	12	1.000E-07	.4950	-1.000
101	12	1.000E-07	.4950	-1.000
102	12	1.000E-07	.4950	-1.000
103	12	1.000E-07	.4950	-1.000
104	12	1.000E-07	.4950	-1.000
105	12	1.000E-07	.4950	-1.000
106	12	1.000E-07	.4950	-1.000
107	12	1.000E-07	.4950	-1.000

108	12	1.000E-07	.4950	-1.000
109	12	1.000E-07	.4950	-1.000
110	12	1.000E-07	.4950	-1.000
111	12	1.000E-07	.4950	-1.000
112	12	1.000E-07	.4950	-1.000
113	12	1.000E-07	.4950	-1.000
114	12	1.000E-07	.4950	-1.000
115	12	1.000E-07	.4950	-1.000
116	12	1.000E-07	.4950	-1.000
117	12	1.000E-07	.4950	-1.000
118	12	1.000E-07	.4950	-1.000
119	12	1.000E-07	.4950	-1.000
120	12	1.000E-07	.4950	-1.000
121	12	1.000E-07	.4950	-1.000
122	12	1.000E-07	.4950	-1.000
123	12	1.000E-07	.4950	-1.000
124	12	1.000E-07	.4950	-1.000
125	12	1.000E-07	.4950	-1.000
126	12	1.000E-07	.4950	-1.000
127	12	1.000E-07	.4950	-1.000
128	12	1.000E-07	.4950	-1.000
129	12	1.000E-07	.4950	-1.000
130	12	1.000E-07	.4950	-1.000
131	12	1.000E-07	.4950	-1.000
132	1	1.760E-02	.3950	-16.960
133	1	1.760E-02	.3950	-16.960
134	1	1.760E-02	.3950	-16.960
135	1	1.760E-02	.3950	-16.960
136	1	1.760E-02	.3950	-16.960
137	1	1.760E-02	.3950	-16.960
138	1	1.760E-02	.3950	-16.960
139	1	1.760E-02	.3950	-16.960
140	1	1.760E-02	.3950	-16.960
141	1	1.760E-02	.3950	-16.960
142	1	1.760E-02	.3950	-16.960
143	1	1.760E-02	.3950	-16.960
144	1	1.760E-02	.3950	-16.960
145	1	1.760E-02	.3950	-16.960
146	1	1.760E-02	.3950	-16.960
147	1	1.760E-02	.3950	-16.960
148	1	1.760E-02	.3950	-16.960
149	1	1.760E-02	.3950	-16.960
150	1	1.760E-02	.3950	-16.960
151	1	1.760E-02	.3950	-16.960

INITIAL PSI

NODE	PSI
1	-150.00
2	-150.00

3	-150.00
4	-150.00
5	-150.00
6	-150.00
7	-150.00
8	-150.00
9	-150.00
10	-150.00
11	-150.00
12	-150.00
13	-150.00
14	-150.00
15	-150.00
16	-150.00
17	-150.00
18	-150.00
19	-150.00
20	-150.00
21	-150.00
22	-150.00
23	-150.00
24	-150.00
25	-150.00
26	-150.00
27	-150.00
28	-150.00
29	-150.00
30	-150.00
31	-150.00
32	-150.00
33	-150.00
34	-150.00
35	-150.00
36	-150.00
37	-150.00
38	-150.00
39	-150.00
40	-150.00
41	-150.00
42	-150.00
43	-150.00
44	-150.00
45	-150.00
46	-150.00
47	-150.00
48	-150.00
49	-150.00
50	-150.00
51	-150.00
52	-150.00
53	-150.00
54	-150.00

55	-150.00
56	-150.00
57	-150.00
58	-150.00
59	-150.00
60	-150.00
61	-150.00
62	-150.00
63	-150.00
64	-150.00
65	-150.00
66	-150.00
67	-150.00
68	-150.00
69	-150.00
70	-150.00
71	-150.00
72	-150.00
73	-150.00
74	-150.00
75	-150.00
76	-150.00
77	-150.00
78	-150.00
79	-150.00
80	-150.00
81	-150.00
82	-150.00
83	-150.00
84	-150.00
85	-150.00
86	-150.00
87	-150.00
88	-150.00
89	-150.00
90	-150.00
91	-150.00
92	-150.00
93	-150.00
94	-150.00
95	-150.00
96	-150.00
97	-150.00
98	-150.00
99	-150.00
100	-150.00
101	-150.00
102	-150.00
103	-150.00
104	-150.00
105	-150.00
106	-150.00

107	-150.00
108	-150.00
109	-150.00
110	-150.00
111	-150.00
112	-150.00
113	-150.00
114	-150.00
115	-150.00
116	-150.00
117	-150.00
118	-150.00
119	-150.00
120	-150.00
121	-150.00
122	-150.00
123	-150.00
124	-150.00
125	-150.00
126	-150.00
127	-150.00
128	-150.00
129	-150.00
130	-150.00
131	-150.00
132	-300.00
133	-285.00
134	-270.00
135	-255.00
136	-240.00
137	-225.00
138	-210.00
139	-195.00
140	-180.00
141	-165.00
142	-150.00
143	-135.00
144	-120.00
145	-105.00
146	-90.00
147	-75.00
148	-60.00
149	-45.00
150	-30.00
151	-15.00
152	.00

==================
BOUNDARY CONDITIONS
==================

HEAD IN IMPOUNDMENT....................H=   100.00
UNDERLYING SOIL SUCTION PRESSURE.....PSIBOT=   .000

****
STEADY STATE ALGORITHM TOOK   23 ITERATIONS, ERROR= 7.3772E-02
****

****

TIME(YRS)=   .00 TIME STEP=   0 DT=   1.000E+02 ERR=   .000E+00

ITER=   0   TOTAL K ITERATIONS=   23

PARTICLE DEPTH IN LINER=   .000E+00

****

NODE	POTENTIAL	MOISTURE	K
1	1.0000E+02	.4950	1.0000E-07
2	-1.5000E+02	.3242	1.7230E-09
3	-1.5000E+02	.3242	1.7230E-09
4	-1.5000E+02	.3242	1.7230E-09
5	-1.5000E+02	.3242	1.7230E-09
6	-1.5000E+02	.3242	1.7230E-09
7	-1.5000E+02	.3242	1.7230E-09
8	-1.5000E+02	.3242	1.7230E-09
9	-1.5000E+02	.3242	1.7230E-09
10	-1.5000E+02	.3242	1.7230E-09
11	-1.5000E+02	.3242	1.7230E-09
12	-1.5000E+02	.3242	1.7230E-09
13	-1.5000E+02	.3242	1.7230E-09
14	-1.5000E+02	.3242	1.7230E-09
15	-1.5000E+02	.3242	1.7230E-09
16	-1.5000E+02	.3242	1.7230E-09
17	-1.5000E+02	.3242	1.7230E-09
18	-1.5000E+02	.3242	1.7230E-09
19	-1.5000E+02	.3242	1.7230E-09
20	-1.5000E+02	.3242	1.7230E-09
21	-1.5000E+02	.3242	1.7230E-09
22	-1.5000E+02	.3242	1.7230E-09
23	-1.5000E+02	.3242	1.7230E-09
24	-1.5000E+02	.3242	1.7230E-09
25	-1.5000E+02	.3242	1.7230E-09
26	-1.5000E+02	.3242	1.7230E-09
27	-1.5000E+02	.3242	1.7230E-09
28	-1.5000E+02	.3242	1.7230E-09
29	-1.5000E+02	.3242	1.7230E-09

30	-1.5000E+02	.3242	1.7230E-09
31	-1.5000E+02	.3242	1.7230E-09
32	-1.5000E+02	.3242	1.7230E-09
33	-1.5000E+02	.3242	1.7230E-09
34	-1.5000E+02	.3242	1.7230E-09
35	-1.5000E+02	.3242	1.7230E-09
36	-1.5000E+02	.3242	1.7230E-09
37	-1.5000E+02	.3242	1.7230E-09
38	-1.5000E+02	.3242	1.7230E-09
39	-1.5000E+02	.3242	1.7230E-09
40	-1.5000E+02	.3242	1.7230E-09
41	-1.5000E+02	.3242	1.7230E-09
42	-1.5000E+02	.3242	1.7230E-09
43	-1.5000E+02	.3242	1.7230E-09
44	-1.5000E+02	.3242	1.7230E-09
45	-1.5000E+02	.3242	1.7230E-09
46	-1.5000E+02	.3242	1.7230E-09
47	-1.5000E+02	.3242	1.7230E-09
48	-1.5000E+02	.3242	1.7230E-09
49	-1.5000E+02	.3242	1.7230E-09
50	-1.5000E+02	.3242	1.7230E-09
51	-1.5000E+02	.3242	1.7230E-09
52	-1.5000E+02	.3242	1.7230E-09
53	-1.5000E+02	.3242	1.7230E-09
54	-1.5000E+02	.3242	1.7230E-09
55	-1.5000E+02	.3242	1.7230E-09
56	-1.5000E+02	.2682	1.7488E-07
57	-1.5000E+02	.2122	1.7750E-05
58	-1.5000E+02	.2122	1.7750E-05
59	-1.5000E+02	.2122	1.7750E-05
60	-1.5000E+02	.2122	1.7750E-05
61	-1.5000E+02	.2122	1.7750E-05
62	-1.5000E+02	.2682	1.7488E-07
63	-1.5000E+02	.3242	1.7230E-09
64	-1.5000E+02	.3242	1.7230E-09
65	-1.5000E+02	.3242	1.7230E-09
66	-1.5000E+02	.3242	1.7230E-09
67	-1.5000E+02	.3242	1.7230E-09
68	-1.5000E+02	.3242	1.7230E-09
69	-1.5000E+02	.3242	1.7230E-09
70	-1.5000E+02	.3242	1.7230E-09
71	-1.5000E+02	.3242	1.7230E-09
72	-1.5000E+02	.3242	1.7230E-09
73	-1.5000E+02	.3242	1.7230E-09
74	-1.5000E+02	.3242	1.7230E-09
75	-1.5000E+02	.3242	1.7230E-09
76	-1.5000E+02	.3242	1.7230E-09
77	-1.5000E+02	.3242	1.7230E-09
78	-1.5000E+02	.3242	1.7230E-09
79	-1.5000E+02	.3242	1.7230E-09
80	-1.5000E+02	.3242	1.7230E-09
81	-1.5000E+02	.3242	1.7230E-09

82	-1.5000E+02	.3242	1.7230E-09
83	-1.5000E+02	.3242	1.7230E-09
84	-1.5000E+02	.3242	1.7230E-09
85	-1.5000E+02	.3242	1.7230E-09
86	-1.5000E+02	.3242	1.7230E-09
87	-1.5000E+02	.3242	1.7230E-09
88	-1.5000E+02	.3242	1.7230E-09
89	-1.5000E+02	.3242	1.7230E-09
90	-1.5000E+02	.3242	1.7230E-09
91	-1.5000E+02	.3242	1.7230E-09
92	-1.5000E+02	.3242	1.7230E-09
93	-1.5000E+02	.3242	1.7230E-09
94	-1.5000E+02	.3242	1.7230E-09
95	-1.5000E+02	.3242	1.7230E-09
96	-1.5000E+02	.3242	1.7230E-09
97	-1.5000E+02	.3242	1.7230E-09
98	-1.5000E+02	.3242	1.7230E-09
99	-1.5000E+02	.3242	1.7230E-09
100	-1.5000E+02	.3242	1.7230E-09
101	-1.5000E+02	.3242	1.7230E-09
102	-1.5000E+02	.3242	1.7230E-09
103	-1.5000E+02	.3242	1.7230E-09
104	-1.5000E+02	.3242	1.7230E-09
105	-1.5000E+02	.3242	1.7230E-09
106	-1.5000E+02	.3242	1.7230E-09
107	-1.5000E+02	.3242	1.7230E-09
108	-1.5000E+02	.3242	1.7230E-09
109	-1.5000E+02	.3242	1.7230E-09
110	-1.5000E+02	.3242	1.7230E-09
111	-1.5000E+02	.3242	1.7230E-09
112	-1.5000E+02	.3242	1.7230E-09
113	-1.5000E+02	.3242	1.7230E-09
114	-1.5000E+02	.3242	1.7230E-09
115	-1.5000E+02	.3242	1.7230E-09
116	-1.5000E+02	.3242	1.7230E-09
117	-1.5000E+02	.3242	1.7230E-09
118	-1.5000E+02	.3242	1.7230E-09
119	-1.5000E+02	.3242	1.7230E-09
120	-1.5000E+02	.3242	1.7230E-09
121	-1.5000E+02	.3242	1.7230E-09
122	-1.5000E+02	.3242	1.7230E-09
123	-1.5000E+02	.3242	1.7230E-09
124	-1.5000E+02	.3242	1.7230E-09
125	-1.5000E+02	.3242	1.7230E-09
126	-1.5000E+02	.3242	1.7230E-09
127	-1.5000E+02	.3242	1.7230E-09
128	-1.5000E+02	.3242	1.7230E-09
129	-1.5000E+02	.3242	1.7230E-09
130	-1.5000E+02	.3242	1.7230E-09
131	-1.5000E+02	.3242	1.7230E-09
132	-3.0000E+02	.2277	3.6855E-08
133	-2.8500E+02	.1811	3.0568E-06

134	-2.7000E+02	.1835	3.5450E-06
135	-2.5500E+02	.1861	4.1461E-06
136	-2.4000E+02	.1889	4.8955E-06
137	-2.2500E+02	.1919	5.8427E-06
138	-2.1000E+02	.1952	7.0588E-06
139	-1.9500E+02	.1989	8.6484E-06
140	-1.8000E+02	.2028	1.0770E-05
141	-1.6500E+02	.2072	1.3670E-05
142	-1.5000E+02	.2122	1.7750E-05
143	-1.3500E+02	.2177	2.3693E-05
144	-1.2000E+02	.2242	3.2719E-05
145	-1.0500E+02	.2317	4.7177E-05
146	-9.0000E+01	.2407	7.1979E-05
147	-7.5000E+01	.2518	1.1863E-04
148	-6.0000E+01	.2660	2.1868E-04
149	-4.5000E+01	.2856	4.8107E-04
150	-3.0000E+01	.3157	1.4615E-03
151	-1.5000E+01	.3705	8.6547E-03
152	.0000E+00	.3950	1.7600E-02

****

TIME(YRS)= .50 TIME STEP= 53 DT= 2.820E+05 ERR= 6.202E-02

ITER= 3 TOTAL K ITERATIONS= 556

PARTICLE DEPTH IN LINER= -1.942E+01

****

NODE	POTENTIAL	MOISTURE	K
1	1.0000E+02	.4950	1.0000E-07
2	9.9762E+01	.4950	1.0000E-07
3	9.9524E+01	.4950	1.0000E-07
4	9.9287E+01	.4950	1.0000E-07
5	9.9049E+01	.4950	1.0000E-07
6	9.8811E+01	.4950	1.0000E-07
7	9.8573E+01	.4950	1.0000E-07
8	9.8336E+01	.4950	1.0000E-07
9	9.8098E+01	.4950	1.0000E-07
10	9.7860E+01	.4950	1.0000E-07
11	9.7622E+01	.4950	1.0000E-07
12	9.6434E+01	.4950	1.0000E-07
13	9.5245E+01	.4950	1.0000E-07
14	9.4056E+01	.4950	1.0000E-07
15	9.2867E+01	.4950	1.0000E-07
16	9.1679E+01	.4950	1.0000E-07
17	9.0490E+01	.4950	1.0000E-07
18	8.9301E+01	.4950	1.0000E-07
19	8.8112E+01	.4950	1.0000E-07

20	8.6924E+01	.4950	1.0000E-07
21	8.5735E+01	.4950	1.0000E-07
22	8.3357E+01	.4950	1.0000E-07
23	8.0980E+01	.4950	1.0000E-07
24	7.8602E+01	.4950	1.0000E-07
25	7.6225E+01	.4950	1.0000E-07
26	7.3847E+01	.4950	1.0000E-07
27	7.1470E+01	.4950	1.0000E-07
28	6.9092E+01	.4950	1.0000E-07
29	6.6715E+01	.4950	1.0000E-07
30	6.4337E+01	.4950	1.0000E-07
31	6.1959E+01	.4950	1.0000E-07
32	5.8393E+01	.4950	1.0000E-07
33	5.4827E+01	.4950	1.0000E-07
34	5.1261E+01	.4950	1.0000E-07
35	4.7694E+01	.4950	1.0000E-07
36	4.4128E+01	.4950	1.0000E-07
37	4.0562E+01	.4950	1.0000E-07
38	3.6995E+01	.4950	1.0000E-07
39	3.3429E+01	.4950	1.0000E-07
40	2.9863E+01	.4950	1.0000E-07
41	2.6297E+01	.4950	1.0000E-07
42	2.1541E+01	.4950	1.0000E-07
43	1.6786E+01	.4950	1.0000E-07
44	1.2031E+01	.4950	1.0000E-07
45	7.2763E+00	.4950	1.0000E-07
46	2.5212E+00	.4950	1.0000E-07
47	-2.3571E+00	.4947	9.6468E-08
48	-8.1784E+00	.4855	7.5128E-08
49	-1.6487E+01	.4664	4.6619E-08
50	-3.0123E+01	.4379	2.3107E-08
51	-5.2751E+01	.4020	1.0029E-08
52	-8.4043E+01	.3678	4.6604E-09
53	-1.1554E+02	.3437	2.7074E-09
54	-1.3908E+02	.3298	1.9648E-09
55	-1.5431E+02	.3221	1.6400E-09
56	-1.6531E+02	.2621	1.4066E-07
57	-1.6031E+02	.2087	1.4794E-05
58	-1.5531E+02	.2103	1.6135E-05
59	-1.5032E+02	.2120	1.7648E-05
60	-1.4532E+02	.2138	1.9362E-05
61	-1.4032E+02	.2157	2.1311E-05
62	-1.3532E+02	.2747	2.2023E-07
63	-1.3544E+02	.3318	2.0573E-09
64	-1.3555E+02	.3317	2.0543E-09
65	-1.3567E+02	.3317	2.0513E-09
66	-1.3578E+02	.3316	2.0483E-09
67	-1.3590E+02	.3315	2.0453E-09
68	-1.3601E+02	.3315	2.0424E-09
69	-1.3612E+02	.3314	2.0394E-09
70	-1.3623E+02	.3313	2.0366E-09
71	-1.3634E+02	.3313	2.0337E-09

72	-1.3645E+02	.3312	2.0308E-09
73	-1.3699E+02	.3309	2.0170E-09
74	-1.3752E+02	.3306	2.0036E-09
75	-1.3803E+02	.3304	1.9907E-09
76	-1.3854E+02	.3301	1.9782E-09
77	-1.3903E+02	.3298	1.9660E-09
78	-1.3952E+02	.3296	1.9542E-09
79	-1.3999E+02	.3293	1.9426E-09
80	-1.4046E+02	.3291	1.9314E-09
81	-1.4092E+02	.3288	1.9205E-09
82	-1.4137E+02	.3286	1.9099E-09
83	-1.4224E+02	.3281	1.8896E-09
84	-1.4307E+02	.3277	1.8707E-09
85	-1.4385E+02	.3273	1.8530E-09
86	-1.4459E+02	.3269	1.8367E-09
87	-1.4526E+02	.3266	1.8218E-09
88	-1.4589E+02	.3263	1.8083E-09
89	-1.4646E+02	.3260	1.7962E-09
90	-1.4697E+02	.3257	1.7853E-09
91	-1.4742E+02	.3255	1.7757E-09
92	-1.4783E+02	.3253	1.7673E-09
93	-1.4834E+02	.3250	1.7567E-09
94	-1.4875E+02	.3248	1.7483E-09
95	-1.4907E+02	.3247	1.7417E-09
96	-1.4932E+02	.3246	1.7366E-09
97	-1.4951E+02	.3245	1.7328E-09
98	-1.4965E+02	.3244	1.7300E-09
99	-1.4976E+02	.3243	1.7279E-09
100	-1.4983E+02	.3243	1.7264E-09
101	-1.4989E+02	.3243	1.7253E-09
102	-1.4992E+02	.3243	1.7245E-09
103	-1.4996E+02	.3242	1.7239E-09
104	-1.4997E+02	.3242	1.7235E-09
105	-1.4999E+02	.3242	1.7233E-09
106	-1.4999E+02	.3242	1.7232E-09
107	-1.5000E+02	.3242	1.7231E-09
108	-1.5000E+02	.3242	1.7230E-09
109	-1.5000E+02	.3242	1.7230E-09
110	-1.5000E+02	.3242	1.7229E-09
111	-1.5001E+02	.3242	1.7228E-09
112	-1.5002E+02	.3242	1.7226E-09
113	-1.5004E+02	.3242	1.7223E-09
114	-1.5007E+02	.3242	1.7217E-09
115	-1.5012E+02	.3242	1.7207E-09
116	-1.5021E+02	.3241	1.7189E-09
117	-1.5036E+02	.3240	1.7159E-09
118	-1.5061E+02	.3239	1.7110E-09
119	-1.5101E+02	.3237	1.7031E-09
120	-1.5164E+02	.3234	1.6907E-09
121	-1.5261E+02	.3230	1.6720E-09
122	-1.5408E+02	.3223	1.6444E-09
123	-1.5624E+02	.3212	1.6050E-09

124	-1.5937E+02	.3198	1.5505E-09
125	-1.6383E+02	.3178	1.4777E-09
126	-1.7007E+02	.3151	1.3844E-09
127	-1.7872E+02	.3115	1.2694E-09
128	-1.9062E+02	.3069	1.1342E-09
129	-2.0688E+02	.3012	9.8270E-10
130	-2.2902E+02	.2942	8.2217E-10
131	-2.5908E+02	.2860	6.6202E-10
132	-2.9963E+02	.2277	3.6959E-08
133	-2.8469E+02	.1811	3.0659E-06
134	-2.6975E+02	.1835	3.5541E-06
135	-2.5480E+02	.1861	4.1553E-06
136	-2.3984E+02	.1889	4.9047E-06
137	-2.2487E+02	.1920	5.8518E-06
138	-2.0990E+02	.1953	7.0679E-06
139	-1.9493E+02	.1989	8.6575E-06
140	-1.7994E+02	.2028	1.0779E-05
141	-1.6496E+02	.2072	1.3679E-05
142	-1.4997E+02	.2122	1.7760E-05
143	-1.3498E+02	.2178	2.3702E-05
144	-1.1999E+02	.2242	3.2728E-05
145	-1.0499E+02	.2317	4.7186E-05
146	-8.9996E+01	.2407	7.1988E-05
147	-7.4998E+01	.2518	1.1864E-04
148	-5.9999E+01	.2660	2.1868E-04
149	-4.5000E+01	.2856	4.8108E-04
150	-3.0000E+01	.3157	1.4615E-03
151	-1.5000E+01	.3705	8.6547E-03
152	.0000E+00	.3950	1.7600E-02

VOLUME BALANCE CALCULATIONS

TOP FLUX 3.3775E-07
BOTTOM FLUX 1.2555E-08
STORAGE RATE 3.1835E-07

ERROR 6.8494E-09

VOLUME IN 9.5256E-02
VOLUME OUT 3.5408E-03
STORAGE VOLUME 8.9783E-02

ERROR 1.9317E-03

CUMULATIVE CHANGES

VOLUME IN (-) 9.3592E+00
VOLUME OUT 3.1859E-01
STORAGE 8.3925E+00

ERROR 6.4816E-01

RELATIVE ERROR .077231

****

TIME(YRS)= 1.00 TIME STEP= 80 DT= 3.800E+04 ERR= 5.908E-02

ITER= 1 TOTAL K ITERATIONS= 665

PARTICLE DEPTH IN LINER= -2.926E+01

****

NODE	POTENTIAL	MOISTURE	K
1	1.0000E+02	.4950	1.0000E-07
2	9.9808E+01	.4950	1.0000E-07
3	9.9615E+01	.4950	1.0000E-07
4	9.9423E+01	.4950	1.0000E-07
5	9.9231E+01	.4950	1.0000E-07
6	9.9039E+01	.4950	1.0000E-07
7	9.8846E+01	.4950	1.0000E-07
8	9.8654E+01	.4950	1.0000E-07
9	9.8462E+01	.4950	1.0000E-07
10	9.8270E+01	.4950	1.0000E-07
11	9.8077E+01	.4950	1.0000E-07
12	9.7116E+01	.4950	1.0000E-07
13	9.6155E+01	.4950	1.0000E-07
14	9.5193E+01	.4950	1.0000E-07
15	9.4232E+01	.4950	1.0000E-07
16	9.3271E+01	.4950	1.0000E-07
17	9.2310E+01	.4950	1.0000E-07
18	9.1348E+01	.4950	1.0000E-07
19	9.0387E+01	.4950	1.0000E-07
20	8.9426E+01	.4950	1.0000E-07
21	8.8464E+01	.4950	1.0000E-07
22	8.6542E+01	.4950	1.0000E-07
23	8.4619E+01	.4950	1.0000E-07
24	8.2697E+01	.4950	1.0000E-07
25	8.0774E+01	.4950	1.0000E-07
26	7.8851E+01	.4950	1.0000E-07
27	7.6929E+01	.4950	1.0000E-07
28	7.5006E+01	.4950	1.0000E-07
29	7.3084E+01	.4950	1.0000E-07
30	7.1161E+01	.4950	1.0000E-07
31	6.9238E+01	.4950	1.0000E-07
32	6.6354E+01	.4950	1.0000E-07
33	6.3471E+01	.4950	1.0000E-07
34	6.0587E+01	.4950	1.0000E-07

35	5.7703E+01	.4950	1.0000E-07
36	5.4819E+01	.4950	1.0000E-07
37	5.1935E+01	.4950	1.0000E-07
38	4.9051E+01	.4950	1.0000E-07
39	4.6167E+01	.4950	1.0000E-07
40	4.3283E+01	.4950	1.0000E-07
41	4.0399E+01	.4950	1.0000E-07
42	3.6554E+01	.4950	1.0000E-07
43	3.2709E+01	.4950	1.0000E-07
44	2.8864E+01	.4950	1.0000E-07
45	2.5018E+01	.4950	1.0000E-07
46	2.1173E+01	.4950	1.0000E-07
47	1.7328E+01	.4950	1.0000E-07
48	1.3483E+01	.4950	1.0000E-07
49	9.6376E+00	.4950	1.0000E-07
50	5.7924E+00	.4950	1.0000E-07
51	1.9472E+00	.4950	1.0000E-07
52	-1.9802E+00	.4949	9.7381E-08
53	-6.5260E+00	.4889	8.1831E-08
54	-1.3054E+01	.4743	5.6902E-08
55	-2.5282E+01	.4474	2.9066E-08
56	-6.0802E+01	.3285	1.2970E-06
57	-5.5807E+01	.2708	2.6670E-04
58	-5.0812E+01	.2772	3.4486E-04
59	-4.5814E+01	.2843	4.5799E-04
60	-4.0816E+01	.2926	6.2857E-04
61	-3.5817E+01	.3022	8.9919E-04
62	-3.0818E+01	.3751	5.5144E-06
63	-3.1206E+01	.4359	2.2016E-08
64	-3.1599E+01	.4351	2.1637E-08
65	-3.1998E+01	.4344	2.1263E-08
66	-3.2402E+01	.4337	2.0893E-08
67	-3.2812E+01	.4329	2.0528E-08
68	-3.3228E+01	.4322	2.0167E-08
69	-3.3650E+01	.4314	1.9810E-08
70	-3.4077E+01	.4307	1.9458E-08
71	-3.4510E+01	.4299	1.9110E-08
72	-3.4949E+01	.4291	1.8766E-08
73	-3.7236E+01	.4252	1.7116E-08
74	-3.9680E+01	.4212	1.5578E-08
75	-4.2287E+01	.4170	1.4153E-08
76	-4.5063E+01	.4128	1.2840E-08
77	-4.8009E+01	.4085	1.1637E-08
78	-5.1127E+01	.4042	1.0540E-08
79	-5.4414E+01	.3998	9.5445E-09
80	-5.7864E+01	.3954	8.6454E-09
81	-6.1468E+01	.3910	7.8372E-09
82	-6.5214E+01	.3867	7.1135E-09
83	-7.3050E+01	.3783	5.8955E-09
84	-8.1192E+01	.3704	4.9395E-09
85	-8.9414E+01	.3631	4.1967E-09
86	-9.7469E+01	.3566	3.6240E-09

87	-1.0512E+02	.3508	3.1846E-09
88	-1.1219E+02	.3459	2.8483E-09
89	-1.1852E+02	.3418	2.5911E-09
90	-1.2407E+02	.3383	2.3945E-09
91	-1.2882E+02	.3355	2.2439E-09
92	-1.3281E+02	.3332	2.1283E-09
93	-1.3749E+02	.3307	2.0044E-09
94	-1.4092E+02	.3288	1.9205E-09
95	-1.4340E+02	.3275	1.8632E-09
96	-1.4518E+02	.3266	1.8237E-09
97	-1.4645E+02	.3260	1.7962E-09
98	-1.4737E+02	.3255	1.7768E-09
99	-1.4804E+02	.3252	1.7630E-09
100	-1.4852E+02	.3249	1.7529E-09
101	-1.4889E+02	.3248	1.7455E-09
102	-1.4916E+02	.3246	1.7400E-09
103	-1.4942E+02	.3245	1.7347E-09
104	-1.4960E+02	.3244	1.7310E-09
105	-1.4974E+02	.3243	1.7282E-09
106	-1.4984E+02	.3243	1.7262E-09
107	-1.4992E+02	.3243	1.7245E-09
108	-1.5000E+02	.3242	1.7230E-09
109	-1.5008E+02	.3242	1.7214E-09
110	-1.5017E+02	.3241	1.7195E-09
111	-1.5030E+02	.3241	1.7171E-09
112	-1.5046E+02	.3240	1.7138E-09
113	-1.5069E+02	.3239	1.7092E-09
114	-1.5101E+02	.3237	1.7029E-09
115	-1.5146E+02	.3235	1.6943E-09
116	-1.5207E+02	.3232	1.6825E-09
117	-1.5289E+02	.3228	1.6667E-09
118	-1.5399E+02	.3223	1.6460E-09
119	-1.5545E+02	.3216	1.6192E-09
120	-1.5737E+02	.3207	1.5850E-09
121	-1.5985E+02	.3196	1.5424E-09
122	-1.6303E+02	.3181	1.4903E-09
123	-1.6710E+02	.3163	1.4277E-09
124	-1.7224E+02	.3142	1.3541E-09
125	-1.7872E+02	.3115	1.2695E-09
126	-1.8684E+02	.3083	1.1746E-09
127	-1.9699E+02	.3046	1.0707E-09
128	-2.0966E+02	.3003	9.5993E-10
129	-2.2545E+02	.2953	8.4518E-10
130	-2.4509E+02	.2896	7.2986E-10
131	-2.6949E+02	.2834	6.1768E-10
132	-2.9973E+02	.2277	3.6931E-08
133	-2.8477E+02	.1811	3.0634E-06
134	-2.6982E+02	.1835	3.5516E-06
135	-2.5485E+02	.1861	4.1528E-06
136	-2.3988E+02	.1889	4.9022E-06
137	-2.2491E+02	.1920	5.8494E-06
138	-2.0993E+02	.1953	7.0655E-06

139	-1.9495E+02	.1989	8.6551E-06
140	-1.7996E+02	.2028	1.0776E-05
141	-1.6497E+02	.2072	1.3677E-05
142	-1.4998E+02	.2122	1.7757E-05
143	-1.3499E+02	.2178	2.3699E-05
144	-1.1999E+02	.2242	3.2726E-05
145	-1.0499E+02	.2317	4.7184E-05
146	-8.9997E+01	.2407	7.1985E-05
147	-7.4999E+01	.2518	1.1864E-04
148	-5.9999E+01	.2660	2.1868E-04
149	-4.5000E+01	.2856	4.8107E-04
150	-3.0000E+01	.3157	1.4615E-03
151	-1.5000E+01	.3705	8.6547E-03
152	.0000E+00	.3950	1.7600E-02

VOLUME BALANCE CALCULATIONS

TOP FLUX 2.9226E-07
BOTTOM FLUX 9.4161E-09
STORAGE RATE 2.8520E-07

ERROR 2.3541E-09

VOLUME IN 1.1106E-02
VOLUME OUT 3.5781E-04
STORAGE VOLUME 1.0838E-02

ERROR 8.9458E-05

CUMULATIVE CHANGES

VOLUME IN (-) 1.4122E+01
VOLUME OUT 4.8415E-01
STORAGE 1.2812E+01

ERROR 8.2564E-01

RELATIVE ERROR .064443

****

TIME(YRS)= 2.00 TIME STEP= 133 DT= 7.600E+04 ERR= 1.153E-02

ITER= 2 TOTAL K ITERATIONS= 829

PARTICLE DEPTH IN LINER= -4.744E+01

****

NODE	POTENTIAL	MOISTURE	K
1	1.0000E+02	.4950	1.0000E-07
2	9.9845E+01	.4950	1.0000E-07
3	9.9691E+01	.4950	1.0000E-07
4	9.9536E+01	.4950	1.0000E-07
5	9.9381E+01	.4950	1.0000E-07
6	9.9227E+01	.4950	1.0000E-07
7	9.9072E+01	.4950	1.0000E-07
8	9.8917E+01	.4950	1.0000E-07
9	9.8762E+01	.4950	1.0000E-07
10	9.8608E+01	.4950	1.0000E-07
11	9.8453E+01	.4950	1.0000E-07
12	9.7680E+01	.4950	1.0000E-07
13	9.6906E+01	.4950	1.0000E-07
14	9.6133E+01	.4950	1.0000E-07
15	9.5359E+01	.4950	1.0000E-07
16	9.4586E+01	.4950	1.0000E-07
17	9.3812E+01	.4950	1.0000E-07
18	9.3039E+01	.4950	1.0000E-07
19	9.2265E+01	.4950	1.0000E-07
20	9.1492E+01	.4950	1.0000E-07
21	9.0718E+01	.4950	1.0000E-07
22	8.9171E+01	.4950	1.0000E-07
23	8.7624E+01	.4950	1.0000E-07
24	8.6077E+01	.4950	1.0000E-07
25	8.4530E+01	.4950	1.0000E-07
26	8.2983E+01	.4950	1.0000E-07
27	8.1436E+01	.4950	1.0000E-07
28	7.9889E+01	.4950	1.0000E-07
29	7.8342E+01	.4950	1.0000E-07
30	7.6796E+01	.4950	1.0000E-07
31	7.5249E+01	.4950	1.0000E-07
32	7.2928E+01	.4950	1.0000E-07
33	7.0608E+01	.4950	1.0000E-07
34	6.8287E+01	.4950	1.0000E-07
35	6.5967E+01	.4950	1.0000E-07
36	6.3646E+01	.4950	1.0000E-07
37	6.1326E+01	.4950	1.0000E-07
38	5.9005E+01	.4950	1.0000E-07
39	5.6685E+01	.4950	1.0000E-07
40	5.4364E+01	.4950	1.0000E-07
41	5.2044E+01	.4950	1.0000E-07
42	4.8950E+01	.4950	1.0000E-07
43	4.5856E+01	.4950	1.0000E-07
44	4.2762E+01	.4950	1.0000E-07
45	3.9668E+01	.4950	1.0000E-07
46	3.6574E+01	.4950	1.0000E-07
47	3.3480E+01	.4950	1.0000E-07
48	3.0386E+01	.4950	1.0000E-07
49	2.7293E+01	.4950	1.0000E-07

50	2.4199E+01	.4950	1.0000E-07
51	2.1105E+01	.4950	1.0000E-07
52	1.8011E+01	.4950	1.0000E-07
53	1.4917E+01	.4950	1.0000E-07
54	1.1823E+01	.4950	1.0000E-07
55	8.7290E+00	.4950	1.0000E-07
56	5.6350E+00	.4450	4.1952E-05
57	1.0635E+01	.3950	1.7600E-02
58	1.5635E+01	.3950	1.7600E-02
59	2.0635E+01	.3950	1.7600E-02
60	2.5635E+01	.3950	1.7600E-02
61	3.0635E+01	.3950	1.7600E-02
62	3.5635E+01	.4450	4.1952E-05
63	3.5480E+01	.4950	1.0000E-07
64	3.5325E+01	.4950	1.0000E-07
65	3.5171E+01	.4950	1.0000E-07
66	3.5016E+01	.4950	1.0000E-07
67	3.4861E+01	.4950	1.0000E-07
68	3.4706E+01	.4950	1.0000E-07
69	3.4552E+01	.4950	1.0000E-07
70	3.4397E+01	.4950	1.0000E-07
71	3.4242E+01	.4950	1.0000E-07
72	3.4088E+01	.4950	1.0000E-07
73	3.3314E+01	.4950	1.0000E-07
74	3.2541E+01	.4950	1.0000E-07
75	3.1767E+01	.4950	1.0000E-07
76	3.0994E+01	.4950	1.0000E-07
77	3.0220E+01	.4950	1.0000E-07
78	2.9447E+01	.4950	1.0000E-07
79	2.8673E+01	.4950	1.0000E-07
80	2.7900E+01	.4950	1.0000E-07
81	2.7126E+01	.4950	1.0000E-07
82	2.6353E+01	.4950	1.0000E-07
83	2.4806E+01	.4950	1.0000E-07
84	2.3259E+01	.4950	1.0000E-07
85	2.1712E+01	.4950	1.0000E-07
86	2.0165E+01	.4950	1.0000E-07
87	1.8618E+01	.4950	1.0000E-07
88	1.7071E+01	.4950	1.0000E-07
89	1.5524E+01	.4950	1.0000E-07
90	1.3977E+01	.4950	1.0000E-07
91	1.2430E+01	.4950	1.0000E-07
92	1.0884E+01	.4950	1.0000E-07
93	8.5631E+00	.4950	1.0000E-07
94	6.2427E+00	.4950	1.0000E-07
95	3.9223E+00	.4950	1.0000E-07
96	1.6019E+00	.4950	1.0000E-07
97	-7.1853E-01	.4950	1.0000E-07
98	-3.1545E+00	.4941	9.4222E-08
99	-5.8554E+00	.4902	8.4513E-08
100	-8.9565E+00	.4838	7.2003E-08
101	-1.2653E+01	.4753	5.8249E-08

102	-1.7205E+01	.4648	4.4748E-08
103	-2.5232E+01	.4475	2.9139E-08
104	-3.6299E+01	.4268	1.7765E-08
105	-5.1053E+01	.4043	1.0564E-08
106	-6.9112E+01	.3824	6.4637E-09
107	-8.8577E+01	.3638	4.2642E-09
108	-1.0676E+02	.3497	3.1014E-09
109	-1.2167E+02	.3398	2.4767E-09
110	-1.3275E+02	.3333	2.1300E-09
111	-1.4053E+02	.3290	1.9297E-09
112	-1.4591E+02	.3263	1.8078E-09
113	-1.4973E+02	.3244	1.7283E-09
114	-1.5265E+02	.3229	1.6713E-09
115	-1.5510E+02	.3218	1.6255E-09
116	-1.5741E+02	.3207	1.5843E-09
117	-1.5978E+02	.3196	1.5435E-09
118	-1.6237E+02	.3184	1.5009E-09
119	-1.6531E+02	.3171	1.4547E-09
120	-1.6870E+02	.3156	1.4040E-09
121	-1.7266E+02	.3140	1.3483E-09
122	-1.7729E+02	.3121	1.2874E-09
123	-1.8272E+02	.3099	1.2213E-09
124	-1.8908E+02	.3075	1.1503E-09
125	-1.9654E+02	.3048	1.0750E-09
126	-2.0528E+02	.3017	9.9610E-10
127	-2.1552E+02	.2984	9.1462E-10
128	-2.2752E+02	.2947	8.3170E-10
129	-2.4157E+02	.2906	7.4864E-10
130	-2.5802E+02	.2862	6.6678E-10
131	-2.7728E+02	.2816	5.8748E-10
132	-2.9980E+02	.2277	3.6911E-08
133	-2.8483E+02	.1811	3.0617E-06
134	-2.6986E+02	.1835	3.5499E-06
135	-2.5489E+02	.1861	4.1511E-06
136	-2.3991E+02	.1889	4.9004E-06
137	-2.2493E+02	.1920	5.8476E-06
138	-2.0995E+02	.1953	7.0637E-06
139	-1.9496E+02	.1989	8.6533E-06
140	-1.7997E+02	.2028	1.0775E-05
141	-1.6498E+02	.2072	1.3675E-05
142	-1.4998E+02	.2122	1.7755E-05
143	-1.3499E+02	.2178	2.3697E-05
144	-1.1999E+02	.2242	3.2724E-05
145	-1.0500E+02	.2317	4.7182E-05
146	-8.9998E+01	.2407	7.1984E-05
147	-7.4999E+01	.2518	1.1864E-04
148	-6.0000E+01	.2660	2.1868E-04
149	-4.5000E+01	.2856	4.8107E-04
150	-3.0000E+01	.3157	1.4615E-03
151	-1.5000E+01	.3705	8.6547E-03
152	.0000E+00	.3950	1.7600E-02

VOLUME BALANCE CALCULATIONS

TOP FLUX 2.5469E-07
BOTTOM FLUX 7.0621E-09
STORAGE RATE 2.4756E-07

ERROR	7.6461E-11

VOLUME IN	1.9357E-02
VOLUME OUT	5.3672E-04
STORAGE VOLUME	1.8814E-02
ERROR	5.8108E-06

CUMULATIVE CHANGES

VOLUME IN (-)	2.3130E+01
VOLUME OUT	7.2775E-01
STORAGE	2.1579E+01
ERROR	8.2387E-01
RELATIVE ERROR	.038180

****

TIME(YRS)= 3.00 TIME STEP= 186 DT= 7.600E+04 ERR= 8.284E-02

ITER= 1 TOTAL K ITERATIONS= 1001

PARTICLE DEPTH IN LINER= -6.215E+01

****

NODE	POTENTIAL	MOISTURE	K
1	1.0000E+02	.4950	1.0000E-07
2	9.9892E+01	.4950	1.0000E-07
3	9.9783E+01	.4950	1.0000E-07
4	9.9675E+01	.4950	1.0000E-07
5	9.9567E+01	.4950	1.0000E-07
6	9.9458E+01	.4950	1.0000E-07
7	9.9350E+01	.4950	1.0000E-07
8	9.9241E+01	.4950	1.0000E-07
9	9.9133E+01	.4950	1.0000E-07
10	9.9025E+01	.4950	1.0000E-07
11	9.8916E+01	.4950	1.0000E-07
12	9.8374E+01	.4950	1.0000E-07

13	9.7833E+01	.4950	1.0000E-07
14	9.7291E+01	.4950	1.0000E-07
15	9.6749E+01	.4950	1.0000E-07
16	9.6207E+01	.4950	1.0000E-07
17	9.5665E+01	.4950	1.0000E-07
18	9.5123E+01	.4950	1.0000E-07
19	9.4582E+01	.4950	1.0000E-07
20	9.4040E+01	.4950	1.0000E-07
21	9.3498E+01	.4950	1.0000E-07
22	9.2414E+01	.4950	1.0000E-07
23	9.1331E+01	.4950	1.0000E-07
24	9.0247E+01	.4950	1.0000E-07
25	8.9163E+01	.4950	1.0000E-07
26	8.8080E+01	.4950	1.0000E-07
27	8.6996E+01	.4950	1.0000E-07
28	8.5912E+01	.4950	1.0000E-07
29	8.4829E+01	.4950	1.0000E-07
30	8.3745E+01	.4950	1.0000E-07
31	8.2661E+01	.4950	1.0000E-07
32	8.1036E+01	.4950	1.0000E-07
33	7.9410E+01	.4950	1.0000E-07
34	7.7785E+01	.4950	1.0000E-07
35	7.6159E+01	.4950	1.0000E-07
36	7.4534E+01	.4950	1.0000E-07
37	7.2908E+01	.4950	1.0000E-07
38	7.1283E+01	.4950	1.0000E-07
39	6.9657E+01	.4950	1.0000E-07
40	6.8032E+01	.4950	1.0000E-07
41	6.6406E+01	.4950	1.0000E-07
42	6.4239E+01	.4950	1.0000E-07
43	6.2072E+01	.4950	1.0000E-07
44	5.9904E+01	.4950	1.0000E-07
45	5.7737E+01	.4950	1.0000E-07
46	5.5569E+01	.4950	1.0000E-07
47	5.3402E+01	.4950	1.0000E-07
48	5.1235E+01	.4950	1.0000E-07
49	4.9067E+01	.4950	1.0000E-07
50	4.6900E+01	.4950	1.0000E-07
51	4.4733E+01	.4950	1.0000E-07
52	4.2565E+01	.4950	1.0000E-07
53	4.0398E+01	.4950	1.0000E-07
54	3.8231E+01	.4950	1.0000E-07
55	3.6063E+01	.4950	1.0000E-07
56	3.3896E+01	.4450	4.1952E-05
57	3.8896E+01	.3950	1.7600E-02
58	4.3896E+01	.3950	1.7600E-02
59	4.8896E+01	.3950	1.7600E-02
60	5.3896E+01	.3950	1.7600E-02
61	5.8896E+01	.3950	1.7600E-02
62	6.3896E+01	.4450	4.1952E-05
63	6.3787E+01	.4950	1.0000E-07
64	6.3679E+01	.4950	1.0000E-07

65	6.3571E+01	.4950	1.0000E-07
66	6.3462E+01	.4950	1.0000E-07
67	6.3354E+01	.4950	1.0000E-07
68	6.3246E+01	.4950	1.0000E-07
69	6.3137E+01	.4950	1.0000E-07
70	6.3029E+01	.4950	1.0000E-07
71	6.2920E+01	.4950	1.0000E-07
72	6.2812E+01	.4950	1.0000E-07
73	6.2270E+01	.4950	1.0000E-07
74	6.1729E+01	.4950	1.0000E-07
75	6.1187E+01	.4950	1.0000E-07
76	6.0645E+01	.4950	1.0000E-07
77	6.0103E+01	.4950	1.0000E-07
78	5.9561E+01	.4950	1.0000E-07
79	5.9020E+01	.4950	1.0000E-07
80	5.8478E+01	.4950	1.0000E-07
81	5.7936E+01	.4950	1.0000E-07
82	5.7394E+01	.4950	1.0000E-07
83	5.6311E+01	.4950	1.0000E-07
84	5.5227E+01	.4950	1.0000E-07
85	5.4143E+01	.4950	1.0000E-07
86	5.3060E+01	.4950	1.0000E-07
87	5.1976E+01	.4950	1.0000E-07
88	5.0893E+01	.4950	1.0000E-07
89	4.9809E+01	.4950	1.0000E-07
90	4.8725E+01	.4950	1.0000E-07
91	4.7642E+01	.4950	1.0000E-07
92	4.6558E+01	.4950	1.0000E-07
93	4.4933E+01	.4950	1.0000E-07
94	4.3307E+01	.4950	1.0000E-07
95	4.1682E+01	.4950	1.0000E-07
96	4.0057E+01	.4950	1.0000E-07
97	3.8431E+01	.4950	1.0000E-07
98	3.6806E+01	.4950	1.0000E-07
99	3.5181E+01	.4950	1.0000E-07
100	3.3555E+01	.4950	1.0000E-07
101	3.1930E+01	.4950	1.0000E-07
102	3.0304E+01	.4950	1.0000E-07
103	2.8137E+01	.4950	1.0000E-07
104	2.5970E+01	.4950	1.0000E-07
105	2.3803E+01	.4950	1.0000E-07
106	2.1636E+01	.4950	1.0000E-07
107	1.9469E+01	.4950	1.0000E-07
108	1.7301E+01	.4950	1.0000E-07
109	1.5134E+01	.4950	1.0000E-07
110	1.2967E+01	.4950	1.0000E-07
111	1.0800E+01	.4950	1.0000E-07
112	8.6326E+00	.4950	1.0000E-07
113	6.4654E+00	.4950	1.0000E-07
114	4.2983E+00	.4950	1.0000E-07
115	2.1311E+00	.4950	1.0000E-07
116	-3.6094E-02	.4950	1.0000E-07

117	-2.2758E+00	.4948	9.6674E-08
118	-4.7508E+00	.4921	8.8764E-08
119	-7.6574E+00	.4866	7.7241E-08
120	-1.1286E+01	.4785	6.3074E-08
121	-1.6099E+01	.4673	4.7670E-08
122	-2.2860E+01	.4524	3.2873E-08
123	-3.2825E+01	.4329	2.0517E-08
124	-4.7880E+01	.4087	1.1686E-08
125	-7.0218E+01	.3813	6.2960E-09
126	-1.0086E+02	.3540	3.4185E-09
127	-1.3767E+02	.3306	1.9999E-09
128	-1.7574E+02	.3127	1.3074E-09
129	-2.1080E+02	.2999	9.5084E-10
130	-2.4176E+02	.2906	7.4761E-10
131	-2.7027E+02	.2832	6.1454E-10
132	-2.9974E+02	.2277	3.6927E-08
133	-2.8479E+02	.1811	3.0630E-06
134	-2.6983E+02	.1835	3.5512E-06
135	-2.5486E+02	.1861	4.1524E-06
136	-2.3989E+02	.1889	4.9018E-06
137	-2.2491E+02	.1920	5.8489E-06
138	-2.0993E+02	.1953	7.0650E-06
139	-1.9495E+02	.1989	8.6546E-06
140	-1.7996E+02	.2028	1.0776E-05
141	-1.6497E+02	.2072	1.3676E-05
142	-1.4998E+02	.2122	1.7757E-05
143	-1.3499E+02	.2178	2.3699E-05
144	-1.1999E+02	.2242	3.2725E-05
145	-1.0500E+02	.2317	4.7183E-05
146	-8.9997E+01	.2407	7.1985E-05
147	-7.4999E+01	.2518	1.1864E-04
148	-5.9999E+01	.2660	2.1868E-04
149	-4.5000E+01	.2856	4.8107E-04
150	-3.0000E+01	.3157	1.4615E-03
151	-1.5000E+01	.3705	8.6547E-03
152	.0000E+00	.3950	1.7600E-02

VOLUME BALANCE CALCULATIONS

TOP FLUX 2.0837E-07
BOTTOM FLUX 8.6315E-09
STORAGE RATE 1.9916E-07

ERROR 5.8116E-10

VOLUME IN 1.5836E-02
VOLUME OUT 6.5599E-04
STORAGE VOLUME 1.5136E-02

ERROR 4.4168E-05

CUMULATIVE CHANGES

VOLUME IN (-)	3.0275E+01
VOLUME OUT	9.3159E-01
STORAGE	2.8348E+01

ERROR	9.9480E-01

RELATIVE ERROR .035092

****

STEADY STATE ACHIEVED DURING TIME STEP 228 YEAR = 3.81E+00

****

****

TIME(YRS)= 3.81 TIME STEP= 228 DT= 6.050E+05 ERR= 5.021E-02

ITER= 2 TOTAL K ITERATIONS= 1127

PARTICLE DEPTH IN LINER= -7.483E+01

****

NODE	POTENTIAL	MOISTURE	K
1	1.0000E+02	.4950	1.0000E-07
2	9.9903E+01	.4950	1.0000E-07
3	9.9806E+01	.4950	1.0000E-07
4	9.9709E+01	.4950	1.0000E-07
5	9.9612E+01	.4950	1.0000E-07
6	9.9515E+01	.4950	1.0000E-07
7	9.9417E+01	.4950	1.0000E-07
8	9.9320E+01	.4950	1.0000E-07
9	9.9223E+01	.4950	1.0000E-07
10	9.9126E+01	.4950	1.0000E-07
11	9.9029E+01	.4950	1.0000E-07
12	9.8544E+01	.4950	1.0000E-07
13	9.8058E+01	.4950	1.0000E-07
14	9.7573E+01	.4950	1.0000E-07
15	9.7087E+01	.4950	1.0000E-07
16	9.6602E+01	.4950	1.0000E-07
17	9.6116E+01	.4950	1.0000E-07
18	9.5631E+01	.4950	1.0000E-07

19	9.5145E+01	.4950	1.0000E-07
20	9.4660E+01	.4950	1.0000E-07
21	9.4174E+01	.4950	1.0000E-07
22	9.3203E+01	.4950	1.0000E-07
23	9.2233E+01	.4950	1.0000E-07
24	9.1262E+01	.4950	1.0000E-07
25	9.0291E+01	.4950	1.0000E-07
26	8.9320E+01	.4950	1.0000E-07
27	8.8349E+01	.4950	1.0000E-07
28	8.7378E+01	.4950	1.0000E-07
29	8.6407E+01	.4950	1.0000E-07
30	8.5436E+01	.4950	1.0000E-07
31	8.4465E+01	.4950	1.0000E-07
32	8.3009E+01	.4950	1.0000E-07
33	8.1552E+01	.4950	1.0000E-07
34	8.0096E+01	.4950	1.0000E-07
35	7.8640E+01	.4950	1.0000E-07
36	7.7183E+01	.4950	1.0000E-07
37	7.5727E+01	.4950	1.0000E-07
38	7.4270E+01	.4950	1.0000E-07
39	7.2814E+01	.4950	1.0000E-07
40	7.1358E+01	.4950	1.0000E-07
41	6.9901E+01	.4950	1.0000E-07
42	6.7959E+01	.4950	1.0000E-07
43	6.6017E+01	.4950	1.0000E-07
44	6.4076E+01	.4950	1.0000E-07
45	6.2134E+01	.4950	1.0000E-07
46	6.0192E+01	.4950	1.0000E-07
47	5.8250E+01	.4950	1.0000E-07
48	5.6308E+01	.4950	1.0000E-07
49	5.4366E+01	.4950	1.0000E-07
50	5.2424E+01	.4950	1.0000E-07
51	5.0482E+01	.4950	1.0000E-07
52	4.8541E+01	.4950	1.0000E-07
53	4.6599E+01	.4950	1.0000E-07
54	4.4657E+01	.4950	1.0000E-07
55	4.2715E+01	.4950	1.0000E-07
56	4.0773E+01	.4450	4.1952E-05
57	4.5773E+01	.3950	1.7600E-02
58	5.0773E+01	.3950	1.7600E-02
59	5.5773E+01	.3950	1.7600E-02
60	6.0773E+01	.3950	1.7600E-02
61	6.5773E+01	.3950	1.7600E-02
62	7.0773E+01	.4450	4.1952E-05
63	7.0676E+01	.4950	1.0000E-07
64	7.0579E+01	.4950	1.0000E-07
65	7.0482E+01	.4950	1.0000E-07
66	7.0384E+01	.4950	1.0000E-07
67	7.0287E+01	.4950	1.0000E-07
68	7.0190E+01	.4950	1.0000E-07
69	7.0093E+01	.4950	1.0000E-07
70	6.9996E+01	.4950	1.0000E-07

71	6.9899E+01	.4950	1.0000E-07
72	6.9802E+01	.4950	1.0000E-07
73	6.9317E+01	.4950	1.0000E-07
74	6.8831E+01	.4950	1.0000E-07
75	6.8346E+01	.4950	1.0000E-07
76	6.7860E+01	.4950	1.0000E-07
77	6.7375E+01	.4950	1.0000E-07
78	6.6889E+01	.4950	1.0000E-07
79	6.6404E+01	.4950	1.0000E-07
80	6.5918E+01	.4950	1.0000E-07
81	6.5433E+01	.4950	1.0000E-07
82	6.4948E+01	.4950	1.0000E-07
83	6.3977E+01	.4950	1.0000E-07
84	6.3006E+01	.4950	1.0000E-07
85	6.2035E+01	.4950	1.0000E-07
86	6.1064E+01	.4950	1.0000E-07
87	6.0093E+01	.4950	1.0000E-07
88	5.9122E+01	.4950	1.0000E-07
89	5.8151E+01	.4950	1.0000E-07
90	5.7180E+01	.4950	1.0000E-07
91	5.6210E+01	.4950	1.0000E-07
92	5.5239E+01	.4950	1.0000E-07
93	5.3782E+01	.4950	1.0000E-07
94	5.2326E+01	.4950	1.0000E-07
95	5.0870E+01	.4950	1.0000E-07
96	4.9413E+01	.4950	1.0000E-07
97	4.7957E+01	.4950	1.0000E-07
98	4.6501E+01	.4950	1.0000E-07
99	4.5044E+01	.4950	1.0000E-07
100	4.3588E+01	.4950	1.0000E-07
101	4.2132E+01	.4950	1.0000E-07
102	4.0675E+01	.4950	1.0000E-07
103	3.8734E+01	.4950	1.0000E-07
104	3.6792E+01	.4950	1.0000E-07
105	3.4850E+01	.4950	1.0000E-07
106	3.2908E+01	.4950	1.0000E-07
107	3.0966E+01	.4950	1.0000E-07
108	2.9025E+01	.4950	1.0000E-07
109	2.7083E+01	.4950	1.0000E-07
110	2.5141E+01	.4950	1.0000E-07
111	2.3199E+01	.4950	1.0000E-07
112	2.1258E+01	.4950	1.0000E-07
113	1.9316E+01	.4950	1.0000E-07
114	1.7374E+01	.4950	1.0000E-07
115	1.5432E+01	.4950	1.0000E-07
116	1.3490E+01	.4950	1.0000E-07
117	1.1549E+01	.4950	1.0000E-07
118	9.6069E+00	.4950	1.0000E-07
119	7.6651E+00	.4950	1.0000E-07
120	5.7233E+00	.4950	1.0000E-07
121	3.7815E+00	.4950	1.0000E-07
122	1.8397E+00	.4950	1.0000E-07

123	-1.0203E-01	.4950	1.0000E-07
124	-2.1028E+00	.4948	9.7096E-08
125	-4.3082E+00	.4927	9.0378E-08
126	-6.9335E+00	.4881	8.0182E-08
127	-1.0320E+01	.4807	6.6680E-08
128	-1.5093E+01	.4696	5.0522E-08
129	-2.2658E+01	.4528	3.3221E-08
130	-3.6972E+01	.4257	1.7295E-08
131	-7.4002E+01	.3773	5.7697E-09
132	-2.9433E+02	.2287	3.8471E-08
133	-2.8031E+02	.1818	3.1991E-06
134	-2.6615E+02	.1841	3.6871E-06
135	-2.5189E+02	.1867	4.2881E-06
136	-2.3751E+02	.1894	5.0374E-06
137	-2.2304E+02	.1924	5.9844E-06
138	-2.0848E+02	.1956	7.2003E-06
139	-1.9385E+02	.1991	8.7897E-06
140	-1.7915E+02	.2031	1.0911E-05
141	-1.6438E+02	.2074	1.3811E-05
142	-1.4957E+02	.2123	1.7891E-05
143	-1.3471E+02	.2179	2.3833E-05
144	-1.1981E+02	.2243	3.2859E-05
145	-1.0489E+02	.2318	4.7316E-05
146	-8.9937E+01	.2407	7.2117E-05
147	-7.4968E+01	.2518	1.1877E-04
148	-5.9987E+01	.2660	2.1881E-04
149	-4.4995E+01	.2856	4.8120E-04
150	-2.9999E+01	.3157	1.4616E-03
151	-1.5000E+01	.3705	8.6549E-03
152	.0000E+00	.3950	1.7600E-02

VOLUME BALANCE CALCULATIONS

TOP FLUX 1.9709E-07
BOTTOM FLUX 1.9382E-07
STORAGE RATE 3.3105E-09

ERROR 3.6360E-11

VOLUME IN 1.1924E-01
VOLUME OUT 1.1726E-01
STORAGE VOLUME 2.0028E-03

ERROR 2.2002E-05

CUMULATIVE CHANGES

VOLUME IN (-) 3.5372E+01
VOLUME OUT 3.5412E+00

```
STORAGE                   3.0782E+01
                          ------------
ERROR                     1.0486E+00

RELATIVE ERROR               .034066

************************************************************************

BREAKTHROUGH OCCURRED DURING TIME STEP   656     YEAR =   1.20E+01

************************************************************************


EXECUTION ENDED NORMALLY -     0 - WARNINGS
```

APPENDIX F

SUBROUTINE DESCRIPTION

This appendix provides a brief description of the subroutines called from the SOILINER main program. Figure F-1 summarizes the order in which subroutines are called. The main program first calculates the steady state PSI distribution as a means of comparison to determine the time at which steady state is achieved during transient simulations. Then, a new PSI distribution is determined for each time step based on the specified initial conditions. At the end of each time step the program (1) checks to see if steady state has been achieved and (2) advances the particle. After steady state is reached, the program continues to advance the particle until breakthrough occurs, at which time the program terminates. The following subroutine descriptions are presented in the order shown in Figure F-1.

FLOPEN - This subroutine is called for the purpose of opening files and assigning the unit numbers to be associated with the desired input and output files.

INPUT - All input is read through this subroutine including control, grid design, soil properties, and initial conditions data. Subroutine NEWBC is called from INPUT to set the pressure boundary conditions. All input data is "echoed" to the general output file if LPRINT set equal to T.

CLEAR - The first two calls to this subroutine fill arrays OLDC and VOLD (the storage array and velocity vector at t^{n-1}, respectively) with zeros. The third call also zeros the C (storage vector at t^n) array, which enables SOILINER to utilize the same algorithm for both the transient and steady state solution when solving the set of simultaneous liner equations for the unknown DPSI. The difference between the two solution strategies is that for steady state there does not exist either a previous time step or a storage term. Thus OLDC, VOLDC, and C are eliminated by setting each array to zero. Note that the C array is zeroed each iteration since it is calculated in subroutine SPROP.

SPROP - This subroutine is called a number of times for both the steady state and transient solution strategies. It is first utilized in the steady state algorithm within the k-iteration loop. Here, the unsaturated soil properties SATK (hydraulic conductivity), RMOIST (moisture content), and C (moisture capacity) are updated using characteristic curves to reflect the newly calculated PSI distribution. Note that the first time SPROP is called, soil properties are updated with respect to the specified initial conditions. When convergence occurs, SPROP is called to update properties consistent with the final PSI distribution. Finally, SPROP is called to re-establish initial soil properties for the transient solution. Subsequent calls to SPROP during a transient simulation serve the identical function described above for steady state.

BUILDV - This subroutine computes the velocity vector held in the array VNEW (see equation 3-10 of Part I). For both the steady state and transient solutions VNEW is incorporated into the forcing vector, F (see the discussion on BUILDF below). Also, BUILDV is re-initialized at the beginning of the transient solution, based on the specified initial conditions of PSI, for subroutine PREDCT.

BUILDF - The right-hand side (RHS) of equation 3-19 (Part I) constitutes the forcing vector F, of the matrix equation solved using the Thomas algorithm for the unknown values DPSI. BUILDF builds the forcing vector which includes VNEW and all other knowns from the RHS of 3-19.

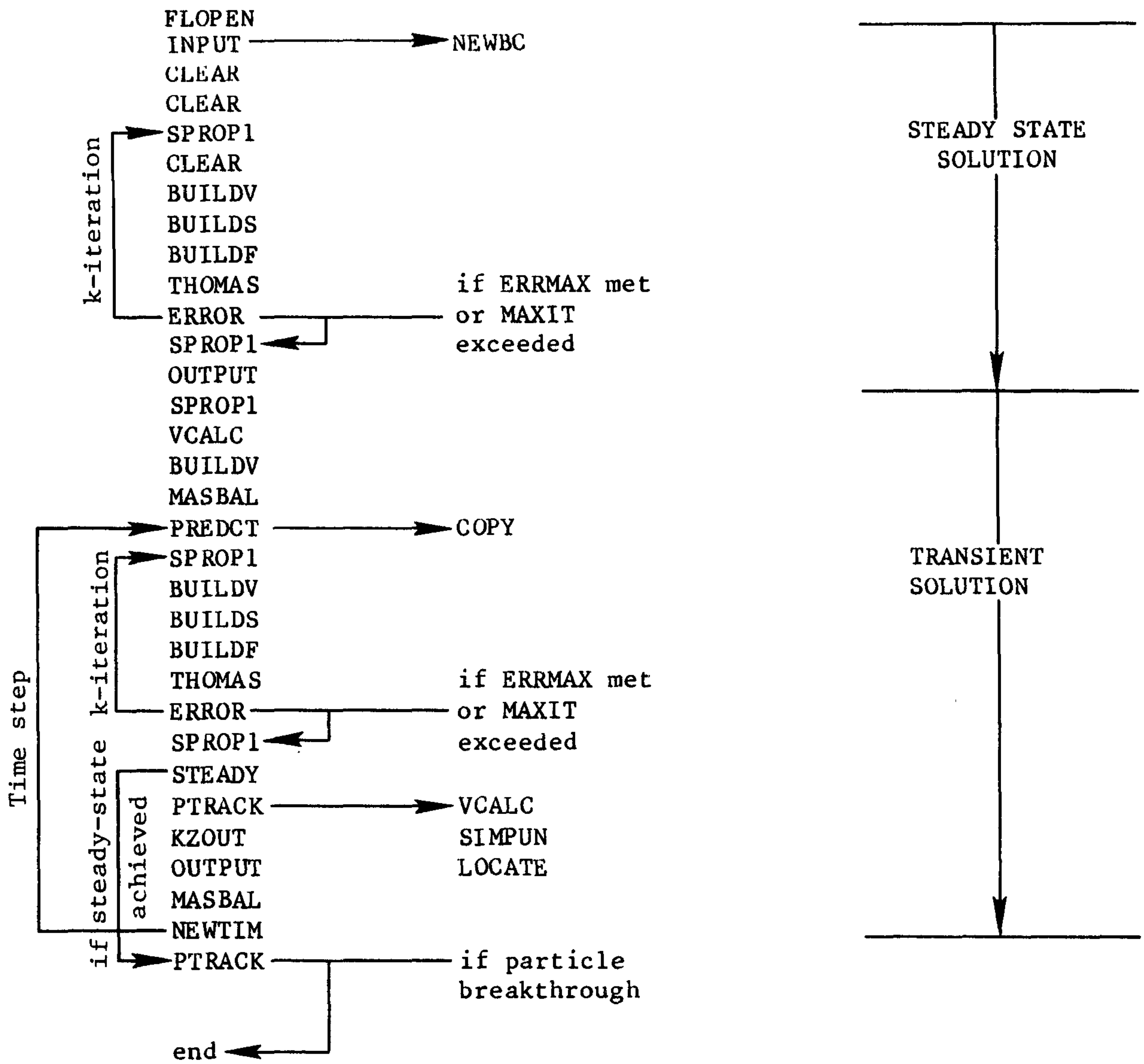

Figure F-1. Sequence of SUBROUTINE calls.

BUILDS - The LHS of equation 3-19 contains the unknown values DPSI for a given node i and it's two surrounding nodes i-1 and i+1. The LHS also contains known coefficients which constitute the stiffness matrix STIFF, a two-dimensional array created by BUILDS. Values assigned to the elements of STIFF depend upon the solution strategy being evaluated. For the transient solution, all diagonals of the tridiagonal matrix being represented by STIFF contain conductivity and grid geometry data. The center diagonal contains additional data from the storage array, C, and the time step size. For steady state, all three diagonals contain only grid and conductivity data since the storage vector, C, is zero for this solution strategy. Once updated, both STIFF and F are passed to subroutine THOMAS for each k-iteration shown in Figure F-1.

THOMAS - The Thomas algorithm solves the tridiagonal matrix equation containing STIFF, F, and the vector of unknowns, DPSI. THOMAS solves for the unknown values of DPSI, which are passed back to the main program. Once DPSI is known, PSI is updated thus completing a k-iteration. Another k-iteration is initiated unless the error for convergence, ERRMAX, is met (see discussion on subroutine ERROR below), or the maximum number of specified k-iterations, MAXIT, is exceeded.

ERROR - As long as MAXIT is not exceeded, SOILINER will continue to iterate during any given time step until convergence is achieved. The convergence criterion is tested at the end of each k-iteration by passing the array DPSI to subroutine ERROR. DPSI contains the nodal values of $\Delta\psi_i$ (see equation 3-19) which are used to update each corresponding value of ψ_i. If the solution is converging at a given time step, the $\Delta\psi_i$ values will continue to decrease. Subroutine ERROR "scans" the DPSI array to find the largest, absolute value (CHMAX) of DPSI after iteration. If CHMAX is less than the specified error for convergence (ERRMAX), the iterative procedure is terminated and the steady state solution is obtained. For transient simulations, when ERRMAX is achieved at a given time level, a new time step (DT) is determined and the iterative procedure is initiated again. For either solution strategy, if the maximum number of iterations specified per time step (MAXIT) is exceeded, a forced exit from the k-iteration loop occurs.

OUTPUT - Depending on the solution strategy chosen, subroutine OUTPUT may be called to write the steady state solution or initial conditions and subsequent output for the specified special output times. In either case, output consists of: (1) time-level data including the specified time (TIME), time step (NT), time step size (DT), maximum calculated DPSI (ERR), the number of iterations for the given time step (ITER), the total number of k-iterations (NKITER) as of time t, and the particle depth (TZDIST); (2) node data, including pressure, moisture, and conductivity (PSI, RMOIST, and RK, respectively); and (3) mass balance calculations.

VCALC - The transient solution begins when the soil properties are re-established with respect to the specified initial PSI distribution. After the call to SPROP, the initial particle velocity is determined by passing soil properties and grid data to subroutine VCALC. Equations 3-25 and 3-26 of Part I are solved for the element flux and velocity respectively in VCALC. Data required to solve these equations include the conductivity for the element in which the particle is located, and ψ and θ values from the two surrounding nodes. For the first call to VCALC, an initial velocity is returned to the main program in the array VELO. After the initial velocity calculation, VCALC will be called at the end of each time step. However, the array VELO will never contain more than 3 velocity values since it is updated after each call to SIMPUN.

MASBAL - SOILINER provides a mass balance for moisture based upon the calculated fluxes entering and leaving the flow domain and the change in storage. Ideally, flux entering the system should equal the rate of storage plus flux leaving the system. Any difference in this relationship is the error. MASBAL calculates the flux and storage rates, their corresponding volumes (per square unit), and the error associated with each time step (note that the first call to MASBAL is used to initialize variables for subsequent mass balance calculations). MASBAL also calculates cumulative changes in flux and storage.

PREDCT - Step one in the finite-difference approximation of the transient, unsaturated flow equation (see 2-1a, Part I) is to predict the new pressure distribution at time $t^{n+1} = t^n + \Delta t$, based on the soil properties at t^n. Subroutine PREDCT contains equation 3-13 of Part I with ALPHA set equal to zero, thus making the prediction fully explicit. PREDCT makes the initial estimate of the new PSI distribution, which is subsequently refined in the k-iteration loop containing the THOMAS subroutine. The initial estimate for the nodal values of ψ^{n+1} is based on the existing (or initial) PSI distribution, velocity vector (VOLD) and storage vector (OLDC), and the time step Δt.

COPY - This subroutine is called from PREDCT prior to calculation of 3-13. The existing pressure, velocity, and storage vectors (PSI, VNEW, and C, respectively) are saved into PSIOLD, VOLD, and OLDC. For the initial time step the arrays PSI, VNEW and C are assigned values in subroutines INPUT, BUILDV, and SPROP respectively, all occurring prior to subroutine PREDCT. The COPY function is required not only for equation 3-13 of PREDCT, but also for subroutine BUILDF in which all six arrays (PSI, PSIOLD, VOLD, C, and COLD) are required.

STEADY - As a means of determining the time at which steady state is achieved during a transient simulation, SOILINER first performs the steady state algorithm. The initial PSI values are read into the array SSPSI, which ultimately contains the steady state solution. Both the PSI and SSPSI arrays are passed from the main program to subroutine STEADY where they are compared node by node. If the maximum difference in PSI of all nodes is less than 1.0 cm, steady state is assumed to have been achieved.

PTRACK - This subroutine is called at the end of each time step to: (1) calculate the particle velocity (subroutine VCALC), (2) determine the distance traveled by the particle during a given time period (subroutine SIMPUN), (3) advance the particle, and (4) determine the new particle position (subroutine LOCATE). Each time step, a new particle velocity is determined. For Simpson's Rule both SIMPUN and LOCATE are called every other time step. Subroutine PTRACK also passes the integer flag KODEBT to indicate breakthrough. Initially, the particle is positioned at the liner surface. When the particle has migrated beyond a given node (NCLAYN - usually specified at the liner/underlying site-soil interface), KODEBT is assigned the value of 1 to indicate breakthrough, and the program is terminated.

SIMPUN - The objective of this subroutine is to pass a value ZDIST, which represents the distance traveled by a particle over the period of integration, to the main program. SIMPUN provides a numerical approximation of the integral and is capable of handling unevenly spaced points in time. In order to employ the numerical integration scheme of SIMPUN, both time and corresponding velocity data are required (VTIME and VELO, respectively). The current time step is saved in VTIME before each call to VCALC. Simpson's Rule requires three velocity-time data points to determine a value of ZDIST. Once returned to the main program, ZDIST is added to the previously calculated, total distance traveled by the particle (TZDIST) since initiation of a particular SOILINER simulation.

LOCATE - Each time ZDIST is calculated, it is necessary to determine the new particle position with respect to the specified grid design. If the particle passes from one element to the next over the period of integration, new element data is required for the flux and velocity calculations of subroutine VCALC. Subroutine LOCATE "scans" the grid design by comparing successive node depths to the value TZDIST. The first node encountered having a depth that exceeds TZDIST becomes the bottom node (NODEV2) defining the element (NELEMV) of the new particle location; the upper node defining NELEMV (NODEV1) is then calculated as NODEV2-1. If during the scanning procedure the specified breakthrough node (NCLAYN) is reached and its depth does not exceed TZDIST, LOCATE sets the integer flag KODEBT equal to one, indicating breakthrough.

NEWTIM - After the new particle position is established and all output requirements are satisfied, mass balance calculations are performed (see subroutine MASBAL). If the specified simulation period or maximum number of time steps (ENDTIM and MAXNT, respectively) have not been exceeded, a new time step is then determined. The existing time step (DT), a change parameter (CHPARM), maximum allowable time step (DTMAX), and current PSI distribution are passed to subroutine NEWTIM. The first algorithm employed by NEWTIM is used to determine the maximum change in PSI at any given node (CHMAX) between successive time steps. Equation 4-1 of Part II is then solved for the new

DT. The size of DT is due in part to CHMAX. If CHMAX is large, the new DT will be small, thus preventing excessively large time steps and potential model inaccuracies. Conversely, if CHMAX is small, DT will increase thus maintaining model efficiency. Finally, as steady state is approached, CHMAX will tend to remain relatively small resulting in an ever increasing DT. Although changes in PSI between successive time steps are small, too large a DT can still lead to model inaccuracy. To eliminate the potential for extremely large time steps, a specified time step size limit (DTMAX) is set. If DT exceeds DTMAX, DT is set equal to DTMAX within NEWTIM.

APPENDIX G

SOILINER
FORTRAN SOURCE CODE

```
C*******************************************************************
C
C       SOILINER.FOR      MAIN PROGRAM
C
C       Finite difference of vertical unsaturated infiltration
C
C       Dan Goode         June 1983
C
C                         GCA/Technology Division, Inc.
C                         213 Burlington Road
C                         Bedford, MA  01730
C                         (617) 275-5444
C
C       Russ Johnson      November 1985
C       Richard Wozmak
C
C
C *****************************************************************
      COMMON/MASSB/STOR,TOTV1,TOTV2,FLUX10,FLUX20
      COMMON/DEVICE/IRD,IPRT
      COMMON/FILES/IFGRD,IFSOIL,IFINIT,IFPOUT,IFMOUT,IFLUX,IPDOUT
      COMMON/INFO/NUMNP,NUMEL,NPM1,NPM2,NUMEL2,AM1,NX,DTV,
     *          DEPTH,ENDTIM,DT,DTMAX,ALPHA,PSIBOT,H,NZOUT,SRPARM,
     *          PSINIT,NCPTS,ERRMAX,MAXIT,CHPARM,MAXNT,SYEAR
      COMMON/TIMES/TIME,TIME1,NT,ERR,NOUT,TOUT(10),NOUT1,TOUT1
      COMMON/TRACK/NODEV1,NODEV2,NELEMV,KODEBT,NCLAYN,TIMEBT,OLDTD,
     *             TDIST
      COMMON/ERRORS/IERR,IWARN
      DIMENSION PSI(200),PSIOLD(200),DZ(200),C(200),
     *          RK(200),WORK(200),SSPSI(200),
     *          RMOIST(200),
     *          STIFF(3,200),DPSI(200),F(200),
     *          Z(200),SATK(200),PSICRT(200),POR(200)
      DIMENSION VNEW(200),VOLD(200),OLDC(200),DZN(200),STARK(200),
     *          IZOUT(10),ISOIL(200),BETA(200)
      DIMENSION RKL(400),CL(400),RMSTL(400),OLDML(400)
      DIMENSION VTIME(3),VELO(3),ZDIST(3)
      REAL SRPSI(200)
      CHARACTER*20 FILE1,FILE2,FILE3,FILE4
      CHARACTER*72 FILEN
      LOGICAL LSTEDY,LPRINT
      DATA NMAX/200/

      IWARN = 0
      NKITER = 0
      ITER = 0

C     Open input and output files
      CALL FLOPEN

C     Read and echo problem parameters and initialize
```

```
      CALL INPUT(PSI,SATK,POR,ISOIL,DZ,BETA,DZN,Z,
     *             PSICRT,IZOUT,LSTEDY,LPRINT)

C     Save ALPHA for transient solution; use ALPHA=1 for steady state
      ALPHAS=ALPHA

C     Save initial pressure distribution in PSI for transient run;
C     use SSPSI for steady state solution.
      DO 100 I=1,NUMNP
  100 SSPSI(I)=PSI(I)

C      ==============================
C       Steady State Solution
C      ==============================
      ALPHA=1.D0
      AM1=0.D0

      CALL CLEAR(OLDC,NUMNP)
      CALL CLEAR(VOLD,NPM2)

C      *******************
C      Main Iteration Loop
C      *******************


  120 CONTINUE
        NKITER=NKITER+1
        ITER=ITER+1

C       Calculate soil properties
        CALL SPROP(SSPSI,RMOIST,C,
     *             RK,RMSTL,CL,RKL,STARK,SATK,POR,ISOIL,PSICRT)

C       Zero storage vector
        CALL CLEAR(C,NUMNP)

C       Build velocity vector
        CALL BUILDV(VNEW,SSPSI,STARK,DZN,DZ,BETA,RK)

C       Build stiffness matrix
        CALL BUILDS(STIFF,RK,STARK,C,OLDC,DZN,DZ,BETA)

C       Build forcing vector -- boundary condition terms
        CALL BUILDF(F,C,OLDC,SSPSI,PSIOLD,VNEW,VOLD)

C       Solve matrix equation for DPSI using Thomas algorithm
        CALL THOMAS(STIFF,F,DPSI,WORK,NPM2)

C       Update PSI using the relaxation parameter SRPARM
        DO 125 I=1,NPM2
          N=I+1
          SRPSI(I)=SRPARM*DPSI(I)
```

```
          SSPSI(N)=SSPSI(N)+SRPSI(I)
  125   CONTINUE

C     Check solution for convergence
       CALL ERROR(SRPSI,NPM2,ERR)
       IF (ERR.LT.ERRMAX) GOTO 140
       IF (ITER.LE.MAXIT) GOTO 120

C      **************************
C      End of Main Iteration Loop
C      **************************

         WRITE(IPRT,2000) NT,ERR
         WRITE(*,2000) NT,ERR
         IWARN=IWARN+1

  140 CONTINUE

C     Output steady state solution if LSTEDY = .TRUE.
       IF(LSTEDY) CALL OUTPUT(SSPSI,RMOIST,RK,Z,RKL,CL,RMSTL,
      *                STARK,POR,DZ,ITER,NKITER,TDIST)

       WRITE(IPRT,1990) ITER,ERR
       WRITE(*,1990) ITER,ERR
       IF(LSTEDY) GO TO 50
       ERR=0.0
       ITER=0

C      ============================
C      End of Steady State Solution
C      ============================

C     Initialize particle tracking variables
       DO 5 I=1,3
         VELO(I)=0.0
         VTIME(I)=0.0
         ZDIST(I)=0.0
     5 CONTINUE
       VTIME(1)=TIME

C     Restore ALPHA to transient solution value
       ALPHA=ALPHAS
       AM1=1.D0-ALPHA

C     KODESS = 1 indicates steady state,
C     NTOUT is the number of time steps required to reach steady state
       KODESS=0
       NTOUT=0

C      ========================
C      Start Transient Solution
C      ========================
```

```
C     Reinitialize soil properties and output initial conditions
       CALL SPROP(PSI,RMOIST,C,
     *              RK,RMSTL,CL,RKL,STARK,SATK,POR,ISOIL,PSICRT)
       CALL OUTPUT(PSI,RMOIST,RK,Z,RKL,CL,RMSTL,STARK,POR,DZ,ITER,NKITER,
     *             TDIST)
       NTOUT=NTOUT+1

C     Calculate initial particle velocity for the top element
       WRITE(*,1900)
       IARG=1
       CALL VCALC(VELO,STARK,DZ,RMOIST,PSI,IARG)

C     Build velocity vector
       CALL BUILDV(VNEW,PSI,STARK,DZN,DZ,BETA,RK)

C     Initialize mass balance calculations
       IARG=-1
       CALL MASBAL(PSI,RMSTL,OLDML,STARK,DZ,IARG)

C      ===================
C      Begin New Time Step
C      ===================

   10 CONTINUE
         ITER=0
         NT=NT+1

C       Save old vectors and predict new PSI from V
         CALL PREDCT(PSI,PSIOLD,VNEW,VOLD,C,OLDC,
     *               OLDML,RMSTL)


C        *******************
C        MAIN ITERATION LOOP
C        *******************

         TIME=TIME1+DT

   20    CONTINUE
           NKITER=NKITER+1
           ITER=ITER+1

C         Compute soil properties
           IARG=0
           CALL SPROP(PSI,RMOIST,C,
     *                 RK,RMSTL,CL,RKL,STARK,SATK,POR,ISOIL,PSICRT)

C         Build velocity vector
           CALL BUILDV(VNEW,PSI,STARK,DZN,DZ,BETA,RK)

C         Build stiffness matrix
```

```
        CALL BUILDS(STIFF,RK,STARK,C,OLDC,DZN,DZ,BETA)

C       Build forcing vector -- transient and boundary condition terms
        CALL BUILDF(F,C,OLDC,PSI,PSIOLD,VNEW,VOLD)

C       Solve matrix equation using Thomas algorithm
C       (result is incremental change in PSI, DPSI)
        CALL THOMAS(STIFF,F,DPSI,WORK,NPM2)

C       Update PSI
        DO 25 I=1,NPM2
          N=I+1
          PSI(N)=PSI(N)+DPSI(I)
   25   CONTINUE

C     Check solution for convergence
      CALL ERROR(DPSI,NPM2,ERR)
      IF (ERR.LT.ERRMAX) GOTO 30
      IF (ITER.LE.MAXIT) GOTO 20

C     **************************
C     End of Main Iteration Loop
C     **************************

        WRITE(IPRT,2000) NT,ERR
        WRITE(*,2000) NT,ERR
        IWARN=IWARN+1

   30 CONTINUE
      IARG=1
      CALL SPROP(PSI,RMOIST,C,
     *           RK,RMSTL,CL,RKL,STARK,SATK,POR,ISOIL,PSICRT)

C     Compare transient PSI with steady state SSPSI
      CALL STEADY(PSI,SSPSI,KODESS)

C     Track particle and check for breakthrough, steady state
      CALL PTRACK(STARK,DZ,RMOIST,PSI,Z,VTIME,VELO,ZDIST,ITER,NKITER)
      IF (KODEBT.EQ.1) GO TO 34
      IF(KODESS.EQ.1) GO TO 40

C     Write to output files
   34 CONTINUE
      IF (IOUT.EQ.1.AND.NOUT.NE.-1) THEN
        NOUT1=NOUT1+1
        TOUT1=TOUT(NOUT1)
        IF(NOUT1.GT.NOUT.OR.TOUT1.LE.0.D0) NOUT=0
      ENDIF
      IF (NOUT.EQ.-1.OR.TIME.GE.ENDTIM.OR.NT.GE.MAXNT) IOUT=1
      IF (IOUT.EQ.1) THEN
        CALL OUTPUT(PSI,RMOIST,RK,Z,RKL,CL,RMSTL,STARK,POR,
     *              DZ,ITER,NKITER,TDIST)
```

```
          NTOUT=NTOUT+1
         ENDIF

C        Calculate mass balance of flux
          CALL MASBAL(PSI,RMSTL,OLDML,STARK,DZ,IOUT)

          IF (TIME.GE.ENDTIM.OR.NT.GE.MAXNT) GOTO 50

C        Step ahead in time
          TIME1=TIME
          IOUT=0
          CALL NEWTIM(PSI,PSIOLD,IOUT)
        GOTO 10

C       ======================
C       End of Solution Period
C       ======================

C      Particle tracking after steady state
   40 CONTINUE
        CALL OUTPUT(PSI,RMOIST,RK,Z,RKL,CL,RMSTL,STARK,POR,DZ,ITER,
     *             NKITER,TDIST)
        NTOUT=NTOUT+1
        IOUT=1
        CALL MASBAL(PSI,RMSTL,OLDML,STARK,DZ,IOUT)
   45 CONTINUE
          TIME=TIME+DTMAX
          NT=NT+1
          IF(KODEBT.EQ.1) GO TO 50
          CALL PTRACK(STARK,DZ,RMOIST,PSI,Z,VTIME,VELO,ZDIST,ITER,NKITER)
        GO TO 45

C      End of execution
   50 CONTINUE
        WRITE(IPRT,2010) IWARN
        WRITE(*,2011) IWARN
        STOP ''

 1000 FORMAT(/I5/)
 1900 FORMAT(3X,'T-STEP',4X,'DT(S)',5X,'TIME(Y)',5X,'ZDIST',6X,'TDIST',4
     *X,'NELEMV',2X,'ITER',2X,'NKITER'/)
 1990 FORMAT(//72(1H*)/1X,'STEADY STATE ALGORITHM TOOK',I5,' ITERATIONS,
     * ERROR= ',1PE10.4/72(1H*)//)
 2000 FORMAT(//' **** WARNING **** '//' MAXIMUM ITERATIONS EXCEEDED,  ',
     *'TIME STEP ',I5,'   ERR= ',1PE12.3)
 2010 FORMAT(/' EXECUTION ENDED NORMALLY - ',I5,' - WARNINGS')
 2011 FORMAT(//' Execution ended normally - ',I5,' warnings.'//)

        END
```

```
C*****************************************************************
C
C       SOILINER            SUBPROGRAMS
C
C       Dan Goode           June 1983
C
C                           GCA/Technology Division, Inc.
C                           213 Burlington Road
C                           Bedford, MA  01730
C                           (617) 275-5444
C
C       Russ Johnson        November 1985
C       Richard Wozmak
C
C*****************************************************************
C*****************************************************************
C
      SUBROUTINE FLOPEN
C
C*****************************************************************
C
C     Assigns file unit specifiers (fills DEVICE and FILES commons)
C     and opens all i/o files.
C
C*****************************************************************
      COMMON/DEVICE/IRD,IPRT
      COMMON/FILES/IFGRD,IFSOIL,IFINIT,IFPOUT,IFMOUT,IFLUX,IPDOUT
      CHARACTER*72 FILEN

C    Assign file unit specifiers
      IRD = 5
      IPRT = 7
      IFGRD = 10
      IFSOIL = 11
      IFINIT = 12
      IFPOUT = 13
      IFMOUT = 14
      IFLUX = 15
      IPDOUT = 16
      INFN = 17

C    Open all output files
      OPEN(IFMOUT,FILE='MOIST.PRN',STATUS='NEW')
      OPEN(IFPOUT,FILE='PSI.PRN',STATUS='NEW')
      OPEN(IFLUX,FILE='FLUX.OUT',STATUS='NEW')
      OPEN(IPDOUT,FILE='PDT.OUT',STATUS='NEW')
      OPEN(IPRT,FILE='GEN.OUT',STATUS='NEW')

C    Open INFN.DAT, read input file names, and open them
      OPEN(INFN,FILE='INFN.DAT')

      READ(INFN,201) FILEN
```

```
      OPEN(IRD,FILE=FILEN,STATUS='OLD')
      READ(INFN,201) FILEN
      OPEN(IFGRD,FILE=FILEN,STATUS='OLD')
      READ(INFN,201) FILEN
      OPEN(IFSOIL,FILE=FILEN,STATUS='OLD')
      READ(INFN,201) FILEN
      OPEN(IFINIT,FILE=FILEN,STATUS='OLD')

  201 FORMAT(A72)
      RETURN
      END
C*********************************************************************
C
      SUBROUTINE INPUT(PSI,SATK,POR,ISOIL,
     *                 DZ,BETA,DZN,Z,PSICRT,IZOUT,LSTEDY,LPRINT)
C
C*********************************************************************
C
C     Read control parameters, grid specification, soil properties,
C     and initial conditions.  If LPRINT = T, echo all input files
C     to general output file (GEN.OUT), else echo control file only.
C
C*********************************************************************
      DIMENSION PSI(1),SATK(1),
     *          POR(1),ISOIL(1),DZ(1),BETA(1),DZN(1),Z(1),PSICRT(1),
     *          IZOUT(1)
      COMMON/MASSB/STOR,TOTV1,TOTV2,FLUX10,FLUX20
      COMMON/TIMES/TIME,TIME1,NT,ERR,NOUT,TOUT(10),NOUT1,TOUT1
      COMMON/DEVICE/IRD,IPRT
      COMMON/INFO/NUMNP,NUMEL,NPM1,NPM2,NUMEL2,AM1,NX,DTV,
     *          DEPTH,ENDTIM,DT,DTMAX,ALPHA,PSIBOT,H,NZOUT,SRPARM,
     *          PSINIT,NCPTS,ERRMAX,MAXIT,CHPARM,MAXNT,SYEAR
      COMMON/ERRORS/IERR,IWARN
      COMMON/FILES/IFGRD,IFSOIL,IFINIT,IFPOUT,IFMOUT,IFLUX,IPDOUT
      COMMON/TRACK/NODEV1,NODEV2,NELEMV,KODEBT,NCLAYN,TIMEBT,OLDTD,
     *             TDIST
      CHARACTER*80 TITLE
      INTEGER UPLIM,LOWLIM
      LOGICAL LSTEDY,LPRINT

      TIME=0.D0
      TIME1=0.D0
      SYEAR=3.1536E+7
      NT=0

      READ(IRD,628) TITLE
  628 FORMAT(A80)
      WRITE(IPRT,2001) TITLE

C     Read temporal control parameters
      READ(IRD,*) LPRINT,LSTEDY,ENDTIM,DT,DTMAX,MAXNT,ALPHA,ERRMAX,
     *            MAXIT,CHPARM
```

```
      IF(DT.LE.0.D0) DT=1.D0
      IF(ERRMAX.LE.0.D0) ERRMAX=0.05D0
      IF(MAXIT.LE.0) MAXIT=10
      AM1=1.D0-ALPHA
      WRITE(IPRT,2000) LSTEDY,ENDTIM,DT,DTMAX,MAXNT,ALPHA,ERRMAX,
     *                 MAXIT,CHPARM
      IF(MAXIT.GT.200) MAXIT=200
      ENDTIM=ENDTIM*SYEAR
      READ(IRD,*) NOUT
      IF(NOUT.GT.4) NOUT=4
      IF(LSTEDY) THEN
        NTOUT=1
      ELSE
        NTOUT=NOUT+2
      ENDIF
      WRITE(IFPOUT,1000) NTOUT
      WRITE(IFMOUT,1000) NTOUT
      WRITE(IPRT,2010) NOUT

C     Read special output times, convert to years, and assign TOUT1
      IF (NOUT.GT.0) THEN
        READ(IRD,*) (TOUT(IT),IT=1,NOUT)
        WRITE(IPRT,2060) (TOUT(IT),IT=1,NOUT)
        DO 11 I=1,NOUT
           TOUT(I)=TOUT(I)*SYEAR
   11   CONTINUE
        DO 18 IT=1,NOUT
          NOUT1=IT
          IF(TOUT(IT).GT.DT) GOTO 17
   18   CONTINUE
   17   TOUT1=TOUT(NOUT1)
      ENDIF

C     Read spatial control parameters
      READ(IRD,*) NUMNP,SRPARM,NCLAYN
      NPM2=NUMNP-2
      NPM1=NUMNP-1
      NUMEL=NUMNP-1
      NUMEL2=2*NUMEL
      WRITE(IPRT,2004) NUMNP,NUMEL,SRPARM,NCLAYN

C     Read grid specification
      Z(1) = 0.D0
      LOWLIM = 2
   10 CONTINUE
        READ(IFGRD,*,END=199) NUM,DELTA
        UPLIM = LOWLIM + NUM - 1
        DO 111 J = LOWLIM, UPLIM
          Z(J) = Z(J-1) - DELTA
  111   CONTINUE
        LOWLIM = UPLIM + 1
      GOTO 10
```

```
  199 CONTINUE
      IF (UPLIM.LT.NUMNP) STOP 'LT NUMNP nodes specified in grid file'
      IF (UPLIM.GT.NUMNP) STOP 'GT NUMNP nodes specified in grid file'
      DO 48 L=1,NUMEL
        DZ(L)=Z(L+1)-Z(L)
   48 CONTINUE
      DO 49 L=1,NPM2
        DZN(L)=(Z(L+2)-Z(L))/2.D0
        BETA(L)=DZ(L+1)/DZ(L)
   49 CONTINUE

C     Echo grid data
      IF (LPRINT) THEN
        WRITE(IPRT,2091)
        WRITE(IPRT,2093)
        WRITE(IPRT,2069) 1,Z(1)
        WRITE(IPRT,2071) 1,DZ(1)
        DO 69 I=2,NPM1
          WRITE(IPRT,2069) I,Z(I),DZN(I-1)
          WRITE(IPRT,2071) I,DZ(I)
   69   CONTINUE
        WRITE(IPRT,2069) NUMNP,Z(NUMNP)
      ENDIF

C     Read soil properties
      IF (LPRINT) WRITE(IPRT,2092)
      LOWLIM = 1
  168 CONTINUE
        READ(IFSOIL,*,END=299) NUM,ISOILX,SATKX,PORX,PSICRX
        IF (PSICRX.GT.0) STOP 'PSICRT > 0'
        UPLIM = LOWLIM + NUM - 1
        DO 169 K = LOWLIM, UPLIM
          ISOIL(K) = ISOILX
          SATK(K) = SATKX
          POR(K) = PORX
          PSICRT(K) = PSICRX
          IF (LPRINT) WRITE(IPRT,2070) K,ISOILX,SATKX,PORX,PSICRX
  169   CONTINUE
        LOWLIM = UPLIM + 1
      GOTO 168
  299 CONTINUE
      IF (UPLIM.LT.NUMEL) STOP 'LT NUMEL elements in properties file'
      IF (UPLIM.GT.NUMEL) STOP 'GT NUMEL elements in properties file'

C     Read initial conditions
      IF (LPRINT) WRITE(IPRT,2089)
      LOWLIM = 1
   54 CONTINUE
        READ(IFINIT,*,END=399) NUM,PSIQ
        UPLIM = LOWLIM + NUM - 1
        DO 56 J = LOWLIM, UPLIM
          PSI(J) = PSIQ
```

```
          IF (LPRINT) WRITE(IPRT,2090)  J,PSIQ
   56    CONTINUE
         LOWLIM = UPLIM + 1
      GOTO 54
  399 CONTINUE
      IF (UPLIM.LT.NUMNP) STOP 'LT NUMNP nodes specified in init file'
      IF (UPLIM.GT.NUMNP) STOP 'GT NUMNP nodes specified in init file'

C     Read pressure boundary conditions
   55 CALL NEWBC(PSI)

C     Initialize particle tracking parameters
      NX=3
      NODEV1=1
      NODEV2=2
      NELEMV=1
      KODEBT=0
      TDIST=0.0
      OLDTD=0.0
      DTV=0.0
      TIMEBT=0.0

      RETURN

 1000 FORMAT(/I5/)
 1001 FORMAT(18A4)
 1003 FORMAT(2I5,3F10.0,I5,2F10.0,I5,F10.0)
 1006 FORMAT(16I5)
 1010 FORMAT(10I5)
 1011 FORMAT(2I5,F10.2,I5)
 1030 FORMAT(I5)
 1040 FORMAT(3F10.0)
 1050 FORMAT(8F10.0)
 1061 FORMAT(I5,F10.0)
 1070 FORMAT(2I5,3F10.0)
 1080 FORMAT(I10,F10.0)
 1081 FORMAT(F10.0)

 2000 FORMAT(//' TEMPORAL DISCRETIZATION PARAMETERS'//
     *' STEADY STATE PARM',12(1H.),'LSTEDY=',L10/
     *'    IF LSTEDY EQ T, COMPUTE STEADY STATE ONLY'/
     *'    OTHERWISE, COMPUTE TRANSIENT SOLUTION'/
     *' SIMULATION TIME',19(1H.),'ENDTIM(YRS)=',F10.2/
     *' TIME STEP',30(1H.),'DT=',E10.3/
     *' MAXIMUM ALLOWABLE TIME STEP',10(1H.),'DTMAX=',E10.4/
     *' MAXIMUM NUMBER OF TIME STEPS',12(1H.),'MAXNT=',I10/
     *' TEMPORAL WEIGHTING PARAMETER',10(1H.),'ALPHA=',F10.2/
     *' MAXIMUM ERROR FOR CONVERGENCE',7(1H.),'ERRMAX=',F10.4/
     *' MAXIMUM ITERATIONS PER TIME STEP',6(1H.),'MAXIT=',I10/
     *' TIME STEP CHANGE PARAMETER',15(1H.),'CHPARM=',F10.4)
 2001 FORMAT(//1X,80(1H*)//' SOILINER OUTPUT'//
     *1X,80(1H*)//1X,A80//1X,80(1H*))
```

```
 2004 FORMAT(///' SPATIAL DISCRETIZATION PARAMATERS'//
     *' NUMBER OF NODE POINTS',10(1H.),'NUMNP=',I10/
     *' NUMBER OF ELEMENTS (COMPUTED)',8(1H.),'NUMEL=',I10/
     *'SUCCESSIVE RELAXATION PARAMETER',7(1H.),'SRPARM=',F10.2/
     *'NUMBER OF LINER NODES',17(1H.),'NCLAYN=',I10/
     *'(USED FOR BREAKTHROUGH DETERMINATION)')
 2006 FORMAT(/' NUMBER OF SPECIAL OUTPUT NODES',10(1H.),'NZOUT=',I10)
 2007 FORMAT(//' SPECIAL OUTPUT NODES'//
     *1X,10I10)
 2010 FORMAT(/' NUMBER OF SPECIAL OUTPUT TIMES',10(1H.),'NOUT=',I10)
 2025 FORMAT(//1X,10(1H=)/' SATURATED CONDUCTIVITY',10(1H.),'SATK=',
     *1PE10.3/1X,20(1H=)//
     *' CRITICAL POTENTIAL',10(1H.),'PSICRT=',1PE10.3/1X,18(1H=))
 2030 FORMAT(//1X,30(1H=)/' MOISTURE RETENTION AND CONDUCTIVITY CURVES'/
     *1X,30(1H=)//
     *' NUMBER OF POINTS ON CURVE',15(1H.),'NCPTS=',I10//
     *'     SUCTION        MOISTURE     RELATIVE CONDUCTIVITY'/
     *'        (L)            (-)              (-)'/)
 2040 FORMAT(3F13.3)
 2050 FORMAT(//' DATA FROM INPUT CURVES'//
     *' CRITICAL SUCTION PRESSURE',13(1H.),'PSICRT=',F10.3/
     *' EFFECTIVE POROSITY',22(1H.),'POR=',F10.4)
 2060 FORMAT(//' SPECIAL OUTPUT TIMES(YRS)'//
     *1X,8F10.2)
 2069 FORMAT(I10,2F10.3)
 2070 FORMAT(2I10,1PE10.3,0PF10.4,0PF10.3)
 2071 FORMAT(I5,F10.3)
 2089 FORMAT(//4X,'INITIAL PSI'//11X,'NODE',11X,'PSI'/)
 2090 FORMAT(4X,I10,6X,F10.2)
 2091 FORMAT(//' GRID DATA'//
     *'         NODE     Z          DZN'/)
 2092 FORMAT(//'SOIL PROPERTIES'//
     *'   ELEMENT     ISOIL   SATK          POR      PSICRT'/)
 2093 FORMAT(' ELEMENT     DZ'//)
      END
C*********************************************************************
C
      SUBROUTINE BUILDV(V,PSI,STARK,DZN,DZ,BETA,RK)
C
C*********************************************************************
C
C     Computes vector V = C*dPSI/dT
C     using centered finite difference method (CFDM).
C
C*********************************************************************
      COMMON/MASSB/STOR,TOTV1,TOTV2,FLUX10,FLUX20
      COMMON/TIMES/TIME,TIME1,NT,ERR,NOUT,TOUT(10),NOUT1,TOUT1
      COMMON/DEVICE/IRD,IPRT
      COMMON/INFO/NUMNP,NUMEL,NPM1,NPM2,NUMEL2,AM1,NX,DTV,
     *            DEPTH,ENDTIM,DT,DTMAX,ALPHA,PSIBOT,H,NZOUT,SRPARM,
     *            PSINIT,NCPTS,ERRMAX,MAXIT,CHPARM,MAXNT,SYEAR
      COMMON/ERRORS/IERR,IWARN
```

```
      DIMENSION V(1),PSI(1),STARK(1),DZN(1),DZ(1),RK(1),BETA(1)

      DO 10 L=1,NPM2
        LP1=L+1
        V(L)=( STARK(LP1) * (PSI(L+2)-PSI(LP1))/ DZ(LP1) -
     *          STARK(L) * (PSI(LP1)-PSI(L))/ DZ(L) +
     *          STARK(LP1)-STARK(L)) /DZN(L)
   10 CONTINUE
      RETURN
      END
C*****************************************************************
C
      SUBROUTINE PREDCT(PSI,PSIOLD,VNEW,VOLD,C,OLDC,OLDML,RMSTL)
C
C****************************************************************
C
C     Stores old vectors PSIOLD, VOLD, OLDC, and OLDML,
C     and predicts new PSI values.
C
C*****************************************************************
      COMMON/MASSB/STOR,TOTV1,TOTV2,FLUX10,FLUX20
      COMMON/DEVICE/IRD,IPRT
      COMMON/INFO/NUMNP,NUMEL,NPM1,NPM2,NUMEL2,AM1,NX,DTV,
     *            DEPTH,ENDTIM,DT,DTMAX,ALPHA,PSIBOT,H,NZOUT,SRPARM,
     *            PSINIT,NCPTS,ERRMAX,MAXIT,CHPARM,MAXNT,SYEAR
      COMMON/TIMES/TIME,TIME1,NT,ERR,NOUT,TOUT(10),NOUT1,TOUT1
      COMMON/ERRORS/IERR,IWARN
      DIMENSION PSI(1),PSIOLD(1),VNEW(1),VOLD(1),C(1),OLDC(1),
     *          OLDML(1),RMSTL(1)

C    Store old vectors
      CALL COPY(PSI,NUMNP,PSIOLD)
      CALL COPY(VNEW,NPM2,VOLD)
      CALL COPY(C,NUMNP,OLDC)
      CALL COPY(RMSTL,NUMEL2,OLDML)

C    Predict new PSI from V                                    .
      DO 10 I=1,NPM2
        N=I+1
        IF (OLDC(N).GT.0.D0) PSI(N)=PSI(N)+VOLD(I)/OLDC(N)*DT
   10 CONTINUE

      RETURN
      END
C****************************************************
C
      SUBROUTINE BUILDS(STIFF,RK,STARK,C,OLDC,DZN,DZ,BETA)
C
C****************************************************
C
C     Builds stiffness matrix STIFF for active nodes only.
C     This is standard finite difference.
```

```
C
C****************************************************
      COMMON/MASSB/STOR,TOTV1,TOTV2,FLUX10,FLUX20
      COMMON/DEVICE/IRD,IPRT
      COMMON/INFO/NUMNP,NUMEL,NPM1,NPM2,NUMEL2,AM1,NX,DTV,
     *            DEPTH,ENDTIM,DT,DTMAX,ALPHA,PSIBOT,H,NZOUT,SRPARM,
     *            PSINIT,NCPTS,ERRMAX,MAXIT,CHPARM,MAXNT,SYEAR
      COMMON/ERRORS/IERR,IWARN
      DIMENSION STIFF(3,1),STARK(1),C(1),OLDC(1),DZN(1),DZ(1),
     *          BETA(1),RK(1)

C     Loop over interior nodes
      DO 10 L=1,NPM2
        LP1=L+1
        ADZINV=ALPHA/DZN(L)
        STIFF(1,L)=-ADZINV*STARK(L)/DZ(L)
        STIFF(2,L)=ADZINV*(STARK(L)/DZ(L)+STARK(LP1)/DZ(LP1))+
     *             (ALPHA*C(LP1)+AM1*OLDC(LP1))/DT
        STIFF(3,L)=-ADZINV*STARK(LP1)/DZ(LP1)
   10 CONTINUE
      RETURN
      END
C
C************************************************************
C
      SUBROUTINE BUILDF(F,C,OLDC,PSI,PSIOLD,VNEW,VOLD)
C
C***********************************************************
C
C     Builds forcing vector F with
C     boundary conditions and transients.
C
C************************************************************
      COMMON/MASSB/STOR,TOTV1,TOTV2,FLUX10,FLUX20
      COMMON/DEVICE/IRD,IPRT
      COMMON/INFO/NUMNP,NUMEL,NPM1,NPM2,NUMEL2,AM1,NX,DTV,
     *            DEPTH,ENDTIM,DT,DTMAX,ALPHA,PSIBOT,H,NZOUT,SRPARM,
     *            PSINIT,NCPTS,ERRMAX,MAXIT,CHPARM,MAXNT,SYEAR
      COMMON/ERRORS/IERR,IWARN
      DIMENSION F(1),C(1),PSI(1),PSIOLD(1)
      DIMENSION VNEW(1),VOLD(1),OLDC(1)

      DO 10 L=1,NPM2
        LP1=L+1
        F(L)=ALPHA*VNEW(L) + AM1*VOLD(L) -
     *       (PSI(LP1)-PSIOLD(LP1))*(ALPHA*C(LP1)+
     *        AM1*OLDC(LP1))/DT
   10 CONTINUE
      RETURN
      END
C
C*******************************************************
```

```
C
      SUBROUTINE NEWTIM(RNEW,OLD,KODE)
C
C************************************************************
C
C     Computes new time step DT based on maximum change of PSI
C     during previous step, CHMAX.
C
C************************************************************
      COMMON/DEVICE/IRD,IPRT
      COMMON/TIMES/TIME,TIME1,NT,ERR,NOUT,TOUT(10),NOUT1,TOUT1
      COMMON/INFO/NUMNP,NUMEL,NPM1,NPM2,NUMEL2,AM1,NX,DTV,
     *            DEPTH,ENDTIM,DT,DTMAX,ALPHA,PSIBOT,H,NZOUT,SRPARM,
     *            PSINIT,NCPTS,ERRMAX,MAXIT,CHPARM,MAXNT,SYEAR
      COMMON/ERRORS/IERR,IWARN
      DIMENSION RNEW(1),OLD(1)

C    Assign absolute value of largest change in PSI to CHMAX
      CHMAX=0.D0
      DO 10 I=2,NUMEL
CDP    CHANGE NEXT CARD FOR DOUBLE/SINGLE PRECISION
        CHANGE=ABS(RNEW(I)-OLD(I))
        IF (CHANGE.GT.CHMAX) CHMAX=CHANGE
   10 CONTINUE

C    Compute new time step DT as product of old time step
C    times ratio of allowed change to actual change at last step
      DT=DT*CHPARM/CHMAX
      IF(DT.GT.DTMAX) DT=DTMAX
      IF(NOUT.LE.0) GOTO 20
        TIM=TIME1+DT
        IF (TIM.LT.TOUT1) GOTO 20
          DT=TOUT1-TIME1
          KODE=1
   20 CONTINUE
      TIM=TIME1+DT
      IF(TIM.GT.ENDTIM) DT=ENDTIM-TIME1

      RETURN
      END
C***************************************************************
C
      SUBROUTINE OUTPUT(PSI,RMOIST,RK,Z,RKL,CL,RMSTL,STARK,POR,DZ,
     *                  ITER,NKITER,TDIST)
C
C**************************************************************
C
C    Write time stepping, pressure, moisture, and conductivity data
C    to general output file.  Also write pressure, moisture, and
C    flux data to corresponding output files.
C
C**************************************************************
```

```
      COMMON/DEVICE/IRD,IPRT
      COMMON/TIMES/TIME,TIME1,NT,ERR,NOUT,TOUT(10),NOUT1,TOUT1
      COMMON/INFO/NUMNP,NUMEL,NPM1,NPM2,NUMEL2,AM1,NX,DTV,
     *            DEPTH,ENDTIM,DT,DTMAX,ALPHA,PSIBOT,H,NZOUT,SRPARM,
     *            PSINIT,NCPTS,ERRMAX,MAXIT,CHPARM,MAXNT,SYEAR
      COMMON/ERRORS/IERR,IWARN
      COMMON/FILES/IFGRD,IFSOIL,IFINIT,IFPOUT,IFMOUT,IFLUX,IPDOUT
      DIMENSION PSI(1),RMOIST(1),RK(1),Z(1),RKL(1),CL(1),RMSTL(1),
     *          STARK(1),POR(1),DZ(1)

C    Write assorted info. to general output file (GEN.OUT)
      YTIME=TIME/SYEAR
      WRITE(IPRT,1900) YTIME,NT,DT,ERR,ITER,NKITER,TDIST
      WRITE(IPRT,2010) (I,PSI(I),RMOIST(I),RK(I),I=1,NUMNP)

C    Write pressure data to pressure output file (PSI.PRN)
      WRITE(IFPOUT,'(/)')
      DO 4 I=1,NUMNP
CDP    CHANGE NEXT CARD FOR DOUBLE/SINGLE PRECISION
        IF (PSI(I).LT.0.D0) THEN
          PF=ALOG10(-PSI(I))
        ELSE
          PF=0.D0
        ENDIF
        WRITE(IFPOUT,2040) I,Z(I),PSI(I),PF
    4 CONTINUE

C    Write moisture data to moisture output file (MOIST.PRN)
      WRITE(IFMOUT,'(/)')
      WRITE(IFMOUT,2030) (L,Z(L),RMOIST(L),L=1,NUMNP)

C    Calculate flux and particle velocity and
C    write to flux output file (FLUX.OUT).
      DO 40 L=1,NUMEL
        LP1=L+1
        FLUX=-STARK(L)*(1.D0+(PSI(LP1)-PSI(L))/DZ(L))
        VEL=FLUX/((RMOIST(L)+RMOIST(LP1))/2.D0)
        ZZ=(Z(L)+Z(LP1))/2.D0
        WRITE(IFLUX,2040) L,ZZ,FLUX,VEL
   40 CONTINUE

      RETURN

 1900 FORMAT(//1X,71(1H*)//' TIME(YRS)=',F10.2,'  TIME STEP=',I5,'  DT='
     *,1PE12.3,'  ERR=',1PE12.3//' ITER=',I5,10X,'TOTAL K ITERATIONS=',I
     *10//' PARTICLE DEPTH IN LINER=',1PE12.3//1X,71(1H*)//
     *'  NODE      POTENTIAL         MOISTURE                 K'/)
 2010 FORMAT(I6,4X,1PE13.4,4X,0PF12.4,4X,1PE13.4)
 2030 FORMAT(I10,F10.3,F10.7)
 2040 FORMAT(I10,0PF10.2,2(1PE10.2,1X,1PE10.3))
      END
C*******************************************************************
```

```
C
      SUBROUTINE ERROR(A,N,ERRM)
C
C*********************************************************************
C
C     Compute scalar error ERRM from maximum absolute value of
C     first N components of vector A.
C
C*********************************************************************
      DIMENSION A(1)
      ERRM=0.D0
      DO 10 I=1,N
        AERR=ABS(A(I))
        IF(AERR.GT.ERRM) ERRM=AERR
   10 CONTINUE
      RETURN
      END
C*********************************************************************
C
      SUBROUTINE THOMAS(A,B,C,W,N)
C
C*******************************************************************
C
C      Solves tridiagonal matrix equation, A*B=C
C      See, for example, Pinder and Gray 'Finite Element Simulation
C      in Surface and Subsurface Hydrology', Academic Press 1977.
C
C*******************************************************************
      DIMENSION A(3,1),B(1),C(1),W(1)

C     Reduction
      W(1)=A(2,1)
      DO 10 I=2,N
      IM1=I-1
      W(I)=A(2,I)-A(1,I)*A(3,IM1)/W(IM1)
   10 CONTINUE

C     Substitution
      C(1)=B(1)
      DO 30 I=2,N
      IM1=I-1
      C(I)=B(I)-A(1,I)*C(IM1)/W(IM1)
   30 CONTINUE
      C(N)=C(N)/W(N)
      DO 40 I=2,N
      M=N-I+1
      MP1=M+1
      C(M)=(C(M)-A(3,M)*C(MP1))/W(M)
   40 CONTINUE
      RETURN
      END
C*********************************************************************
```

```
C
      SUBROUTINE MASBAL(PSI,RMSTL,OLDML,STARK,DZ,IOUT)
C
C*****************************************************************
C
C     Computes and prints mass balance of flux calculations.
C
C*****************************************************************
      COMMON/MASSB/STOR,TOTV1,TOTV2,FLUX10,FLUX20
      COMMON/TIMES/TIME,TIME1,NT,ERR,NOUT,TOUT(10),NOUT1,TOUT1
      COMMON/DEVICE/IRD,IPRT
      COMMON/INFO/NUMNP,NUMEL,NPM1,NPM2,NUMEL2,AM1,NX,DTV,
     *            DEPTH,ENDTIM,DT,DTMAX,ALPHA,PSIBOT,H,NZOUT,SRPARM,
     *            PSINIT,NCPTS,ERRMAX,MAXIT,CHPARM,MAXNT,SYEAR
      COMMON/ERRORS/IERR,IWARN
      DIMENSION PSI(1),RMSTL(1),STARK(1),OLDML(1),DZ(1)

C     Initialization
      IF(IOUT.EQ.-1) THEN
        TOTV1=0.D0
        TOTV2=0.D0
        STOR=0.D0
        FLUX10=STARK(1)*(1.D0+(PSI(2)-PSI(1))/DZ(1))
        FLUX20=STARK(NUMEL)*(1.D0+(PSI(NUMNP)-PSI(NUMEL))/DZ(NUMEL))
        RETURN
      ENDIF

C     Top flux
      FLUX1=STARK(1)*(1.D0+(PSI(2)-PSI(1))/DZ(1))
      VOL1=(FLUX1*ALPHA+FLUX10*AM1)*DT
      FLUX10=FLUX1
      TOTV1=TOTV1+VOL1

C     Bottom flux
      FLUX2=STARK(NUMEL)*(1.D0+(PSI(NUMNP)-PSI(NUMEL))/DZ(NUMEL))
      VOL2=(FLUX2*ALPHA+AM1*FLUX20)*DT
      FLUX20=FLUX2
      TOTV2=TOTV2+VOL2

C     Change in storage (assumes linear variation)
      DSTOR=0.D0
      DO 10 L=1,NUMEL
CDP     CHANGE NEXT CARD FOR DOUBLE/SINGLE PRECISION
        DELZ=ABS(DZ(L))
        L2 = L*2
        L1 = L2-1
        DSTOR=DSTOR+(RMSTL(L1)-OLDML(L1)+RMSTL(L2)-OLDML(L2))*DELZ/
     *          2.D0
   10 CONTINUE
      STOR=STOR+DSTOR
      IF(IOUT.NE.1) RETURN
      RATEST=DSTOR/DT
```

```
C    Compute flux and volume errors EFLUX and EVOL
CDP  CHANGE NEXT THREE CARDS FOR DOUBLE/SINGLE PRECISION
      EFLUX=ABS(-FLUX1+FLUX2+RATEST)
      EVOL=ABS(DSTOR-VOL1+VOL2)
      ETOT=ABS(STOR-TOTV1+TOTV2)
      EREL=ETOT/STOR

      WRITE(IPRT,2020) FLUX1,FLUX2,RATEST,EFLUX,VOL1,VOL2,
     *                 DSTOR,EVOL,TOTV1,TOTV2,STOR,ETOT,EREL
      RETURN

 2020 FORMAT(//' VOLUME BALANCE CALCULATIONS'//
     *' TOP FLUX   ',1PE12.4/
     *' BOTTOM FLUX',1PE12.4/
     *' STORAGE RATE',1PE12.4/
     *20X,12(1H-)/
     *' ERROR               ',1PE12.4///
     *' VOLUME IN           ',1PE12.4/
     *' VOLUME OUT          ',1PE12.4/
     *' STORAGE VOLUME      ',1PE12.4/
     *20X,12(1H-)/
     *' ERROR               ',1PE12.4///
     *' CUMULATIVE CHANGES'//
     *' VOLUME IN (-)       ',1PE12.4/
     *' VOLUME OUT          ',1PE12.4/
     *' STORAGE             ',1PE12.4/
     *20X,12(1H-)/
     *' ERROR               ',1PE12.4//
     *' RELATIVE ERROR      ',0PF12.6)
      END
C*****************************************************************
C
      SUBROUTINE NEWBC(PSI)
C
C*****************************************************************
C
C    Reads pressure boundary conditions PSI(1) and PSI(NUMNP)
C    from control file.
C
C*****************************************************************
      COMMON/INFO/NUMNP,NUMEL,NPM1,NPM2,NUMEL2,AM1,NX,DTV,
     *            DEPTH,ENDTIM,DT,DTMAX,ALPHA,PSIBOT,H,NZOUT,SRPARM,
     *            PSINIT,NCPTS,ERRMAX,MAXIT,CHPARM,MAXNT,SYEAR
      COMMON/DEVICE/IRD,IPRT
      DIMENSION PSI(1)

      READ(IRD,*) H,PSIBOT
      WRITE(IPRT,2020) H,PSIBOT
      PSI(1)=H
      PSI(NUMNP)=PSIBOT
      RETURN
```

```
 2020 FORMAT(//1X,19(1H=)/' BOUNDARY CONDITIONS'/1X,19(1H=)/
     */' HEAD IN IMPOUNDMENT',20(1H.),'H=',F10.2/
     *' UNDERLYING SOIL SUCTION PRESSURE',5(1H.),'PSIBOT=',F10.3)
      END
C*************************************************************
      SUBROUTINE CLEAR(A,N)
C*************************************************************
C     Fills first N components of vector A with zeros.
C*************************************************************
      DIMENSION A(1)
      DO 10 I=1,N
   10 A(I)=0.D0
      RETURN
      END
C******************************************************************
      SUBROUTINE COPY(A,N,B)
C****************************************************************
C     Copies first N components of vector A into B.
C******************************************************************
      DIMENSION A(1), B(1)
      DO 10 I=1,N
   10 B(I) = A(I)
      RETURN
      END
C******************************************************************
C
      SUBROUTINE PTRACK(STARK,DZ,RMOIST,PSI,Z,VTIME,VELO,ZDIST,
     *                  ITER,NKITER)
C
C******************************************************************
C
C     Particle tracking master subroutine -- numerically integrates
C     velocity function to determine distance travelled using
C     Simpson's rule (requires 3 points); and writes position to
C     screen and particle depth vs. time output file (PDT.OUT).
C
C******************************************************************
      COMMON/DEVICE/IRD,IPRT
      COMMON/FILES/IFGRD,IFSOIL,IFINIT,IFPOUT,IFMOUT,IFLUX,IPDOUT
      COMMON/INFO/NUMNP,NUMEL,NPM1,NPM2,NUMEL2,AM1,NX,DTV,
     *          DEPTH,ENDTIM,DT,DTMAX,ALPHA,PSIBOT,H,NZOUT,SRPARM,
     *          PSINIT,NCPTS,ERRMAX,MAXIT,CHPARM,MAXNT,SYEAR
      COMMON/TIMES/TIME,TIME1,NT,ERR,NOUT,TOUT(10),NOUT1,TOUT1
      COMMON/TRACK/NODEV1,NODEV2,NELEMV,KODEBT,NCLAYN,TIMEBT,OLDTD,
     *          TDIST
      DIMENSION STARK(1),DZ(1),RMOIST(1),PSI(1),Z(1),VTIME(3),
     *          VELO(3),ZDIST(3)

C     Write heading to standard output device (usually the screen)
C     every 20 lines of output
      IF(MOD(NT,20).EQ.0)  WRITE(*,900)
```

```
C     Evaluate second point (velocity as a function of time) to be
C     used in Simpson's method
       IF (MOD(NT,2).NE.0) THEN
         IARG=2
         VTIME(IARG)=TIME
         CALL VCALC(VELO,STARK,DZ,RMOIST,PSI,IARG)
         RETURN
       ENDIF

C     Evaluate third point
       IARG=3
       VTIME(IARG)=TIME
       CALL VCALC(VELO,STARK,DZ,RMOIST,PSI,IARG)

C     Numerically integrate velocity function and update position
       CALL SIMPUN(VTIME,VELO,ZDIST)
       OLDTD=TDIST
       TDIST=TDIST+ZDIST(3)
       CALL LOCATE(Z,VELO,VTIME)
       TYRS=TIME/SYEAR
       WRITE(IPDOUT,1002) NT,TYRS,TDIST
       WRITE(*,1001) NT,DT,TYRS,ZDIST(3),TDIST,NELEMV,ITER,NKITER
       IF(KODEBT.EQ.1) RETURN

C     Reassign third point of previous calculation to first point
C     of subsequent calculation
       NODEV1=NELEMV
       NODEV2=NODEV1+1
       VELO(1)=VELO(3)
       VTIME(1)=VTIME(3)
       RETURN

  900 FORMAT(/3X,'T-STEP',4X,'DT(S)',5X,'TIME(Y)',5X,'ZDIST',6X,'TDIST',
     *4X,'NELEMV',2X,'ITER',2X,'NKITER'/)
 1000 FORMAT(3(5X,1PE10.3))
 1001 FORMAT(4X,I5,4(1X,1PE10.3),2X,I5,2X,I5,3X,I5)
 1002 FORMAT(I10,2(5X,1PE10.3))
       END
C ***************************************************************
C
       SUBROUTINE VCALC(VELO,STARK,DZ,RMOIST,PSI,M)
C
C ***************************************************************
C
C      Calculates particle velocity, VELO(M), where M corresponds
C      to the point number in Simpson's integration scheme.
C
C ***************************************************************
       COMMON/TRACK/NODEV1,NODEV2,NELEMV,KODEBT,NCLAYN,TIMEBT,OLDTD,
     *             TDIST
       DIMENSION STARK(1),DZ(1),RMOIST(2),PSI(2),VELO(1)
```

```
      FLUX=-STARK(NELEMV)*(1.D0+(PSI(NODEV2)-PSI(NODEV1))/DZ(NELEMV))
      VELO(M)=FLUX/((RMOIST(NODEV1)+RMOIST(NODEV2))/2.0)
      RETURN
      END
C ******************************************************************
C
      SUBROUTINE SIMPUN(XX,FX,AX)
C
C ******************************************************************
C
C     Numerically evaluates the velocity integral (Simpson's rule).
C
C     SUBPROGRAM AUTHOR: J. BARISH, COMPUTING TECHNOLOGY CENTER,
C     UNION CARBIDE CORP., NUCLEAR DIVISION, OAK RIDGE TENN.
C
C ******************************************************************
      COMMON/INFO/NUMNP,NUMEL,NPM1,NPM2,NUMEL2,AM1,NX,DTV,
     *           DEPTH,ENDTIM,DT,DTMAX,ALPHA,PSIBOT,H,NZOUT,SRPARM,
     *           PSINIT,NCPTS,ERRMAX,MAXIT,CHPARM,MAXNT,SYEAR
      DIMENSION XX(2), FX(2), AX(2)

      AX(1)=0.0
      DO 10 IX=2,NX,2
         D1=XX(IX)-XX(IX-1)
         AX(IX)=AX(IX-1)+D1/2.0*(FX(IX)+FX(IX-1))
         IF(NX.EQ.IX) RETURN
         D2=XX(IX+1)-XX(IX-1)
         D3=D2/D1
         A2=D3/6.0*D2**2/(XX(IX+1)-XX(IX))
         A3=D2/2.0-A2/D3
   10 AX(IX+1)=AX(IX-1)+(D2-A2-A3)*FX(IX-1)+A2*FX(IX)+A3*FX(IX+1)
      RETURN
      END
C ******************************************************************
C
      SUBROUTINE LOCATE(Z,VELO,VTIME)
C
C ******************************************************************
C
C     Updates particle position in grid NELEMV, and
C     sets breakthrough flag KODEBT = 1 when particle passes
C     through liner base [ TDIST > Z(NCLAYN) ].
C
C ******************************************************************
      COMMON/DEVICE/IRD,IPRT
      COMMON/FILES/IFGRD,IFSOIL,IFINIT,IFPOUT,IFMOUT,IFLUX,IPDOUT
      COMMON/TIMES/TIME,TIME1,NT,ERR,NOUT,TOUT(10),NOUT1,TOUT1
      COMMON/TRACK/NODEV1,NODEV2,NELEMV,KODEBT,NCLAYN,TIMEBT,OLDTD,
     *              TDIST
      COMMON/INFO/NUMNP,NUMEL,NPM1,NPM2,NUMEL2,AM1,NX,DTV,
     *           DEPTH,ENDTIM,DT,DTMAX,ALPHA,PSIBOT,H,NZOUT,SRPARM,
```

```
     *             PSINIT,NCPTS,ERRMAX,MAXIT,CHPARM,MAXNT,SYEAR
      DIMENSION Z(1),VELO(3),VTIME(1)
      DATA AVEL/0.0/,CDIST/0.0/

      IF(TDIST.GT.0.0) TDIST=0.0

      I=0
   10 CONTINUE
        I=I+1
        IF (I.GT.NCLAYN) THEN
          KODEBT=1
          AVEL=(VELO(1)+VELO(2)+VELO(3))/3.D0
          CDIST=Z(NCLAYN)-OLDTD
          DTV=CDIST/AVEL
          YDTV=DTV/SYEAR
          YTIME=VTIME(1)/SYEAR
          TIMEBT=YTIME+YDTV
          WRITE(*,1000) NT,TIMEBT
          WRITE(IPRT,1000) NT,TIMEBT
        ELSE
          IF (TDIST.GT.Z(I)) THEN
            NELEMV=I-1
          ELSE
            IF (TDIST.EQ.Z(I)) THEN
              NELEMV=I
            ELSE
              GOTO 10
            ENDIF
          ENDIF
        ENDIF
      RETURN

 1000 FORMAT(//1X,71(1H*)//1X,'BREAKTHROUGH OCCURRED DURING TIME STEP ',
     *I5,5X,'YEAR = ',1PE10.2//1X,71(1H*)//)
      END
C ********************************************************************
C
      SUBROUTINE STEADY(PSI,SSPSI,KODESS)
C
C ********************************************************************
C
C     Determines time at which steady state is achieved by comparing
C     PSI at steady state SSPSI with PSI at current step.
C     When maximum change at any node is less than 1.0 cm,
C     steady state is achieved, KODESS is set to 1, and
C     steady state information is written to the screen.
C
C ********************************************************************
      COMMON/DEVICE/IRD,IPRT
      COMMON/FILES/IFGRD,IFSOIL,IFINIT,IFPOUT,IFMOUT,IFLUX,IPDOUT
      COMMON/TIMES/TIME,TIME1,NT,ERR,NOUT,TOUT(10),NOUT1,TOUT1
      COMMON/INFO/NUMNP,NUMEL,NPM1,NPM2,NUMEL2,AM1,NX,DTV,
```

```
     *           DEPTH,ENDTIM,DT,DTMAX,ALPHA,PSIBOT,H,NZOUT,SRPARM,
     *           PSINIT,NCPTS,ERRMAX,MAXIT,CHPARM,MAXNT,SYEAR
      DIMENSION PSI(1),SSPSI(1)

      CHMAX=0.0
      DO 10 I=2,NUMEL
         CHANGE=ABS(PSI(I)-SSPSI(I))
         IF(CHANGE.GT.CHMAX) CHMAX=CHANGE
   10 CONTINUE
      IF(CHMAX.LE.1.0) KODESS=1
      IF(KODESS.NE.1) RETURN
      YTIME=TIME/SYEAR
      WRITE(*,1930) NT,YTIME
      WRITE(IPRT,1930) NT,YTIME
      RETURN

 1930 FORMAT(//1X,71(1H*)//1X,'STEADY STATE ACHIEVED DURING TIME STEP',
     *I5,5X,'YEAR = ',1PE10.2//1X,71(1H*)//)
      END
```

```
C*********************************************************************
C
      SUBROUTINE SPROP(PSI,RMOIST,C,RK,
     *     RMSTL,CL,RKL,STARK,SATK,POR,ISOIL,PSICRT)
C
C*********************************************************************
C
C     Compute soil properties at nodes at each end of each element.
C     Since adjacent elements may have different soil types, a node
C     has two values, one for element above and one for below.
C
C*********************************************************************
      COMMON/MASSB/STOR,TOTV1,TOTV2,FLUX10,FLUX20
      COMMON/DEVICE/IRD,IPRT
      COMMON/INFO/NUMNP,NUMEL,NPM1,NPM2,NUMEL2,AM1,NX,DTV,
     *            DEPTH,ENDTIM,DT,DTMAX,ALPHA,PSIBOT,H,NZOUT,SRPARM,
     *            PSINIT,NCPTS,ERRMAX,MAXIT,CHPARM,MAXNT,ISTEDY,SYEAR
      COMMON/ERRORS/IERR,IWARN
      COMMON/FILES/IFGRD,IFSOIL,IFINIT,IFPOUT,IFMOUT,IFLUX,
     *             IFVOUT
      DIMENSION PSI(1),RMOIST(1),RKL(1),SATK(1),POR(1),C(1),
     *          RK(1),STARK(1),CL(1),RMSTL(1),ISOIL(1),PSICRT(1)

      IP = 0
      N = 1

C    Calculate soil properties
      DO 10 L = 1,NUMEL
        JSOIL = ISOIL(L)
        DO 20 NODE = 1, 2
          IF (NODE.EQ.2) N = N + 1
          IP = IP + 1
          IF (PSI(N).LE.PSICRT(L)) THEN

C          Relationship for PSI < PSI at inflection point

           PSINEG = -PSI(N)
           PSILN = LOG(PSINEG)
C          BRANCH ON Soil type
           GOTO (101,102,103,104,105,106,107,108,109,110,
     *          111,112) JSOIL
             IF (JSOIL.NE.0) STOP 'JSOIL out of bounds in SPROP'
C          Soil type 0
             RMSTL(IP) = 1.611D6*(POR(L)-0.075D0)/
     *                  (1.611D6+PSINEG**3.96D0)+0.075D0
             RKL(IP) = SATK(L)*1.175D6/(1.175D6+PSINEG**4)
             CL(IP) = 6.3796D6*(POR(L)-0.075D0)*PSINEG**2.96D0/
     *                 ((1.61D6+PSINEG**3.96D0)**2)
             GOTO 20
C          Soil type 1
  101        CONTINUE
             RMSTL(IP) = POR(L)/((PSINEG/12.1)**0.2469)
```

```
      RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**11.10
      CL(IP) = .4570*POR(L)/PSINEG**1.2470
      GOTO 20
C     Soil type 2
  102 CONTINUE
      RMSTL(IP) = POR(L)/((PSINEG/ 9.0)**0.2283)
      RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**11.76
      CL(IP) = .3770*POR(L)/PSINEG**1.2283
      GOTO 20
C     Soil type 3
  103 CONTINUE
      RMSTL(IP) = POR(L)/((PSINEG/21.8)**0.2041)
      RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**12.80
      CL(IP) = .1088*POR(L)/PSINEG**1.2041
      GOTO 20
C     Soil type 4
  104 CONTINUE
      RMSTL(IP) = POR(L)/((PSINEG/78.6)**0.1887)
      RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**13.60
      CL(IP) = .4299*POR(L)/PSINEG**1.1887
      GOTO 20
C     Soil type 5
  105 CONTINUE
      RMSTL(IP) = POR(L)/((PSINEG/47.8)**0.1855)
      RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**13.78
      CL(IP) = .3802*POR(L)/PSINEG**1.1855
      GOTO 20
C     Soil type 6
  106 CONTINUE
      RMSTL(IP) = POR(L)/((PSINEG/29.9)**0.1405)
      RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**17.24
      CL(IP) = .2264*POR(L)/PSINEG**1.1405
      GOTO 20
C     Soil type 7
  107 CONTINUE
      RMSTL(IP) = POR(L)/((PSINEG/35.6)**0.1290)
      RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**18.50
      CL(IP) = .2046*POR(L)/PSINEG**1.1290
      GOTO 20
C     Soil type 8
  108 CONTINUE
      RMSTL(IP) = POR(L)/((PSINEG/63.0)**0.1174)
      RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**20.04
      CL(IP) = .1909*POR(L)/PSINEG**1.1174
      GOTO 20
C     Soil type 9
  109 CONTINUE
      RMSTL(IP) = POR(L)/((PSINEG/15.3)**0.0962)
      RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**23.80
      CL(IP) = .1250*POR(L)/PSINEG**1.0962
      GOTO 20
C     Soil type 10
```

```
  110          CONTINUE
               RMSTL(IP) = POR(L)/((PSINEG/49.0)**0.0962)
               RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**23.80
               CL(IP) = .1398*POR(L)/PSINEG**1.0962
               GOTO 20
C            Soil type 11
  111          CONTINUE
               RMSTL(IP) = POR(L)/((PSINEG/40.5)**0.0877)
               RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**25.80
               CL(IP) = .1214*POR(L)/PSINEG**1.0877
               GOTO 20
C            Soil type 12
  112          CONTINUE
               RMSTL(IP) = 739.D0*(POR(L)-0.124D0)/
     *                      (739.D0+PSILN**4)+0.124D0
               RKL(IP) = SATK(L)*124.6D0/(124.6D0+PSINEG**1.77D0)
               CL(IP)  = 2956.D0*(POR(L)-.124D0)*PSILN**3/PSINEG/
     *                   (739.D0+PSILN**4)**2

          ELSE

C           Point of saturation

             IF (PSI(N).GE.0.0) THEN
               RMSTL(IP) = POR(L)
               CL(IP) = 0.0D0
               RKL(IP) = SATK(L)
               GOTO 20
             ENDIF

C           Relationship for PSI > PSI at inflection point

             PSINEG = -PSI(N)
C           Branch on soil type
             GOTO (201,202,203,204,205,206,207,208,209,210,
     *             211,212) JSOIL
               IF (JSOIL.NE.0) STOP 'JSOIL out of bounds in SPROP'
C            Soil type 0
               RMSTL(IP) = POR(L)
               RKL(IP) = SATK(L)
               CL(IP) = 0.0D0
               GOTO 20
C            Soil type 1
  201          CONTINUE
               RMSTL(IP) = POR(L)*.89850+(POR(L)*(9517112.6-
     *                     (9395661.6+(6867.*PSINEG)))**.5)/3433.5
               RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**11.10
               CL(IP) = POR(L)/(9517112.6-
     *                  (9395661.6+(6867.0*PSINEG)))**0.5
               GOTO 20
C            Soil type 2
  202          CONTINUE
```

```
            RMSTL(IP) = POR(L)*.89540+(POR(L)*(5046664.3-
     *                  (4977793.7+(5017.8 *PSINEG)))**.5)/2508.9
            RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**11.76
            CL(IP) = POR(L)/(5046664.3-
     *              (4977793.7+(5017.8*PSINEG)))**0.5
            GOTO 20
C         Soil type 3
  203       CONTINUE
            RMSTL(IP) = POR(L)*.89030+(POR(L)*(27432013.6-
     *                (27015529.7 +(11765.8*PSINEG)))**.5)/5882.9
            RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**12.80
            CL(IP) = POR(L)/(27432013.6-
     *              (27015529.7 +(11765.8*PSINEG)))**0.5
            GOTO 20
C         Soil type 4
  204       CONTINUE
            RMSTL(IP) = POR(L)*.88581+(POR(L)*(333012588.2-
     *                (327478085.9+(41202.4*PSINEG)))**.5)/20601.2
            RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**13.60
            CL(IP) = POR(L)/(333012588.2-
     *               (327478085.9+(41202.4*PSINEG)))**0.5
            GOTO 20
C         Soil type 5
  205       CONTINUE
            RMSTL(IP) = POR(L)*.88472+(POR(L)*(121120298.2-
     *               (119063670.1+(24879.1*PSINEG)))**.5)/12439.6
            RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**13.78
            CL(IP) = POR(L)/(121120298.2-
     *               (119063670.1+(24879.1*PSINEG)))**0.5
            GOTO 20
C         Soil type 6
  206       CONTINUE
            RMSTL(IP) = POR(L)*.85500+(POR(L)*(30352144.1-
     *                 (29479186.5 +(12887.2*PSINEG)))**.5)/6443.6
            RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**17.24
            CL(IP) = POR(L)/(30352144.1-
     *                 (29479186.5 +(12887.2*PSINEG)))**0.5
            GOTO 20
C         Soil type 7
  207       CONTINUE
            RMSTL(IP) = POR(L)*.83734+(POR(L)*(33601180.2-
     *                 (32333107.1 +(13845.5*PSINEG)))**.5)/6922.7
            RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**18.50
            CL(IP) = POR(L)/(33601180.2-
     *                (32333107.1 +(13845.5*PSINEG)))**0.5
            GOTO 20
C         Soil type 8
  208       CONTINUE
            RMSTL(IP) = POR(L)*.80564+(POR(L)*(69944841.1-
     *                (65873969.3 +(20761.9*PSINEG)))**.5)/10380.9
            RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**20.04
            CL(IP) = POR(L)/(69944841.1
```

```
     *                 -(65873969.3 +(20761.9*PSINEG)))**0.5
               GOTO 20
C            Soil type 9
  209          CONTINUE
               RMSTL(IP) = POR(L)*.54182+(POR(L)*(347853.3-
     *                     (99106.0    +(2177.1 *PSINEG)))**.5)/1088.5
               RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**23.80
               CL(IP) = POR(L)/(347853.3-
     *                  (99106.0+(2177.1 *PSINEG)))**0.5
               GOTO 20
C            Soil type 10
  210          CONTINUE
               RMSTL(IP) = POR(L)*.54182+(POR(L)*(3567840.6-
     *                    (1016504.7  +(6972.3 *PSINEG)))**.5)/3486.2
               RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**23.80
               CL(IP) = POR(L)/(3567840.6-
     *                  (1016504.7  +(6972.3 *PSINEG)))**0.5
               GOTO 20
C            Soil type 11
  211          CONTINUE
               RMSTL(IP) = POR(L)*(-3.64)+(POR(L)*
     *                (1074124.4+(671246.7-(569.5*PSINEG)))**.5)/284.7
               RKL(IP) = SATK(L)*(RMSTL(IP)/POR(L))**25.80
               CL(IP) = POR(L)/(1074124.4+
     *                 (671246.7 -(569.5 *PSINEG)))**0.5
               GOTO 20
C            Soil type 12
  212          CONTINUE
               RMSTL(IP) = POR(L)
               RKL(IP) = SATK(L)
               CL(IP) = 0.0D0
             ENDIF
   20      CONTINUE
   10 CONTINUE

C     Compute geometric mean interblock conductivity
      DO 120 L = 1,NUMEL
CDP     CHANGE NEXT CARD FOR DOUBLE/SINGLE PRECISION
          STARK(L) = SQRT(RKL(L*2-1)*RKL(L*2))
  120 CONTINUE

C     Compute mean RMOIST, RK, and C at nodes
      DO 30 I = 2,NPM1
          L1 = 2*(I-1)
          L2 = L1 + 1
          C(I) = (CL(L1)+CL(L2))/2.D0
          RMOIST(I) = (RMSTL(L1)+RMSTL(L2))/2.D0
CDP     CHANGE NEXT CARD FOR DOUBLE/SINGLE PRECISION
          RK(I) = SQRT(RKL(L1)*RKL(L2))
   30 CONTINUE

C     Use one node value at top and bottom
```

```
C(1) = CL(1)
RMOIST(1) = RMSTL(1)
RK(1) = RKL(1)
C(NUMNP) = CL(NUMEL2)
RMOIST(NUMNP) = RMSTL(NUMEL2)
RK(NUMNP) = RKL(NUMEL2)

RETURN
END
```

APPENDIX H

SOILINER PACKAGE

USING THE SOILINER PACKAGE

The SOILINER software package consists of three distinct modules:

(1) PRESOIL.BAS, the BASIC input preprocessor,
(2) SOILINER.EXE, the numerical model, and
(3) SGRAPH.WKS, the LOTUS 1-2-3 graphics macro.

The first two are linked together through the batch file SL.BAT. In addition, the package includes 8 default data sets for a variety of clay liner configurations, assorted utility files, and the FORTRAN source code for the model. The entire package is stored on one double-density, double-sided, 5 1/4 inch floppy disk. The package was developed for use on the standard EPA microcomputer configuration -- an IBM PC-AT equipped to display and print LOTUS graphics. Equivalent software is available for use on a Compaq or an IBM PC-XT, the only requirements being (a) a hard disk with LOTUS drivers properly installed, and (b) graphics capabilities. A desirable option is a math coprocessor (Intel 8087 for Compaq and XT, 80287 for AT), which significantly accelerates execution time. This manual describes installation and use of the SOILINER package, valid for all the above mentioned hardware configurations.

> ***This manual references a number of keystrokes. If the user is asked to type a command, only those letters within the quotes are to be typed. When the user is asked to press a key (or sequence of keys) the text within quotes usually represents a function key (i.e. "Return"). Simultaneous keystrokes are denoted by brackets (i.e. <Ctrl, Break>). Finally, all LOTUS command options are capitalized and denoted by quotes.

Section 1 - INSTALLATION

1.1 Insert SOILINER disk in drive A, and type "A:INSTALL" and press "Return".

1.2 The first installation action is to copy SOILINER.WKS, a small pointer macro, into the LOTUS 1-2-3 directory. If the default directory, C:\LOTUS, does not exist, an error condition will halt installation, and the user will be advised on how to proceed.

1.3 The directory C:\SOIL will be created and filled with the necessary files.

Section 2 - USING THE PREPROCESSOR

2.1 If the current drive is not C:, type "C:" and press "Return". If the current directory is not C:\SOIL, type "CD\SOIL" and press "Return".

2.2 Type "SL" and press "Return". This invokes the batch file SL.BAT. At this point, the user will be prompted for the first decision, whether to edit input data sets or not. If editing is selected, the user must first choose the flow scenario to be simulated. A second prompt will appear asking the user whether the original, default data sets or the most recently revised data sets are to be edited. Finally, the user is given a choice of editing the control file, the soil properties file, and/or the initial conditions file. Editing lasts until the user finally opts to exit the preprocessor, at which time the SOILINER program is invoked.

If the edit option is not chosen, the user will be prompted to select the default flow scenario to be simulated which, then invokes the SOILINER model. If one chooses to run user-created input data sets instead, a final series of prompts will appear requesting the appropriate data set names. Upon successfully entering all four data set names, the simulation will proceed.

> *** Note that the original default data sets have the extension .DEF and cannot be changed, whereas the most recently edited data sets have the extension .DAT.

Section 3 - EXECUTION OF THE MODEL

3.1 It is recommended that the user press <Ctrl, PrtSc> upon exiting the preprocessor, to print the screen activity of the simulation.

3.2 After exiting the preprocessor, the SOILINER program will execute. If the steady state algorithm is chosen (LSTEDY = T in the control file), the only screen display will be a brief summary of the results: (1) the number of iterations, and (2) the error at the last iteration. If the number of iterations exceeds the specified maximum number of iterations (ITER > MAXIT), a warning will be issued and the user should check for convergence.

If the transient solution strategy is chosen (LSTEDY = F in the control file), steady state information will be followed by time-stepping data (including the particle's position) and the times to steady state and breakthrough.

Section 4 - USING SGRAPH

Due to the flexibility of the SOILINER graphics post-processor, SGRAPH, this section of the manual is more detailed in scope.

Section 4.1 - Invoking SGRAPH

4.1.1 Get into the LOTUS directory by typing "CD\LOTUS" and pressing "Return".

4.1.2 Enter LOTUS 1-2-3 by typing "LOTUS" and pressing "Return".

4.1.3 Select 1-2-3 from the menu by pressing "Return". At the prompt, strike any key.

> *** If you hear a beep and get an error message at this point, press

"Return" and you should see a blank worksheet.

4.1.4 To retrieve the SOILINER worksheet, enter the 1-2-3 main menu by pressing "/", then select "FILE", then "RETRIEVE", then "SOILINER" from the consecutively displayed menus. The SOILINER worksheet (copied into the LOTUS directory during installation) will, in turn, retrieve the SGRAPH worksheet from C:\SOIL.

*** To make a selection from a menu, point to the desired selection using the arrow keys, and press "Return". One can also select from a menu on the very top line by typing the first letter of the desired option.

*** If SOILINER is not listed as an available worksheet, press "Esc", then "/" to enter the 1-2-3 main menu, then select "FILE" and then "DIRECTORY". Then type "C:\LOTUS" and press "Return", and perform step 4.1.4. again.

Section 4.2 - Creating Graphics

The following procedure will convert two sets of SOILINER data (pressure and moisture) into graphics using SGRAPH. Following this procedure the first time will provide familiarity with the SGRAPH COMMAND (CMD) MENU. Be sure that the 1-2-3 and PRINTGRAPH device drivers are properly installed. Refer to Section 4.3 when in trouble; Section 5 presents a breakdown of all SGRAPH menu options.

4.2.1 Select "IMPORT" from the SGRAPH main menu.

4.2.2 Select "PRESSURE" from the next menu. A list of data files will appear. Select the SOILINER pressure output file (PSI is the most recent) which you wish to graphically display.

4.2.3 Sit back and wait. This process takes a few moments, and the screen will jump and flash as the data is imported and prepared for graphical presentation.

4.2.4 Select "MOISTURE" from the displayed menu, and a list of data files will appear again. Select the SOILINER moisture output file (MOIST is the most recent) which you wish to graphically display, and wait again.

4.2.5 At this point the worksheet contains two sets of data, the maximum SGRAPH can handle at one time. Select "QUIT", then "GRAPH".

4.2.6 To graph the pressure data, select "PRESSURE".

4.2.7 Select "VIEW" to see the graph on the screen. Strike any key to return to the options menu.

4.2.8 Assuming that the graph is satisfactory, it may be saved for later printing. Refer to Section 5.0 to modify graphs using SGRAPH, or the LOTUS 1-2-3 manual to modify graphs using 1-2-3 commands directly. Select "SAVE", or skip to Section 4.2.11 if no graphics hadcopy is desired.

4.2.9 Enter a name for the graph being saved and press <return>.

4.2.10 Select "QUIT" to return to the main graph menu.

4.2.11 To graph the moisture data, select "MOISTURE" and repeat instructions 4.2.7 through 4.2.10.

4.2.12 At this point both graphs are prepared for printing. Quit to READY mode by selecting "QUIT" or pressing "Esc" three times (see section 4.3). Press "/" for the 1-2-3 main menu, then select "QUIT", then "YES".

4.2.13 Now in the LOTUS Access System menu, select "PRINTGRAPH" to create graphics hardcopies.

4.2.14 If PRINTGRAPH is not configured to read picture files from C:\SOIL, refer to the 1-2-3 manual for instructions on specifying C:\SOIL as the picture menu.

4.2.15 Select "SELECT", then the graphs to print. Making sure the printer is on line, select "GO".

Section 4.3 - Error Correction Keys

Users experienced with 1-2-3 will be accustomed to hitting the "Esc" key to "back out" of any unfamiliar territory they might run into in the 1-2-3 menu. This will work most of the time in SGRAPH, but not always. The following keystroke sequences will get you out of almost any unpleasant situations you might encounter in these steps - expect to become very familiar with all of these as you use SGRAPH.

4.3.1 <Ctrl, Break> "Esc", "Esc", "Esc",...<Alt, M>

First, press <Ctrl, Break> which stops SGRAPH. Then press "Esc" until the word READY appears in the mode indicator in the top right corner of the screen. Then press <Alt, M> to get back into the SGRAPH menu. This is the safest and most versatile escape mechanism, but has one quirk, in that SGRAPH will sometimes hang on to the <Ctrl, Break> command. If the computer beeps and you see "CTRL-BREAK" in the bottom left corner of the screen, try step 4.3.2.

*** If you should strike the "Break" (Scroll Lock) key without the "Ctrl" key, the word "SCROLL" will appear in the bottom right of the screen. If this happens, press "Scroll Lock" again to make this indicator disappear.

4.3.2 "Esc", "Esc",... "Esc"

Hit "Esc" until you are in familiar territory. If, in doing so, you find yourself in READY mode, press <ALT, M>. Or, if the mode indicator reads CMD READY, press "Return", then <Alt, M>. This sequence of key strokes should put you back into the SGRAPH menu.

4.3.3 QUIT (or Q)

QUIT is an option on all SGRAPH submenus. Select "QUIT" or type "Q" to back up to a more general menu, or to get back to the READY mode. The QUIT option is generally identical in function to the Esc key.

4.3.4 <Alt, M>

Should you select "QUIT" too many times, you lose the SGRAPH menu altogether and wind up in READY mode. If you see the word "READY" in the top right hand cell of the screen, press <Alt, M> to reenter the SGRAPH menu system.

Section 5 - THE SGRAPH MENU SYSTEM

The SGRAPH menu system is outlined in the table below. Each row of this table represents a single selection/command sequence. All commands are marked by asterisks and described in Sections 5.2 through 5.13. The remaining selections will route the user to a new menu.

5.1 The SGRAPH Menu

SELECTIONS AND COMMANDS

LEVEL I	LEVEL II	LEVEL III	LEVEL IV	LEVEL V
IMPORT	* PRESSURE			
IMPORT	* MOISTURE			
IMPORT	* CLEAR			
IMPORT	QUIT			
GRAPH	* PRESSURE			
GRAPH	* MOISTURE			
GRAPH	OPTIONS	XYSCALES	XSCALE	* AUTOMATIC
GRAPH	OPTIONS	XYSCALES	XSCALE	* MANUAL
GRAPH	OPTIONS	XYSCALES	XSCALE	QUIT
GRAPH	OPTIONS	XYSCALES	YSCALE	* AUTOMATIC
GRAPH	OPTIONS	XYSCALES	YSCALE	* MANUAL
GRAPH	OPTIONS	XYSCALES	YSCALE	QUIT
GRAPH	OPTIONS	XYSCALES	QUIT	
GRAPH	OPTIONS	* VIEW		
GRAPH	OPTIONS	* NAME		
GRAPH	OPTIONS	* SAVE		
GRAPH	OPTIONS	QUIT		
GRAPH	QUIT			
QUIT				

All QUIT selections lead back to the previous menu, except for the Level I QUIT selection, which leads to the 1-2-3 ready mode. To enter the Level I menu from READY mode, press <Alt, M>.

5.2 PRESSURE (Import Menu)

This command takes pressure data from a SOILINER output file and puts it into a predesignated range on the worksheet. Using this command will automatically delete any pressure data currently on the worksheet, but must be executed before the pressure data can be graphed.

5.3 MOISTURE (Import Menu)

Identical to 5.2, except that it is used for importing moisture data.

5.4 CLEAR (Import Menu)

This command erases all pressure and moisture data from the worksheet, leaving a clean sheet. Use this command for aesthetic purposes only - it is not necessary to execute this command before importing additional sets of pressure or moisture data.

5.5 PRESSURE (Graph Menu)

This command takes previously imported pressure data and puts it into a basic graphic format, ready for viewing. Note that while SGRAPH accomodates both pressure and moisture data simultaneously, it normally will store only one graph at a time. This function converts the pressure data into the current graph. Exercise all desired graphics options (XYSCALES, VIEW, NAME, SAVE) after using this function and before graphing a new data set.

5.6 MOISTURE (Graph Menu)

Identical to 5.5 but for moisture data.

5.7 XYSCALES XSCALE Automatic (Options Menu)

This step tells SGRAPH to automatically set the X (horizontal) scale on the current graph. Note that 1-2-3 will normally do this by default. Use this step only if you have tried adjusting the X scale manually and prefer the original format.

5.8 XYSCALES XSCALES Manual (Options Menu)

This step allows you to set the X scales manually. It will request a lower and an upper limit - press RETURN after entering each. Remember that you can always use step 5.7 if this function leads you astray.

5.9 XYSCALES YSCALE Automatic (Options Menu)

Same as 5.7, except that it sets the Y (vertical) scale.

5.10 XYSCALES YSCALE Manual (Options Menu)

Same as 5.8, except that it enables you to set the Y (vertical) scale.

5.11 VIEW (Options Menu)

Use this function to view the current graph on the screen. Press any key to get back to the OPTIONS menu.

5.12 NAME (Options Menu)

This is one of the links between SGRAPH and the 1-2-3 main menu. Use it if you are somewhat familiar with 1-2-3 graphics and want to store more graphs on the worksheet than just the current one. Enter a new name each time you use this function, since SGRAPH contains no provision to avoid overwriting graphs already named on the worksheet.

5.13 SAVE (Options Menu)

This is the major link between SGRAPH and Lotus's PRINTGRAPH. It saves the current graph in the directory C:\SOIL for future printing using PRINTGRAPH. See the 1-2-3 manual for a detailed explanation of PRINTGRAPH.

*** Remember that SAVE commands in the SGRAPH menu do not contain any provisions to avoid overwriting of old files. Type a new name for each new file you create.

DEFAULT DATA SETS

Included in the SOILINER package are default data sets for eight flow scenarios. Associated with each scenario are the four required data sets, as listed below:

1. Clay liner only (60 cm)

   CTRC60.DEF
   GRDC60.DEF
   PRPC60.DEF
   INTC60.DEF

2. Clay liner only (90 cm)

   CTRC90.DEF
   GRDC90.DEF
   PRPC90.DEF
   INTC90.DEF

3. Clay liner only (180 cm)

   CTRC180.DEF
   GRDC180.DEF
   PRPC180.DEF
   INTC180.DEF

4. Clay liner (60 cm), drainage layer (30 cm), and clay liner (90 cm)

   CTRCSC.DEF
   GRDCSC.DEF
   PRPCSC.DEF
   INTCSC.DEF

5. Clay liner (60 cm) with underlying native soil (300 cm)

   CTRCS60.DEF
   GRDCS60.DEF
   PRPCS60.DEF
   INTCS60.DEF

6. Clay liner (90 cm) with underlying native soil (300 cm)

   CTRCS90.DEF
   GRDCS90.DEF
   PRPCS90.DEF
   INTCS90.DEF

7. Clay liner (180 cm) with underlying native soil (300 cm)

   CTRCS180.DEF
   GRDCS180.DEF
   PRPCS180.DEF
   INTCS180.DEF

8. Clay liner (60 cm), drainage layer (30 cm), clay liner (90cm), with underlying native soil (300 cm)

   CTRCSCS.DEF
   GRDCSCS.DEF
   PRPCSCS.DEF
   INTCSCS.DEF

Note that for each set of input data files, CTR, GRD, PRP, and INT represent the control, grid, soil properties and initial conditions files respectively.