R80-33 OSP 87474

THE COUPLED TRANSPORT OF WATER AND HEAT IN A VERTICAL SOIL COLUMN UNDER ATMOSPHERIC EXCITATION

TC171
.M41
.H99
no.258

MIT LIBRARIES
3 9080 00151 4063

and

Peter S. Eagleson

RALPH M. PARSONS LABORATORY
FOR
WATER RESOURCES AND HYDRODYNAMICS

Department of Civil Engineering
Massachusetts Institute of Technology

Report No. 258

Prepared with the Support of
The National Science Foundation
Grant No. ATM-7812327

July 1980

MIT

Water Engineering Library
AUG 29 1980
LIBRARIES

DEPARTMENT
OF
CIVIL
ENGINEERING

SCHOOL OF ENGINEERING
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
Cambridge, Massachusetts 02139

R80-33

OSP 87474

THE COUPLED TRANSPORT OF WATER AND HEAT IN A VERTICAL SOIL COLUMN UNDER ATMOSPHERIC EXCITATION

by

P. Christopher D. Milly

and

Peter S. Eagleson

RALPH M. PARSONS LABORATORY

FOR

WATER RESOURCES AND HYDRODYNAMICS

Department of Civil Engineering

Massachusetts Institute of Technology

Report No. 258

Prepared with the support of

The National Science Foundation

Grant No. ATM-7812327

July 1980

M.I.T. LIB
AUG 29 1980
RECEIVED

THE COUPLED TRANSPORT OF WATER AND HEAT IN A VERTICAL SOIL COLUMN UNDER ATMOSPHERIC EXCITATION

ABSTRACT

The purpose of this work is to develop a detailed, physically-based model of the response of the land surface to atmospheric forcing.

The coupled, nonlinear partial differential equations governing mass and heat transport in the soil are derived. The theory of Philip and de Vries is re-cast in terms of the soil water matric potential, accounting for soil inhomogeneities and hysteresis of the moisture retention process. An existing model of hysteresis is modified to incorporate the effect of temperature and to facilitate numerical analysis.

The Galerkin finite element method is applied in the development of a numerical algorithm for the solution of the governing equations. The numerical procedure is coded in FORTRAN for computer solution and several examples are run in order to test the method. The various modes of mass and heat transport are simulated accurately. A proposed procedure for the evaluation of non-linear storage coefficients in the numerical scheme yields excellent mass and energy balances.

0739511

ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation under Grant No. ATM-7812327.

This document is essentially the Master's thesis submitted by Mr. P. Christopher D. Milly to the Department of Civil Engineering at M.I.T. Its completion was supervised by Dr. Peter S. Eagleson, Professor of Civil Engineering.

Ms. Anne Clee typed the manuscript and Mr. Pedro Restrepo provided technical assistance.

TABLE OF CONTENTS

LIST OF FIGURES

NOTATION

$\underline{\underline{A}}_1$	global mass storage matrix associated with ψ
$\underline{\underline{A}}_2$	global energy storage matrix associated with ψ
a	negative of C_ψ
$\underline{\underline{B}}_1$	global mass storage matrix associated with T
$\underline{\underline{B}}_2$	global energy storage matrix associated with T
C	overall volumetric heat capacity of porous medium
C_D	temperature coefficient of isothermal diffusivity
C_d	bulk volumetric heat capacity of porous medium
C_K	temperature coefficient of hydraulic conductivity
C_ψ	temperature coefficient of matric potential
$\underline{\underline{C}}_1$	global mass conductance matrix associated with ψ
$\underline{\underline{C}}_2$	global energy conductance matrix associated with ψ
c_ℓ	specific heat of liquid water
c_p	specific heat of water vapor at constant pressure
c_1-c_6	mass equation coefficients defined in (4.3)
D_a	molecular diffusion coefficient of water vapor in air ($= 5.8 \times 10^{-7} T^{2.3}$)
$\underline{\underline{D}}_1$	global mass conductance matrix associated with T
$\underline{\underline{D}}_2$	global energy conductance matrix associated with T

D_{Tv}^{ψ} temperature diffusivity of vapor in ψ-T system

$D_{\psi v}$ matric head diffusivity of vapor in ψ-T system

d_1-d_5 energy equation coefficients defined in (4.4)

$\underline{E}_1$ global vector for gravity flow term in mass equation

$\underline{E}_2$ global vector for gravity flow term in heat equation

e rate of evaporation from the land surface

$\underline{F}_1$ root sink vector in mass equation

f vapor diffusion correction factor for liquid islands

$\underline{G}$ portion of grad ψ due to variations of $\underline{R}$ and $\underline{s}$

g acceleration of gravity

H sensible heat transport up from land surface

h relative humidity

I_ℓ net longwave back radiation from surface

I_s net incident shortwave radiation

$K(\theta,T)$ hydraulic conductivity

K_a atmospheric vapor transport coefficient

$K_r(\theta)$ relative hydraulic conductivity function

K_s value of hydraulic conductivity at $\theta=\theta_u$, $T=T_o$

$K_T(T)$ temperature correction for hydraulic conductivity

$\underline{k}$ unit vector in positive z direction

$L(T)$	latent heat of vaporization
L_o	L evaluated at $T = T_o$
n	total porosity
PF	value of pF referenced to $T = T_o$
pF	base-10 logarithm of negative matric head in cm.
Q_m	volumetric vertical water flux, excluding root flow
$\underline{Q}_1$	boundary mass flux vector
$\underline{Q}_2$	boundary heat flux vector
$\underline{q}$	general flux term
$\underline{q}_h$	heat flux
$\underline{q}_\ell$	mass flux of liquid water
$\underline{q}_m$	total mass flux of water in medium
$\underline{q}_r$	mass flux of water in root system "continuum"
$\underline{q}_v$	mass flux of water vapor
R	gas constant of water vapor
$\underline{R}$	wetting history vector
R_s	surface runoff rate
r	precipitation rate
S	volumetric root moisture sink term
S_h	volumetric heat storage
S_m	total volumetric water storage

S_{mi} S_m at local node number i

S_m^k S_m at time t^k

$\underline{s}$ vector of all soil parameters needed to define main wetting curve

T absolute (Kelvin) temperature

$\hat{T}$ Galerkin approximation of T field

T_j value of T at node j

T_r temperature of precipitation

T_o reference value of T

$\underline{T}^k$ vector of T_j's at time t^k

t^k time at end of k'th time step

U union symbol

W differential heat of wetting

z vertical coordinate, datum at land surface

α air phase tortuosity factor

γ specific gravity of liquid water

Δ^e length of element e

ε surface retention water depth

ε_o surface retention capacity

ζ temperature gradient enhancement factor for vapor diffusion

θ volumetric liquid water content

θ_a volumetric air content

$\theta_d(\cdot)$ main drying curve

θ_k maximum value of θ at which liquid phase is effectively discontinuous

θ_u "saturation" θ after drying and re-wetting

$\theta_w(\cdot)$ main wetting curve

θ^k value of θ at time t^k

λ effective thermal conductivity ($=\lambda^{\psi} \cong \lambda^{\theta}$)

λ^{θ} effective thermal conductivity in θ-T system

λ^{ψ} effective thermal conductivity in ψ-T system

λ_* hypothetical thermal conductivity in absence of water mass transport

μ dynamic viscosity of liquid water

ρ_a water vapor density at z = a above surface

ρ_ℓ density of liquid water

ρ_v density of water vapor

ρ_o saturation water vapor density over free water

Σ^k volume of water stored in element at time t^k

σ interfacial surface tension between air and water

Υ bulk volumetric storage of water in roots

Φ total potential of soil water

ϕ_i, ϕ_i^e trial function associated with element e and local node i

Ψ matric potential referenced to $T = T_o$, i.e., $\Psi = \psi e^{a(T-T_o)}$

Ψ^k value of Ψ at time t^k

Ψ_o effective reversal value of Ψ for computation of current scanning curve

ψ matric potential

$\hat{\psi}$ Galerkin approximation of ψ field

ψ_j value of ψ at node j

$\underline{\psi}^k$ vector of ψ_j's at step k

ω_j^e trial function associated with element e, node j

Chapter 1

INTRODUCTION

1.1 Motivation for this Study

The climate and weather at all scales on Earth is the result of the dynamic interaction of the atmosphere with the oceans, with the land surface, and with "outer space." Interest in the special role of the land surface as an atmospheric boundary is the motivation for the work described in this document, which is part of a larger study, "A Dynamic Land Surface Boundary Condition for Climate Models," being conducted at M.I.T. with the support of the National Science Foundation. The purpose of that study is to develop simple, physically-based parameterizations of the transport of (water) mass, heat, and momentum across the land surface for use in global numerical models of atmospheric general circulation.

The behavior of those simplified land surface models will be validated by comparison with a detailed mathematical model of the coupled physics of the soil, the vegetation, and the atmospheric boundary layer, driven by a stochastic model of the atmosphere. This physical system, which is considered to be one-dimensional (in the vertical direction), is depicted in Figure 1.1. This report describes the detailed "primitive" equation model of the soil component of this system. The soil column may be loosely defined as a vertical section of earth extending downward from the land surface either to the water table or to a depth beyond which the water mass and heat fluxes are negligible.

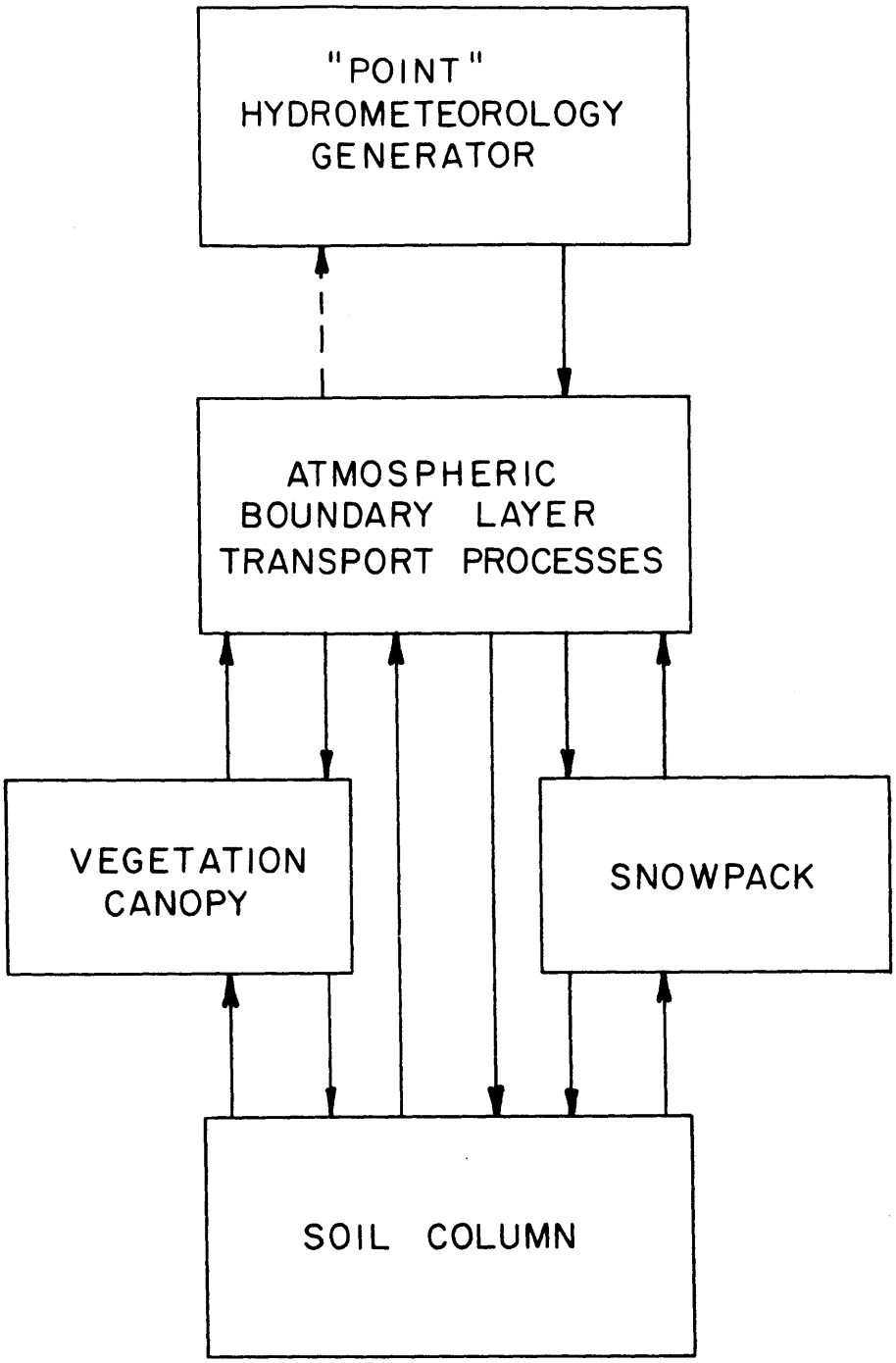


Figure 1.1

THE COMPONENTS OF THE LAND SURFACE BOUNDARY OF THE ATMOSPHERE

1.2 Goals and Scope

The first goal of the present work is to present a general mathematical statement of the physics of water and heat transport in the soil boundary layer. Chapters 2 and 3 deal with this problem. The second goal is to outline a numerical algorithm for the solution of the governing equations and to demonstrate its validity. These issues are treated in Chapters 4 and 5.

In formulating a general mathematical statement, the approach will be to consider all physical effects that might be important in natural environments. Hysteresis in the moisture retention curve, vapor transport, and temperature effects are therefore included. Subsequent work will address the question of simplifications of these general equations. It is precisely the purpose of this model to justify the simplification process, as the complex governing equations do not submit readily to an *a priori* scale analysis.

The role of vegetation is not treated in detail here. In order that the soil model be compatible with a mathematical model of the vegetal canopy, however, a general moisture sink (root extraction) term has been included in the derivation of the equations.

This work should have applications beyond studies of the planetary boundary layer. The theoretical presentation provides a convenient synthesis of much of the previous work. The numerical model will be a useful tool for the simulation of many soil physical processes, especially those involving coupled moisture and heat fields and/or vapor effects - situations for which general numerical codes are not readily

available.

1.3 Review of Relevant Literature

The development, during the first half of this century, of the isothermal theory of moisture flow in unsaturated soil is described by Swartzendruber (1969) and by Philip (1957). They credit Buckingham, followed by L. A. Richards, W. Gardner, E. C. Childs, N. Collis-George and others for their early contributions. The 1950's saw systematic application of the established theory to particular problems, the most well-known, perhaps, being the work of J. R. Philip on the infiltration process. Today, the Richards equation of liquid continuity, or a variant thereof, is probably the most widely used tool for the analysis of moisture transport in soil.

Philip and de Vries (1957) extended the isothermal theory of liquid water flow by incorporating vapor flow and clarifying the effect of temperature gradients in a synthesis of several divergent areas of research. They also presented a heat conduction equation that included the transport of latent heat by diffusing water vapor. Shortly thereafter, de Vries (1958) presented a more general heat equation, which accounted for many secondary storage and transport mechanisms and incorporated a new method for the calculation of effective thermal conductivity.

The theory of moisture transport in porous media has been complicated by the well-known existence of a hysteretic relationship between the moisture potential and the moisture content. A physical

theory explaining this phenomenon qualitatively in terms of capillarity was set forth by Miller and Miller (1956). The "independent domain" theory for hysteretic phenomena, developed by L. Néel and D. H. Everett, was first applied to soil moisture retention by Poulovassilis (1962). Subsequent developments along this line of attack are described by Mualem (1973). An important step was the suggestion of Philip (1964) that a similarity assumption be invoked. During recent years, Y. Mualem has made several contributions in this area, simplifying the formulation and maximizing the use of information to the point where the independent domain model and related models have become potentially powerful tools for the analysis of soil moisture flow problems.

Although models of hysteresis have been incorporated in several numerical simulations of soil water dynamics, a complete theoretical treatment of the applicability of the various flow equations in the presence of hysteresis has apparently not yet been given. Childs (1964) has made an important contribution, demonstrating that the commonly-employed soil moisture diffusivity must be modified to account for hysteresis. Very little has been written on the theory of hysteresis in the presence of temperature variations.

Several analytic solutions of the non-linear moisture equations have been derived. Philip (1957) has presented a quasi-analytic method for analyzing infiltration into a soil of constant moisture content. Gardner (1958), using a particular soil parameterization, solved the problem of steady flow of water from a fixed water table to the surface, where it evaporates. Covey (1963) has solved a transient evaporation

problem using an exponential diffusivity function. Eagleson (1978), using Philip's method, has defined effective diffusivities in terms of an empirical model of the soil properties in order to solve problems of infiltration and exfiltration for constant initial conditions.

Analytic solutions typically involve idealized assumptions concerning initial conditions, homogeneity, hysteresis, etc. Nevertheless, they are very useful in generic studies of moisture transport processes and in situations where detailed computation would be prohibitive.

The limitations inherent in the analytic solutions, together with the rapid growth of the capabilities of the digital computer, have led many investigators to seek numerical solutions of the flow equations. An early numerical solution was Klute's (1952) treatment of non-linear unidirectional moisture diffusion, neglecting gravity. A more general approach to infiltration, which allows for layered soils, was presented by Hanks and Bowers (1962). Hysteresis was incorporated into an analysis of redistribution after infiltration by Rubin (1967), and later by many others. Freeze (1969) presents a review of various solutions obtained during the 1960's.

Two- and three-dimensional numerical models have been developed to study the areal response of the coupled saturated zone-unsaturated zone system by Rubin (1968), Freeze (1971), Neuman, et al. (1975), Narasimhan and Witherspoon (1977), and others.

Sasamori (1970) used the theory of Philip and de Vries (1957) to model the coupled transport of heat and moisture in the soil for his landmark study of the interaction of the earth and atmosphere surface boundary layers. The numerical model was used to simulate a (precipitation-free) period of evaporation from the soil, and did not include the hysteresis effect. Furthermore, it employed the somewhat limited θ-based formulation of the conservation equations. Vauclin, et al. (1977) have used the (isothermal) Richards equation for liquid flow, and a simple heat diffusion equation, in order to study evaporation. Thermal properties were dependent upon moisture content, providing one coupling mechanism.

Sophocleous (1979) has attempted to convert the Philip and de Vries theory to a matric head-based formulation and has presented the results of some simulation experiments. It appears, however, that his new system of equations is incorrectly formulated (Milly, 1979). In particular, the expression for moisture flow due to temperature gradients is in error. Hysteresis was not considered.

Chapter 2

THE PHYSICS OF WATER AND HEAT TRANSPORT IN POROUS MEDIA

2.1 Introduction

A physically-based analysis of water and heat transport processes in the soil must begin with derivation of the governing equations and accompanying boundary and initial conditions from established principles. This is the objective of this chapter. The development is presented for heterogeneous, non-deforming, isotropic media exhibiting hysteresis. The coupled partial differential equations governing the distributions of temperature and water mass in the porous medium are derived by application of the conservation principle to water and heat at macroscopic scale. This gives

$$\frac{\partial \Sigma}{\partial t} = - \nabla \cdot \underline{q} \qquad (2.1)$$

where Σ is the amount of "substance" (e.g., water or enthalpy) per unit volume $[XL^{-3}]$, and $\underline{q}$ is the local average flux of substance $[XL^{-2}T^{-1}]$. X represents the units of the substance.

Expressions are given for the storage and flux terms as functions of the relevant dependent variables. Substitution into the general conservation equation (2.1) then yields the mass and heat conservation equations. Various mathematical manipulations are then investigated in an attempt to express the equations in diffusion forms, which are potentially more useful. Two pairs of dependent variables are

considered - (ψ, T) and (θ, T).

This chapter draws heavily on the works of Philip and de Vries (1957) and of de Vries (1958), although several extensions and modifications of these works are proposed. In particular, a system based on the dependent variables ψ and T is presented, along with one employing θ and T as proposed by Philip and de Vries (1957). The proposed systems of equations are valid for inhomogeneous and hysteretic soils when applied with care. The possibility of water uptake by a plant root system is included. Combination of the basic heat equation and the mass equation yields two other forms of the heat equation, one of which is equivalent to that given by de Vries (1958). A physically-small mathematical error in his equation is noted.

Following the derivation of the governing equations, the proper boundary and initial conditions for a one-dimensional (vertical) soil system are presented. All of the processes possible at the land surface (infiltration with or without ponding, evaporation and the accompanying heat transfers) are considered. The saturated zone is modeled dynamically as a linear storage reservoir. The conditions necessary to allow modeling of domain discontinuities are given.

The chapter closes with a discussion of the differences between ψ-based and θ-based forms of the conservation equations.

2.2 Derivation of the Governing Equations

2.2.1 Mass Conservation

Let S_m be defined as the total (liquid plus vapor) water

storage per unit volume of the porous medium. The units of S_m are equivalent volume of liquid water per unit bulk volume of the medium. We then have

$$\rho_\ell S_m = [\rho_\ell \theta + \rho_v \theta_a + \rho_\ell T] = \frac{\text{total mass of water}}{\text{unit bulk volume}} \qquad [\text{g cm}^{-3}]$$

or

$$S_m = \theta + \frac{\rho_v}{\rho_\ell} \theta_a + T \tag{2.2}$$

where

$$\rho_v = \text{vapor density} = \frac{\text{mass of water vapor}}{\text{unit volume of air}} \qquad [\text{g cm}^{-3}]$$

$$\rho_\ell = \text{liquid density} = \frac{\text{mass of liquid water}}{\text{unit volume of liquid water}} \qquad [\text{g cm}^{-3}]$$

$$\theta = \frac{\text{volume of liquid water in the soil}}{\text{unit bulk volume}}$$

$$\theta_a = \frac{\text{volume of air in the soil}}{\text{unit bulk volume}}$$

$$T = \frac{\text{volume of liquid water stored in roots}}{\text{unit bulk volume}}$$

The first term in (2.2) is thus storage in the liquid phase, the second is water vapor storage in the air-filled portion of the medium, and the third is root storage. In this work, θ refers to <u>total</u> soil liquid content, including the portion that is sometimes referred to as "residual" moisture content. The liquid water is assumed incompressible.

The flux of <u>liquid</u> in an unsaturated porous medium is described by the Buckingham-Darcy equation (Swartzendruber, 1969). This relation, for isotropic media, is

$$\underline{q}_\ell / \rho_\ell = - K \nabla \Phi \tag{2.3}$$

in which

$\underline{q}_\ell$ = $\frac{\text{liquid mass flow rate in soil}}{\text{unit bulk cross-sectional area}}$ [g cm^{-2} s^{-1}]

K = hydraulic conductivity [cm s^{-1}]

Φ = total potential [cm]

The total potential Φ is a measure of the local energy level of the water. It is conveniently resolved into a gravity term and a matric, or "pressure," term,

$$\Phi = \psi + z \tag{2.4}$$

where the z-coordinate is the elevation relative to an arbitrary fixed datum (positive upward). When the presence of solutes is important, there is another component, the osmotic potential, on the right-hand side of Eq. (2.4).

The quantity, ψ, the matric head, is fully determined by the liquid moisture content, the temperature, the soil properties, and (in the general case of hysteretic soils) the wetting history of the particular point in the medium. This dependence is discussed in Chapter 3. Keeping in mind the possibility of hysteresis, as well as the possible spatial variability (heterogeneity) of the soil, we shall write

$$\psi = \psi(\theta, T; \underline{R}; \underline{s}) \tag{2.5}$$

where T is temperature and $\underline{R}$ and $\underline{s}$ are vectors providing all the requisite information about wetting history and soil properties, respectively. Their physical significance is dealt with in Chapter 3.

Substitution of Equation (2.4) into (2.3) yields

$$q_\ell/\rho_\ell = - K\nabla\psi - K\underline{k} \qquad (2.6)$$

where $\underline{k}$ is the unit vector in the z direction. Since K is ordinarily discontinuous where $\underline{s}$ is discontinuous, the use of Equations (2.3) and (2.6) will be restricted to media in which the properties are either constant or continuously varying in space.

The "simple theory" of <u>vapor</u> flow in porous media is an intuitive extension of Fick's law to the macroscopic scale of a porous material. It is written (Philip and de Vries, 1957)

$$\underline{q}_v = - D_a \nu \alpha \theta_a \nabla\rho_v \qquad (2.7)$$

where

$\underline{q}_v$ = $\frac{\text{vapor mass flow rate in soil}}{\text{unit bulk cross-sectional area}}$ [g cm^{-2} s^{-1}]

D_a = molecular diffusion coefficient of water vapor in air [cm^2 s^{-1}]

ν = mass flow factor, a correction for the difference in boundary conditions governing air and vapor

α = tortuosity of the air-filled pore space

The mass flow factor, which is normally very close to unity, will be neglected here. The tortuosity, which is less than unity, accounts for the effect of the increased path length that results from the irregular geometry of the pore space. The proportion of the bulk cross-section available for flow is assumed to be identical to θ_a.

The water vapor density in the air phase is found by Edlefson and Anderson (1943, p. 145) to be

$$\rho_v = \rho_o h = \rho_o(T) \exp(\psi g/RT) \qquad (2.8)$$

in which

$\rho_o(T)$ is the saturation vapor density over a free, flat water surface [g cm^{-3}]

h is the relative humidity of air

g is the acceleration of gravity [cm s^{-2}]

R is the water vapor gas constant [erg g^{-1} K^{-1}]

Equation (2.8) is a direct consequence of the assumption of thermodynamic equilibrium between the liquid and vapor phases, and is valid for practically all natural soil systems.

The continuity of $\rho_v(\psi, T)$ allows us to expand the gradient in Equation (2.7) to obtain (setting $\nu = 1$)

$$\underline{q}_v = - D_a \alpha\theta_a \left(\left.\frac{\partial \rho_v}{\partial \psi}\right|_T \nabla\psi + \left.\frac{\partial \rho_v}{\partial T}\right|_\psi \nabla T \right) \qquad (2.9)$$

in which the vertical bar indicates which variable is to be held constant in the differentiation. Philip and de Vries (1957) use θ and T in this expansion and assume that h is a function only of θ, obtaining

$$\nabla\rho_v = h \frac{d\rho_o}{dT} \nabla T + \rho_o \frac{dh}{d\theta} \nabla\theta$$

In fact, the temperature dependence of h may be significant. We shall retain the more general Equation (2.9), but follow the same subsequent

steps proposed by Philip and de Vries (1957).

Consider now the case of a discontinuous liquid phase, i.e., $\theta \leq \theta_k$, where θ_k is the highest value of θ for which $K = 0$. Philip and de Vries (1957) argue that the isolated liquid islands act as short-circuits for the thermally-induced vapor flux, effectively increasing the cross-section available for transport to include the liquid phase. (They rejected a similar correction for the flux due to matric head gradients, apparently because the necessary head gradients across the liquid islands would be too large.) The increased cross-section is incorporated by replacing θ_a in the ∇T term of Equation (2.9) by $(\theta_a + \theta)$. Note that this change is rather small, since $\theta < \theta_k$.

A second correction to Equation (2.9) is made by Philip and de Vries (1957) to account for the fact that the average magnitude of the temperature gradient in the air-filled pores exceeds the magnitude of the macroscopic gradient, due to the difference in thermal conductivity among the various phases. They introduce

$$\zeta = \frac{(\nabla T)_a}{\nabla T} \tag{2.10}$$

as the necessary correction, $(\nabla T)_a$ being the average temperature gradient in the air phase, the effect of tortuosity included.

The modified version of Equation (2.9) is

$$\underline{q}_v = - D_a \alpha\theta_a \left.\frac{\partial \rho_v}{\partial \psi}\right|_T \nabla\psi - D_a(\theta_a + \theta) \left.\frac{\partial \rho_v}{\partial T}\right|_\psi \zeta \nabla T \tag{2.11}$$

which is valid for $\theta < \theta_k$.

Finally, Philip and de Vries (1957) propose a generalization of Equation (2.11) for the range of θ where the liquid phase is continuous. In this situation, the quantity $(\theta_a + \theta)$ in Equation (2.11) is replaced by $(\theta_a + f'\theta)$, where f' goes to zero as θ_a goes to zero. The proposed form of f' is linear with θ_a in this regime. The final expression for vapor flux is

$$\underline{q}_v = - D_a \alpha\theta_a \left.\frac{\partial \rho_v}{\partial \psi}\right|_T \nabla\psi - D_a f \zeta \left.\frac{\partial \rho_v}{\partial T}\right|_\psi \nabla T \qquad (2.12)$$

$$= - \rho_\ell D_{\psi v} \nabla\psi - \rho_\ell D^{\psi}_{Tv} \nabla T$$

in which

$$f = \begin{cases} n & \theta \leq \theta_k \\ \theta_a + \dfrac{\theta_a}{n - \theta_k}\theta & \theta_k < \theta \end{cases} \qquad (2.13)$$

$$D_{\psi v} = \frac{D_a}{\rho_\ell} \alpha\theta_a \left.\frac{\partial \rho_v}{\partial \psi}\right|_T = \text{matric head diffusivity of vapor} \quad [\text{cm s}^{-1}]$$

$$D^{\psi}_{Tv} = \frac{D_a}{\rho_\ell} f \zeta \left.\frac{\partial \rho_v}{\partial T}\right|_\psi = \text{temperature diffusivity of vapor in } \psi - T \text{ system} \quad [\text{cm}^2\ \text{s}^{-1}\ {}^\circ\text{K}^{-1}]$$

The validity of this model of vapor transport has not been exhaustively demonstrated, although Philip and de Vries (1957) cite several experiments that support it. The interaction of the ψ- and T-fields on the microscopic (pore-scale) level is deserving of further theoretical work, in conjunction with careful analyses of the existing experimental data. Most likely, further experiments will be required.

Rather than open this Pandora's box of micro-scale physics, we shall accept the present theory as the proper formulation of the process dynamics.

At this point, we can define the total mass flux in the medium, using (2.6) and (2.12), to be

$$\underline{q}_m/\rho_\ell = \underline{q}_\ell/\rho_\ell + \underline{q}_v/\rho_\ell + \underline{q}_r/\rho_\ell$$

$$= -(K + D_{\psi v})\,\nabla\psi - D^{\psi}_{Tv}\,\nabla T - K\underline{k} + \underline{q}_r/\rho_\ell \qquad (2.14)$$

where

$$q_r = \frac{\text{water mass flow rate in plant root system}}{\text{unit bulk cross-sectional area}} \qquad [\text{g cm}^{-s}\ \text{s}^{-1}]$$

Substitution of (2.2) and (2.14) into (2.1) yields the continuity equation of water mass,

$$\frac{\partial}{\partial t}\left(\theta + \frac{\rho_v}{\rho_\ell}\,\theta_a + T\right) = \nabla\cdot[(K + D_{\psi v})\,\nabla\psi + D^{\psi}_{Tv}\,\nabla T + K\underline{k} - \underline{q}_r/\rho_\ell] \qquad (2.15)$$

Considering mass conservation in the root system, we also have

$$\frac{\partial T}{\partial t} = -\nabla\cdot(\underline{q}_r/\rho_\ell) + S \qquad (2.16)$$

where S is the volumetric uptake of water from the soil by the plant roots, i.e.,

$$S = \frac{\text{volume of liquid extracted from soil per unit time}}{\text{unit bulk volume}}$$

Combination of (2.15) and (2.16) yields

$$\left(1 - \frac{\rho_v}{\rho_\ell}\right) \frac{\partial \theta}{\partial t} + \frac{\theta_a}{\rho_\ell} \frac{\partial \rho_v}{\partial t} = \nabla \cdot [(K + D_{\psi v}) \nabla \psi + D_{Tv}^{\psi} \nabla T + K\underline{k}] - S \quad (2.17)$$

where use has been made of the fact that

$$\frac{\partial \theta_a}{\partial t} = - \frac{\partial \theta}{\partial t}$$

2.2.2 Heat Conservation

Let S_h denote the total heat content per unit volume of porous medium. Then the expression of de Vries (1958), generalized for a hysteretic soil, is

$$S_h = (C_d + c_p \rho_v \theta_a + c_\ell \rho_\ell \theta)(T - T_o) + L_o \rho_v \theta_a - \rho_\ell \int_{t_o}^{t} W \frac{d\theta}{d\tau} \, d\tau \quad (2.18)$$

in which

C_d is the heat capacity of dry soil plus roots, bulk volume basis [cal cm^{-3} $°K^{-1}$]

c_p is the specific heat of water vapor at constant pressure [cal g^{-1} $°K^{-1}$]

c_ℓ is the specific heat of liquid water [cal g^{-1} $°K^{-1}$]

T_o is the arbitrary reference temperature [°K]

L_o is the latent heat of vaporization, L, evaluated at $T = T_o$ [cal g^{-1}]

where

L is the latent heat of vaporization [cal g^{-1}]

and

W is the differential heat of wetting [cal g^{-1}]

The differential heat of wetting, W, is the amount of heat released when a small amount of free water is added to the soil matrix (Edlefson and Anderson, 1943). Note that C_d is taken to include all organic material, living roots in particular. The heat capacity of air is neglected.

Following the general approach of de Vries (1958), we tentatively express the heat flux as

$$\underline{q}_h = -\lambda_* \nabla T + L_o \underline{q}_v + c_p (T - T_o)\, \underline{q}_v + c_\ell (T - T_o)\, \underline{q}_\ell + c_\ell (T - T_o) \underline{q}_r \tag{2.19}$$

in which

$\underline{q}_h$ is the heat flux in the medium [cal cm^{-2} s^{-1}]

λ_* is the thermal conductivity of bulk medium [cal cm^{-1} s^{-1} $°K^{-1}$]

The first term in (2.19) is simple conduction in all phases of the medium, the second is latent heat transport (convection) in the vapor phase, while the remaining terms represent the sensible heat convected by vapor, soil liquid, and root water. Convection of air, as well as radiative transfer, is considered negligible.

The basic thermodynamic relation

$$L_o + c_p (T - T_o) = L + c_\ell (T - T_o) \tag{2.20}$$

may be used to convert Equation (2.19) to

$$\underline{q}_h = - \lambda_* \nabla T + L\underline{q}_v + c_\ell (T - T_o)\, \underline{q}_m \qquad (2.21)$$

where $\underline{q}_m$ has been defined in Equation (2.14). Let us substitute for $\underline{q}_v$ using (2.12). This yields

$$\underline{q}_h = - (\lambda_* + \rho_\ell L D^{\psi}_{Tv})\, \nabla T - \rho_\ell L D_{\psi v}\, \nabla \psi + c_\ell (T - T_o)\, \underline{q}_m \qquad (2.22)$$

The two terms multiplying ∇T are not strictly additive (de Vries, 1958), as was assumed in the statement of Equation (2.19). The two effects interact, resulting in an effective thermal conductivity, λ^{ψ}, which depends strongly on θ, as discussed in Chapter 3. We therefore replace Equation (2.22) with the following:

$$\underline{q}_h = - \lambda^{\psi} \nabla T - \rho_\ell L D_{\psi v}\, \nabla \psi + c_\ell (T - T_o)\, \underline{q}_m \qquad (2.23)$$

The substitution of Equations (2.18) and (2.23) into (2.1) yields the heat conservation equation,

$$C \frac{\partial T}{\partial t} + [L_o + c_p (T - T_o)]\, \theta_a \frac{\partial \rho_v}{\partial t}$$

$$+ [c_\ell \rho_\ell (T - T_o) - \rho_\ell W - c_p \rho_v (T - T_o) - L_o \rho_v] \frac{\partial \theta}{\partial t}$$

$$= \nabla \cdot [\lambda^{\psi} \nabla T + \rho_\ell L D_{\psi v}\, \nabla \psi - c_\ell (T - T_o)\, \underline{q}_m] \qquad (2.24)$$

in which

$$C = C_d + c_p \rho_v \theta_a + c_\ell \rho_\ell \theta$$

$$= \text{bulk heat capacity} \qquad [\text{cal cm}^{-3}\ ^\circ\text{K}^{-1}]$$

Two more equivalent forms of the heat equation may be generated by application of the mass conservation equation (2.17). Let us first assume that the temporal variations in the amount of water stored in the root system are negligible. Then, from Equation (2.16),

$$S = \nabla\cdot(\underline{q}_r/\rho_\ell) \tag{2.25}$$

This may be used to eliminate S from Equation (2.17) for use in the development below.

Multiplying Equation (2.17) by the quantity

$$\rho_\ell[L_o + c_p(T - T_o)]$$

and subtracting the result from Equation (2.24), we obtain

$$C\frac{\partial T}{\partial t} - (\rho_\ell L + \rho_\ell W)\frac{\partial\theta}{\partial t} = \nabla\cdot[\lambda^\psi\nabla T + \rho_\ell LD_{\psi v}\nabla\psi] - c_\ell\underline{q}_m\cdot\nabla T + L\nabla\cdot\underline{q}_m \tag{2.26}$$

Multiplying Equation (2.17) instead by $\rho_\ell c_\ell(T - T_o)$ and subtracting from Equation (2.24), we find a third variant of the heat equation, which is

$$C\frac{\partial T}{\partial t} + L\theta_a\frac{\partial\rho_v}{\partial t} - (\rho_v L + \rho_\ell W)\frac{\partial\theta}{\partial t} = \nabla\cdot[\lambda^\psi\nabla T + \rho_\ell LD_{\psi v}\nabla\psi] - c_\ell\underline{q}_m\cdot\nabla T \tag{2.27}$$

On the interior of a suitable space-time domain D, Equation (2.17) and either (2.24) or one of its variants, together with the

appropriate diagnostic equations, constitute a mathematical model of the coupled distributions of the soil state variables. A problem arises in the flux relations (2.14) and (2.23) when either a diffusivity or K is discontinuous in space, or in Equation (2.24), for instance, when some of the storage coefficients may be discontinuous. Assuming that the boundary conditions, the initial conditions, and the extraction term, S, are smooth functions, the only sources of discontinuities in the equations already derived are spatial discontinuities in θ resulting from the juxtaposition of different soil types. For convenience in dealing with this problem, we shall restrict D to be a domain on whose interior the soil properties are continuous. Such sub-domains may then be linked through the appropriate boundary conditions to describe an overall system.

2.2.3 The ψ-T System

It is standard practice to express moisture and heat flow equations in diffusion form. The equation usually used to describe water flow in a porous medium, which neglects temperature and vapor effects, is ordinarily written in terms of a single dependent variable, either ψ (e.g., Whisler and Klute, 1965) or θ (e.g., Klute, 1952). The classical heat equation is naturally expressed in terms of T. Philip and de Vries (1957), and then de Vries (1958), proposed coupled heat and mass conservation equations with θ and T as the dependent variables.

In this section, the conservation equations derived in the previous sections will be written with ψ and T as dependent variables.

In the following section, we will present the equations in terms of θ and T.

In the storage terms of Equation (2.17), both $\frac{\partial \theta}{\partial t}$ and $\frac{\partial \rho_v}{\partial t}$ are to be expressed in terms of $\frac{\partial \psi}{\partial t}$ and $\frac{\partial T}{\partial t}$. Expansion of $\frac{\partial \rho_v}{\partial t}$, using Equation (2.8), yields

$$\frac{\partial \rho_v}{\partial t} = \left.\frac{\partial \rho_v}{\partial \psi}\right|_T \frac{\partial \psi}{\partial t} + \left.\frac{\partial \rho_v}{\partial T}\right|_\psi \frac{\partial T}{\partial t} \tag{2.28}$$

Now consider the other time derivative, $\frac{\partial \theta}{\partial t}$. We assume that θ may be expressed as

$$\theta = \theta(\psi, T; \underline{R}; \underline{s}) \tag{2.29}$$

which is simply an inverse of Equation (2.5). Formal differentiation of (2.29) yields

$$\frac{\partial \theta}{\partial t} = \left.\frac{\partial \theta}{\partial \psi}\right|_T \frac{\partial \psi}{\partial t} + \left.\frac{\partial \theta}{\partial T}\right|_\psi \frac{\partial T}{\partial t} + \sum_{i=1}^{r} \frac{\partial \theta}{\partial R_i} \frac{\partial R_i}{\partial t} + \sum_{i=1}^{n_s} \frac{\partial \theta}{\partial s_i} \frac{\partial s_i}{\partial t} \tag{2.30}$$

(The variables to be held constant in a partial differentiation will be indicated only when they may be ambiguous due to the use of different sets of dependent variables.) In this study, the porous medium is assumed to be non-deforming, so the last summation is dropped. It is quite reasonable to assume the differentiability of (2.29) with respect to its arguments, so the first two terms in (2.30) are valid. The nature of $\underline{R}$, the wetting history, is one of the subjects of Chapter 3. Here, it is sufficient to note that $\underline{R}$ is constant as long

as a wetting reversal does not occur. When there is a reversal, $\underline{R}$ has a discontinuity. We must therefore restrict the use of (2.30) to time periods during which no reversal occurs. This summation is thus zero.

We now substitute the expansion

$$\frac{\partial\theta}{\partial t} = \left.\frac{\partial\theta}{\partial\psi}\right|_T \frac{\partial\psi}{\partial t} + \left.\frac{\partial\theta}{\partial T}\right|_\psi \frac{\partial T}{\partial t} \tag{2.31}$$

and Equation (2.28) into the basic mass conservation equation (2.17) to obtain

$$\left[\left(1 - \frac{\rho_v}{\rho_\ell}\right)\left.\frac{\partial\theta}{\partial\psi}\right|_T + \frac{\theta_a}{\rho_\ell}\left.\frac{\partial\rho_v}{\partial\psi}\right|_T\right]\frac{\partial\psi}{\partial t} + \left[\left(1 - \frac{\rho_v}{\rho_\ell}\right)\left.\frac{\partial\theta}{\partial T}\right|_\psi + \frac{\theta_a}{\rho_\ell}\left.\frac{\partial\rho_v}{\partial T}\right|_\psi\right]\frac{\partial T}{\partial t}$$

$$= \nabla\cdot[(K + D_{\psi v})\,\nabla\psi + D_{Tv}^{\psi}\,\nabla T + K\underline{k}] - S \tag{2.32}$$

This is a non-linear second-order partial differential equation relating the time derivatives and gradients of the two dependent variables, ψ and T, in an expression of the principle of mass conservation.

The same approach may be used on the various heat equations. They become (from 2.24)

$$\left\{C + [L_o + c_p(T - T_o)]\,\theta_a \left.\frac{\partial\rho_v}{\partial T}\right|_\psi + [c_\ell\rho_\ell(T - T_o) - \rho_\ell W - c_p\rho_v(T - T_o) - L_o\rho_v]\left.\frac{\partial\theta}{\partial T}\right|_\psi\right\}\frac{\partial T}{\partial t} + \left\{[L_o + c_p(T - T_o)]\,\theta_a \left.\frac{\partial\rho_v}{\partial\psi}\right|_T + [c_\ell\rho_\ell(T - T_o) - \rho_\ell W - c_p\rho_v(T - T_o) - L_o\rho_v]\left.\frac{\partial\theta}{\partial\psi}\right|_T\right\}\frac{\partial\psi}{\partial t}$$

$$= \nabla\cdot[\lambda^{\psi}\nabla T + \rho_\ell L D_{\psi v}\nabla\psi - c_\ell(T - T_o)\,\underline{q}_m] \tag{2.33}$$

(from 2.26)

$$\left[C - \rho_\ell (L + W) \left.\frac{\partial \theta}{\partial T}\right|_\psi\right] \frac{\partial T}{\partial t} - \left[\rho_\ell (L+W) \left.\frac{\partial \theta}{\partial \psi}\right|_T\right] \frac{\partial \psi}{\partial t}$$

$$= \nabla \cdot [\lambda^\psi \nabla T + \rho_\ell L D_{\psi v} \nabla \psi] - c_\ell \underline{q}_m \cdot \nabla T + L \nabla \cdot \underline{q}_m \qquad (2.34)$$

(from 2.27)

$$\left[C + L\,\theta_a \left.\frac{\partial \rho_v}{\partial T}\right|_\psi - (\rho_\ell W + \rho_v L) \left.\frac{\partial \theta}{\partial T}\right|_\psi\right] \frac{\partial T}{\partial t}$$

$$+ \left[L\,\theta_a \left.\frac{\partial \rho_v}{\partial \psi}\right|_T - (\rho_\ell W + \rho_v L) \left.\frac{\partial \theta}{\partial \psi}\right|_T\right] \frac{\partial \psi}{\partial t}$$

$$= \nabla \cdot [\lambda^\psi \nabla T + \rho_\ell L D_{\psi v} \nabla \psi] - c_\ell \underline{q}_m \cdot \nabla T \qquad (2.35)$$

Equations (2.32) through (2.35) are valid subject to the restrictions placed on the earlier equations from which they are derived, and to the assumptions required by the subsequent derivations. In particular, the domain of validity must contain no discontinuities in soil properties and no points at which a wetting reversal occurs during the time of interest. Note, however, that the wetting history may be discontinuous spatially.

All of the equations developed in this section for unsaturated media are equally valid for saturated media when elastic storage can be neglected. This condition is met when the saturated region communicates freely with an unsaturated region. In this case, many of the terms in Equations (2.32) through (2.35), including all of the storage terms in the moisture equation, go to zero.

The restriction that no wetting reversal occur is rather limiting. It can, however, be dealt with in an approximate way by discretizing the time domain and assuming that changes in wetting history occur in finite regions of space at particular points in time. The solution at the end of one time period provides the initial condition for the following period. Such an approximation is depicted schematically in Figure 2.1 for infiltration into a previously drying soil. The smooth and step-like solid curves are the actual and approximated characteristics of the point at which $\frac{\partial\theta}{\partial t} = 0$. Boundary and initial conditions are employed to match the regions along the vertical and horizontal dashed lines, respectively.

2.2.4 The θ-T System

As an alternative to the ψ-T formulation outlined above, we may express the heat and mass continuity equations in terms of θ and T. The LHS of Equation (2.17) becomes

$$\text{LHS} = \left(1 - \frac{\rho_v}{\rho_\ell} + \frac{\theta_a}{\rho_\ell}\left.\frac{\partial\rho_v}{\partial\theta}\right|_T\right)\frac{\partial\theta}{\partial t} + \left(\frac{\theta_a}{\rho_\ell}\left.\frac{\partial\rho_v}{\partial T}\right|_\theta\right)\frac{\partial T}{\partial t} \qquad (2.36)$$

The time derivative $\frac{\partial\rho_v}{\partial t}$ has been expanded above by assuming that, at each point inside the domain of interest, $\psi = \psi(\theta, T)$ (and therefore $\rho_v = \rho_v(\theta, T)$) is single-valued, i.e., there are no <u>temporal</u> discontinuities in wetting history, and that $\frac{\partial\psi}{\partial\theta}$ is finite.

We now consider the moisture fluxes. The gradient of ψ, appearing in the moisture flux expression, can be expanded in a fashion

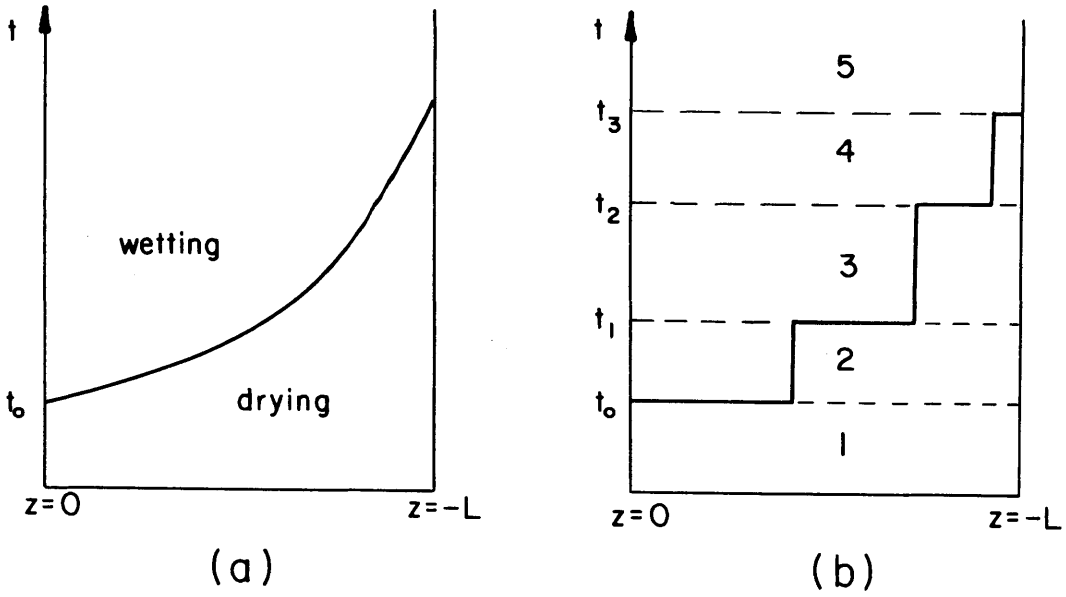


Figure 2.1

SCHEMATIC DIAGRAMS OF INFILTRATION, BEGINNING AT TIME t_o, INTO A PREVIOUSLY DRYING, HOMOGENEOUS SOIL COLUMN OF LENGTH L. (a.) BOUNDARY BETWEEN WETTING AND DRYING, SEPARATING TWO TIME-SPACE DOMAINS WITHIN WHICH EQS. (2.32) THROUGH (2.35) ARE VALID. (b.) AN APPROXIMATION OF THE WETTING BOUNDARY AND THE ASSOCIATED REGIONS (1-5) OF VALIDITY OF THE (ψ, T) EQUATIONS.

analogous to (2.30),

$$\nabla\psi = \left.\frac{\partial\psi}{\partial\theta}\right|_T \nabla\theta + \left.\frac{\partial\psi}{\partial T}\right|_\theta \nabla T + \sum_{i=1}^{r} \frac{\partial\psi}{\partial R_i} \nabla R_i + \sum_{i=1}^{n_s} \frac{\partial\psi}{\partial s_i} \nabla s_i \qquad (2.37)$$

Since the domain has been restricted to one in which $\underline{s}$ is continuous, the last term is well-defined. The third term requires that wetting history be continuous in <u>space</u>. In a non-hysteretic, homogeneous soil, (2.37) reduces to

$$\nabla\psi = \left.\frac{\partial\psi}{\partial\theta}\right|_T \nabla\theta + \left.\frac{\partial\psi}{\partial T}\right|_\theta \nabla T \qquad (2.38)$$

which is the relation used by Philip and de Vries (1957). An expression similar to (2.37), but neglecting the effects of temperature and inhomogeneity, was suggested by Childs (1964).

Substitution of (2.36) and (2.37) into the mass conservation equation (2.17) yields

$$\left(1 - \frac{\rho_v}{\rho_\ell} + \frac{\theta_a}{\rho_\ell} \left.\frac{\partial\rho_v}{\partial\theta}\right|_T\right) \frac{\partial\theta}{\partial t} + \left(\frac{\theta_a}{\rho_\ell} \left.\frac{\partial\rho_v}{\partial T}\right|_\theta\right) \frac{\partial T}{\partial t}$$

$$= \nabla\cdot\left\{\left[(K + D_{\psi v}) \left.\frac{\partial\psi}{\partial\theta}\right|_T\right] \nabla\theta + \left[(K + D_{\psi v}) \left.\frac{\partial\psi}{\partial T}\right|_\theta + D_{Tv}^{\psi}\right] \nabla T + (K + D_{\psi v})\, \underline{G} + K\underline{k}\right\} - S \qquad (2.39)$$

in which

$$\underline{G} = \sum_{i=1}^{r} \frac{\partial\psi}{\partial R_i} \nabla R_i + \sum_{i=1}^{n_s} \frac{\partial\psi}{\partial s_i} \nabla s_i \qquad (2.40)$$

In order to reformulate the heat equations, we first adopt a new expression for the heat flux,

$$\underline{q}_h = - \lambda^{\theta} \nabla T - \rho_\ell L D_{\theta v} \nabla\theta + c_\ell (T - T_o) \underline{q}_m \qquad (2.41)$$

in which λ^{θ} is analogous to λ^{ψ} defined earlier, but represents the effective thermal conductivity in the θ-T system. The individual components of (2.23) and (2.41) are therefore not identical, although their sums are. The diffusivity D_{θ_v} is the analogue of D_{ψ_v} and is given by

$$D_{\theta v} = D_{\psi v} \left.\frac{\partial\psi}{\partial\theta}\right|_T \qquad (2.42)$$

This is derived by expanding $\nabla\psi$ in Equation (2.12).

In the θ-T system, Equation (2.24) is

$$\left\{C + [L_o + c_p(T - T_o)]\ \theta_a \left.\frac{\partial\rho_v}{\partial T}\right|_\theta \right\} \frac{\partial T}{\partial t}$$

$$+ \left\{[L_o + c_p(T - T_o)]\ \theta_a \left.\frac{\partial\rho_v}{\partial\theta}\right|_T + c_\ell\rho_\ell(T - T_o) - \rho_\ell W \right.$$

$$\left. - c_p\rho_v(T - T_o) - L_o\rho_v \right\} \frac{\partial\theta}{\partial t}$$

$$= \nabla\cdot[\lambda^{\theta} \nabla T + \rho_\ell L D_{\theta v} \nabla\theta - c_\ell(T - T_o) \underline{q}_m] \qquad (2.43)$$

Equation (2.26) becomes

$$C \frac{\partial T}{\partial t} - (\rho_\ell L + \rho_\ell W) \frac{\partial\theta}{\partial t} = \nabla\cdot[\lambda^{\theta} \nabla T + \rho_\ell L D_{\theta v} \nabla\theta] - c_\ell \underline{q}_m\cdot\nabla T + L \nabla\cdot\underline{q}_m \qquad (2.44)$$

and Equation (2.27) is equivalent to

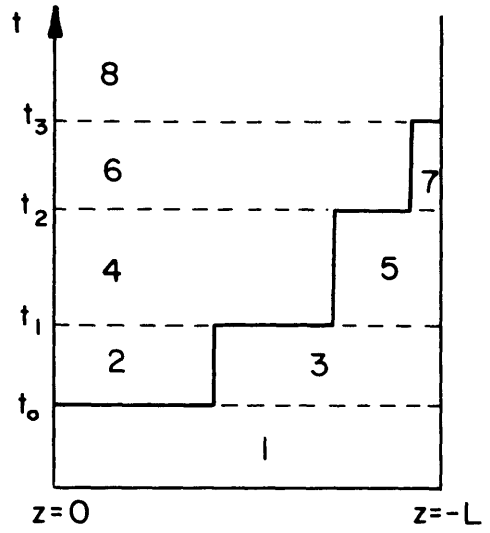


Figure 2.2

AN APPROXIMATION OF THE WETTING BOUNDARY AND THE ASSOCIATED REGIONS (1-8) OF VALIDITY OF THE (θ, T) EQUATIONS, (2.39) AND (2.43) THROUGH (2.45), FOR THE INFILTRATION PROCESS OF FIGURE 2.1.

$$\left(C + L\theta_a \left.\frac{\partial \rho_v}{\partial T}\right|_\theta\right) \frac{\partial T}{\partial t} + \left(L\theta_a \left.\frac{\partial \rho_v}{\partial \theta}\right|_T - \rho_v L - \rho_\ell W\right) \frac{\partial \theta}{\partial t}$$

$$= \nabla \cdot [\lambda^\theta \nabla T + \rho_\ell L D_{\theta v} \nabla \theta] - c_\ell \underline{q}_m \cdot \nabla T \qquad (2.45)$$

Equation (2.45) is the form of the heat equation that should correspond to Equation (16) of de Vries (1958). In fact, there is a slight discrepancy, which apparently results from that author omitting the term $\rho_\ell D_{Tv} \nabla T \cdot \nabla L$. It turns out that Equation (2.45) is more compact, combining two of de Vries' transport terms into one.

The domain of validity for the θ-T equations must have no discontinuities in soil properties, no spatial or temporal discontinuities in wetting history, and no regions in which $\frac{\partial \psi}{\partial \theta}$ is infinite. The last restriction limits applicability to unsaturated soils.

For the wetting process considered earlier (Figure 2.1), the necessary sub-domain divisions for use of the θ-T equations are depicted in Figure 2.2.

2.2.5 Simplifications for Relatively Moist Media

Many terms in the equations derived in the preceding sections are often negligible for practical analyses. In this section, a simplified system is derived by neglecting mass and energy transport in the vapor phase. As can be seen from an analysis of the transport coefficients (Chapter 3), this is a good approximation for sufficiently moist media; exact criteria will depend on the medium in question.

Neglecting mass storage and flux in the vapor phase, we

obtain the usual mass conservation equation of soil physics with an added sink term

$$\frac{\partial \theta}{\partial t} = \nabla \cdot (K \nabla \psi + K \underline{k}) - S \tag{2.46}$$

The expression for heat storage is

$$S_h = (C_d + c_\ell \rho_\ell \theta)(T - T_o) \tag{2.47}$$

and the heat flux is given by

$$\underline{q}_h = - \lambda_o \nabla T + c_\ell (T - T_o)(\underline{q}_m + \underline{q}_r) \tag{2.48}$$

where λ_o does not allow for vapor distillation. The heat conservation equation is thus

$$C \frac{\partial T}{\partial t} + c_\ell \rho_\ell (T - T_o) \frac{\partial \theta}{\partial t}$$

$$= \nabla \cdot (\lambda_o \nabla T) - c_\ell \rho_\ell (T - T_o) \nabla \cdot (\underline{q}_m / \rho_\ell + \underline{q}_r / \rho_\ell)$$

$$- c_\ell (\underline{q}_m + \underline{q}_r) \cdot \nabla T \tag{2.49}$$

Applying mass continuity in Equation (2.49), we obtain

$$C \frac{\partial T}{\partial t} = \nabla \cdot (\lambda_o \nabla T) - c_\ell (\underline{q}_m + \underline{q}_r) \cdot \nabla T \tag{2.50}$$

The expansions used in Sections 2.2.3 and 2.2.4 may be applied to these equations in order to obtain ψ-T and θ-T systems.

In Equations (2.48) and (2.49), λ_o may be replaced by an effective conductivity, λ^ψ, which includes heat transport by vapor

distillation. The effective value is the one that would be measured experimentally, and it is just as easily computed theoretically as is λ_o.

2.3 Boundary and Initial Conditions for Vertical Flow

2.3.1 Surface Boundary Conditions

We now consider the specification of boundary conditions for the equations developed in the previous sections. In particular, we consider a natural soil system in which horizontal flow is negligible. An idealized bare (i.e., non-vegetated) soil surface is assumed. (Accordingly, root water flow is now neglected.) The idealization consists of the following additional assumptions:

1. The land surface is approximately horizontal, and all mass and energy flow is normal thereto.
2. The interface between soil and atmosphere is a smooth flat plane. This surface is treated as a radiation source and sink, and as a momentum sink. (See Geiger, 1975, p. 18, for the extreme extinction depths of solar radiation in soil.) Since vapor flux is allowed within the soil, it is not necessary to confine evaporation (that is, phase change to vapor) to the surface.
3. Excess rainfall is stored on the surface up to a depth ε_o. Further excess is discharged from the system as surface runoff.
4. There is no lateral surface inflow to the system.

With the surface described above, it is possible to outline the various states (boundary conditions) that may come to be. Considering a sufficiently intense and lengthy rainstorm, followed by a sufficiently long period of high potential evaporation, we may distinguish the following states of the surface:

A. Precipitation begins. At first, infiltration proceeds at the precipitation rate and there is no surface retention or runoff. The moisture content near the surface of the soil increases.

B. When the soil surface reaches saturation, the rate of infiltration decreases, eventually approaching the saturation value of K. Ponding of water on the surface results.

C. Surface retention storage capacity is filled. All further excess precipitation is discharged as surface runoff.

D. Rainfall ends. Surface retention is depleted from above by evaporation and from below by infiltration.

E. Surface storage is exhausted. Evaporation from the soil proceeds at the potential rate.

F. The soil becomes so dry at the surface that moist air in equilibrium with it has relative humidity h significantly less than unity (Equation 2.8).

G. Eventually, another rainfall will return the system to state A.

Of course, not all of these states need occur for all meteorological input series. This conceptual model is similar to that proposed by Hillel (1977).

We now proceed to define mathematically the boundary conditions implied by states A through F above.

A. Unsaturated infiltration. This condition is simply expressed by

$$\underline{k} \cdot (\underline{q}_m / \rho_\ell)_{z=0} = - r \tag{2.51}$$

where the coordinate, z, is measured relative to the land surface, and

r = precipitation rate [cm s^{-1}]

The flux of heat into the soil is given by an energy balance at the land surface,

$$\underline{k} \cdot \underline{q}_h \Big|_{z=0} = - I_s + I_\ell + H - \rho_\ell c_\ell (T_r - T_o)\, r \tag{2.52}$$

in which

$\underline{q}_h \Big|_{z=0}$ = $\underline{q}_h$ at surface (given by Equation (2.23) or (2.41)) [cal cm^{-2} s^{-1}]

I_s = net shortwave radiation reaching surface from sky [cal cm^{-2} s^{-1}]

I_ℓ = net longwave radiation leaving surface [cal cm^{-2} s^{-1}]

H = sensible heat transfer upwards [cal cm^{-2} s^{-1}]

T_r = temperature of precipitation [°K]
(may be assumed equal to ambient air temperature)

B. Saturated infiltration with rainfall. The pressure head is non-negative and equal to the depth of ponded water,

$$\psi\Big|_{z=0} = \varepsilon \tag{2.53}$$

where ε is the water depth [cm], whose growth is described by

$$\frac{d\varepsilon}{dt} = r + (\underline{q}_m/\rho_\ell)_{z=0} \cdot \underline{k} \tag{2.54}$$

The heat flux boundary condition is derived by considering a heat balance for the ponded water, where the control volume has its top at the moving water surface and its bottom at an infinitesimally small depth beneath the ground surface.

Considering the shallow depth of the ponded water and the mixing induced by falling raindrops, the temperature of the water will be constant over z and equal to the soil temperature at z = 0. This holds well even after the rain has ceased (Geiger, 1975, p. 190).

$$\frac{\partial}{\partial t}\left[c_\ell \rho_\ell \varepsilon (T\Big|_{z=0} - T_o)\right] = c_\ell \rho_\ell (T_r - T_o) r + \underline{k} \cdot \underline{q}_h\Big|_{z=0} - H + I_s - I_\ell \tag{2.55}$$

where

I_s = the total shortwave radiation absorbed by the water and by the surface of the soil

I_ℓ = the net longwave back radiation leaving the water surface.

Combining (2.54) and (2.55), we obtain the surface heat flux

$$\underline{k}\cdot\underline{q}_h\Big|_{z=0} = c_\ell\rho_\ell\varepsilon\left.\frac{\partial T}{\partial t}\right|_{z=0} - c_\ell\rho_\ell(T_r - T\Big|_{z=0})\ r$$

$$+ c_\ell\underline{k}\cdot\underline{q}_m(T\Big|_{z=0} - T_o) + H - I_s + I_\ell \qquad (2.56)$$

C. Surface retention full. The pressure head is given by

$$\psi\Big|_{z=0} = \varepsilon = \varepsilon_o \qquad (2.57)$$

The heat balance yields

$$\underline{k}\cdot\underline{q}_h\Big|_{z=0} = \frac{\partial}{\partial t}[c_\ell\rho_\ell\varepsilon_o(T\Big|_{z=0} - T_o)] - c_\ell\rho_\ell(T_r - T_o)\ r$$

$$+ H - I_s + I_\ell + c_\ell\rho_\ell(T\Big|_{z=0} - T_o)\ R_s \qquad (2.58)$$

where R_s is the surface runoff rate [cm s^{-1}]. Applying the mass continuity principle to the ponded water, we obtain

$$\underline{k}\cdot\underline{q}_h\Big|_{z=0} = c_\ell\rho_\ell\varepsilon_o\left.\frac{\partial T}{\partial t}\right|_{z=0} - c_\ell\rho_\ell(T_r - T\Big|_{z=0})\ r$$

$$+ c_\ell\underline{k}\cdot\underline{q}_m(T\Big|_{z=0} - T_o) + H - I_s + I_\ell \qquad (2.59)$$

which is the same as Equation (2.56).

D. Surface retention depletion without rainfall. The pressure head at the ground surface is

$$\psi\Big|_{z=0} = \varepsilon \qquad (2.60)$$

where

$$\frac{d\varepsilon}{dt} = \underline{k}\cdot(\underline{q}_m/\rho_\ell)_{z=0} - e \tag{2.61}$$

and e [cm s^{-1}] is the actual rate of evaporation (vertical vapor flux above the ground surface), given by the flux-gradient relation

$$e = - K_a(\rho_a - \rho_v\Big|_{z=0})/\rho_\ell \tag{2.62}$$

in which

K_a = a vapor transport coefficient [cm s^{-1}]

ρ_a = water vapor density at height z = a [g cm^{-3}]

The heat balance now includes an evaporation term:

$$\frac{\partial}{\partial t}\,[c_\ell\rho_\ell\varepsilon(T\Big|_{z=0} - T_o)] = \underline{k}\cdot\underline{q}_h\Big|_{z=0} - [L_o + c_p(T\Big|_{z=0} - T_o)]\,\rho_\ell e$$
$$- H + I_s + I_\ell \tag{2.63}$$

In combination with (2.60) and (2.20), (2.61) leads to

$$\underline{k}\cdot\underline{q}_h\Big|_{z=0} = c_\ell\rho_\ell\varepsilon\,\frac{\partial T}{\partial t}\Big|_{z=0} + c_\ell(T\Big|_{z=0} - T_o)\underline{k}\cdot\underline{q}_m\Big|_{z=0} + H - I_s$$
$$+ I_\ell + \rho_\ell Le \tag{2.64}$$

E. Potential rate evaporation from soil surface. The mass flux condition is

$$\underline{k}\cdot(\underline{q}_m/\rho_\ell)_{z=0} = e \tag{2.65}$$

with e given by Equation (2.62).

The heat balance is

$$\underline{k}\cdot\underline{q}_h\Big|_{z=0} = [L_o + c_p(T\Big|_{z=0} - T_o)]\,\underline{k}\cdot\underline{q}_m\Big|_{z=0} + H - I_s + I_\ell \qquad (2.66)$$

or

$$\underline{k}\cdot\underline{q}_h\Big|_{z=0} = \rho_\ell Le + c_\ell\rho_\ell(T\Big|_{z=0} - T_o)e + H - I_s + I_\ell \qquad (2.67)$$

F. Soil-limited evaporation. Although this stage is often modeled differently than the previous one, the equations derived for stage E are also valid here. This is true because the concept of potential evaporation has not been introduced.

2.3.2 The Bottom Boundary Conditions

If there is an unconfined aquifer (saturated zone), then the lower boundary condition on moisture may be expressed

$$\psi\Big|_{z=z_s} = 0 \qquad (2.68)$$

where $|z_s|$ is the depth to the water table. Significant variation of z_s with time is a possibility in many systems. We may simulate this fluctuation by assuming a linear storage model for the aquifer

$$n\frac{dz_s}{dt} = -\underline{k}\cdot(\underline{q}_m/\rho_\ell)_{z=z_s} - \beta(z_s - z_o) \qquad (2.69)$$

in which

z_s = elevation of water table [cm]

β = groundwater discharge coefficient [s^{-1}]

z_o = base water table elevation [cm]

The last term in Equation (2.69) is lateral discharge from (to) the aquifer.

When it is known *a priori* that all infiltrated water eventually evaporates, and that there is no deep source of water for the soil zone, then the phreatic aquifer need not be modeled. In this case, the boundary condition is specified at a depth $|z_M|$, at which no moisture flow occurs,

$$\underline{k} \cdot \underline{q}_m \Big|_{z_M} = 0 \tag{2.70}$$

The lower heat boundary condition may be specified at a depth $|z_T|$ where $\nabla\psi$ and ∇T are negligible (see 2.23)

$$\underline{k} \cdot \underline{q}_h \Big|_{z=z_T} = c(T - T_o)\, \underline{k} \cdot \underline{q}_m \Big|_{z=z_T} \tag{2.71}$$

In general, either $|z_M|$ or $|z_T|$ may be greater.

2.3.3 Initial Conditions

The initial values of $\psi(z)$ or $\theta(z)$, as well as $T(z)$, must be specified over the appropriate range of z. In addition, for hysteretic soils, the relevant wetting history information $\underline{R}(z)$ must be given. The initial conditions are all specified at some time $t = t_o$ at the beginning of the time domain of interest.

2.3.4 Sub-domain Matching Conditions

Since we have restricted the governing equations to be

applicable only for domains in which certain continuity conditions are satisfied, it may not be possible to describe a particular problem in terms of a single set of dependent variables defined over the entire region. Instead, it will be necessary to apply the appropriate equations within each valid sub-domain and to match these individual solutions by applying physical principles along the discontinuity. As an example, consider the case of a texturally stratified soil in which a discontinuity in soil properties exists along some surface Γ. The curve Γ splits the domain of interest into two sub-domains, D_1 and D_2. Then we define ψ (or θ) and T fields within each domain and invoke the heat and mass equations.

$$H(\psi^{(1)}, T^{(1)}) = 0 \quad \text{on} \quad D_1 \tag{2.72a}$$

$$M(\psi^{(1)}, T^{(1)}) = 0 \quad \text{on} \quad D_1 \tag{2.72b}$$

$$H(\psi^{(2)}, T^{(2)}) = 0 \quad \text{on} \quad D_2 \tag{2.72c}$$

$$M(\psi^{(2)}, T^{(2)}) = 0 \quad \text{on} \quad D_2 \tag{2.72d}$$

where the superscript indexes the sub-domain and H and M represent the conservation equations. Portions of the boundaries of D_1 and D_2 will already have boundary conditions specified, since they lie on the boundary of D. Along Γ, however, additional conditions must be applied to link the sub-domains. These are obtained by requiring the continuity of temperature and pressure head, as well as normal mass and heat fluxes, across Γ. Mathematically, on Γ,

$$T^{(1)} = T^{(2)} \tag{2.73a}$$

$$\psi^{(1)} = \psi^{(2)} \tag{2.73b}$$

$$\underline{q}_h^{(1)} \cdot \underline{n} = \underline{q}_h^{(2)} \cdot \underline{n} \tag{2.73c}$$

$$\underline{q}_m^{(1)} \cdot \underline{n} = \underline{q}_m^{(2)} \cdot \underline{n} \tag{2.73d}$$

where $\underline{n}$ is the unit vector normal to Γ.

The procedure described above can be generalized to any number of sub-domains. Furthermore, Γ may be a moving boundary, such as a surface on which a wetting reversal is occurring.

The problem of a time discontinuity at some time t_1 (where initial conditions have been given at time $t_o < t_1$) is treated by solving the system from $t = t_o$ to $t = t_1$ and then starting from $t = t_1$ with the initial conditions given by the final solution from the previous time period.

2.4 Discussion of ψ- versus θ-based Systems

There are several significant differences between the ψ-based and θ-based equations. These are briefly reviewed here.

Applicability. The use of ψ preserves the validity of the derived equations in saturated media. The θ-based equations employ the derivative $\frac{\partial \psi}{\partial \theta}$, which becomes infinite in saturated media, so ψ must then be used. Note that saturation may occur not only beneath the water table, but also in the cases of surface ponding and minor, temporary "perched water tables" that result from certain media discontinuities.

Sub-domain transitions. Whenever $\underline{R}$ or $\underline{s}$ is discontinuous, the separate sub-domains must be matched as described in Section 2.3.4. The matching

condition $\psi^{(1)} = \psi^{(2)}$ is mathematically simpler when ψ is the dependent variable than when θ is. Note also that for discontinuous $\underline{s}$, θ will itself be discontinuous, which is an undesirable attribute for a dependent variable.

Flux expressions. The mass flux expressions (and the convective heat flux terms) are somewhat simpler when $\nabla\psi$ instead of $\nabla\theta$ is used (e.g., Equation (2.32) versus (2.39)). This is due to the term $\underline{G}$, which represents that part of the gradient of ψ resulting from gradients in $\underline{s}$ and $\underline{R}$. However, if $\underline{s}$ and $\underline{R}$ are constant within a sub-domain, this distinction disappears.

Storage expressions. Moisture and heat storage terms are more conveniently expressed in terms of θ. When ψ is used, time changes in θ must be separated into those due to $\left.\frac{\partial\psi}{\partial t}\right|_T$ and $\left.\frac{\partial T}{\partial t}\right|_\psi$ (e.g., Equation (2.25) versus (2.38)).

Finally, we note that the choice between ψ and θ is not necessarily required; in theory, we can have "the best of both worlds." Retention of both θ and ψ as well as ρ_v in the particular terms where they arise naturally from the physics is also correct (e.g., Equation (2.15)). Exclusive use of one or the other in the derivatives is a mathematical technique that results in equations that may submit more readily to analysis.

In the following chapters, ψ will be treated as a dependent variable, and θ will be considered a variable parameter whose value is determined by the state variables $(\psi, T, \underline{R})$ through a diagnostic relation.

Chapter 3

THE STORAGE AND TRANSPORT COEFFICIENTS OF WATER AND HEAT IN POROUS MEDIA

3.1 Introduction

The various conservation equations derived in Chapter 2 are different expressions of only two independent prognostic (i.e., containing time derivatives) equations, those of mass and heat conservation. They contain more than two variables, however, since the storage and transport coefficients are not constants, but are functions of the state variables. This chapter contains the definitions of the diagnostic (i.e., lacking time derivatives) equations, which express the dependence of the conservation equation coefficients on a set of state variables.

Due to the complications of hysteresis, more than two state variables are required to describe the system dynamics properly. Indeed, the state set is potentially infinite-dimensional. This issue is dealt with first. A proposed model of the water retention process gives rise to the appropriate state set and, in effect, provides the extra prognostic relation necessary to predict the time path of the state set.

Having selected a complete state set, we proceed to define, as functions of the state variables, the other variables appearing in the conservation equations. These definitions constitute the necessary set of diagnostic equations.

In order to facilitate the computer modeling of moisture and heat transport in the soil, we seek analytical expressions defining the

diagnostic relations. Such formulations allow the experimental variation of soil properties in a simulation model with a minimum of effort. Given good mathematical models of the relevant physical properties of the soil, the behavior of all types of soils can be reproduced simply by varying a finite number of parameters.

The parameters of these models may also provide an objective means for hydrologic classification of soils, since there is a one-to-one correspondence between a particular soil and its parameters in a well-defined model. However, the usefulness of such a scheme would not be great unless the parameters could be related directly to some intrinsic, measurable characteristic of the soil. Direct physical significance of the parameters might therefore be considered another attribute of a good soil parameterization.

The macroscopic (hence, integrated, derived or statistical) nature of many of the important variables makes purely theoretical analyses of them difficult or impossible. Conceptual and empirical models therefore arise out of necessity. For the purposes of simulation, any distinction along these lines is immaterial. Provided a model is sufficiently complex to reproduce reasonably well all types of soil behavior, yet is simple enough to be manageable computationally, its origin is not critical. In general, a theoretical model is used here when a realistic one exists. Otherwise, a conceptual or empirical model is used.

3.2 The Retention of Water in a Porous Medium

3.2.1 A Qualitative Physical Description of Water Retention

The matric (or pressure) potential, ψ, is a variable employed in soil science to describe the energy level of liquid water within an unsaturated porous medium. The quantity $g\psi$ is the amount of work that must be done to move a unit mass of water, isothermally and reversibly, from a porous medium to the free (flat surface) state. The datum for ψ is thus conventionally specified as free liquid water at a fixed temperature, T_o, and at atmospheric pressure. Units of ψ are [cm].

The sum of matric plus gravity head may be viewed as a potential, the gradient of which is proportional to the force that causes liquid flow. This mechanical definition is suggestive of, and consistent with, the flux-gradient equation (2.3) when inertial terms are small, the net driving force on a "particle" of water being balanced by (velocity-proportional) viscous dissipation of energy (Philip, 1960).

In thermodynamic terms, ψ is related to the specific differential Gibbs free energy function. Its gradient is thus a measure of the magnitude of local thermodynamic disequilibrium.

The forces ordinarily considered to be the determinants of ψ in unsaturated media are capillarity, which is manifested in the pressure differences across curved air-water interfaces under surface tension, and adsorption, which involves the relatively short-distance interaction of water with the surface of the solid (mineral) phase of the medium. The magnitudes of these forces are determined by the microscopic distribution of water in the medium, by the temperature, and by the nature of the

medium itself. The term "matric" potential satisfactorily accounts for both capillarity and adsorption as inseparable facets of the interaction of water with the soil matrix.

In saturated media, the term "pressure" potential is preferable, being more justifiable physically. We follow convention in using the same symbol, ψ, in saturated media. The resulting continuity between saturated and unsaturated regimes is quite convenient.

In relatively moist media, the effect of capillarity is dominant in determining ψ. Only the largest pores are air-filled, and the air-water interface has relatively small curvature. The pressure in the water just beneath the interface is (Bear, 1977)

$$p_w = p_a + \frac{2\sigma}{r_c} \tag{3.1}$$

in which

p_w =	pressure in water	[dynes cm^{-2}]
p_a =	pressure in air	[dynes cm^{-2}]
σ =	interfacial surface tension	[dynes cm^{-1}]
r_c =	harmonic mean radius of curvature of the interface, negative for a concave water surface	[cm]

In this situation, ψ is given by

$$\psi = p_w/\gamma = \frac{2\sigma}{r_c\gamma} \tag{3.2}$$

where

$$\gamma = \rho_{\ell} g = \text{specific gravity of liquid water} \qquad [\text{dyne cm}^{-3}]$$

and, in accordance with the definition of a datum for ψ, we have set $p_a = 0$. (It is assumed that the air pressure within the soil is in static equilibrium with the atmosphere.)

The amount of water retained at a given level of ψ in the capillary regime is thus determined by the distribution of the larger pore sizes. It follows that soil structure - the nature of aggregation of the soil - is a strong factor in determining the relation between ψ and θ for large θ.

As water is removed from the medium, the remaining water is, on the average, increasingly closer to the soild soil surfaces. The effect of adsorption thus takes over at low values of θ (and of ψ). In the adsorption regime, the moisture content at fixed ψ for any soil is correlated with the specific surface of the medium and can therefore be considered a function of soil texture.

Hillel (1971), on the basis of these considerations, describes how soil moisture characteristics (functions $\psi(\theta)$ for a particular soil at a fixed temperature and fixed wetting history) typically differ for various soils. The influence of texture is shown in Figure 3.1. For fixed ψ, the clayey (high specific surface) soil retains more water than a soil with fewer fine particles. In contrast, a well-sorted sand loses practically all of its water when a critical pressure ($\gamma\psi$) corresponding to the characteristic pore size is reached. Note also that heavier (more clay-like) soils tend to have larger porosities.

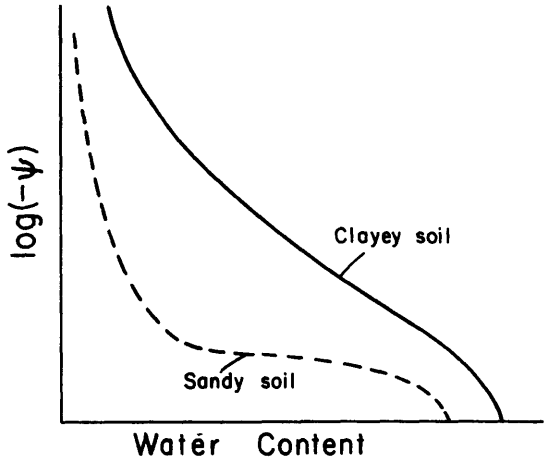


Figure 3.1

THE INFLUENCE OF SOIL TEXTURE ON MOISTURE RETENTION
(After Hillel, 1971.)

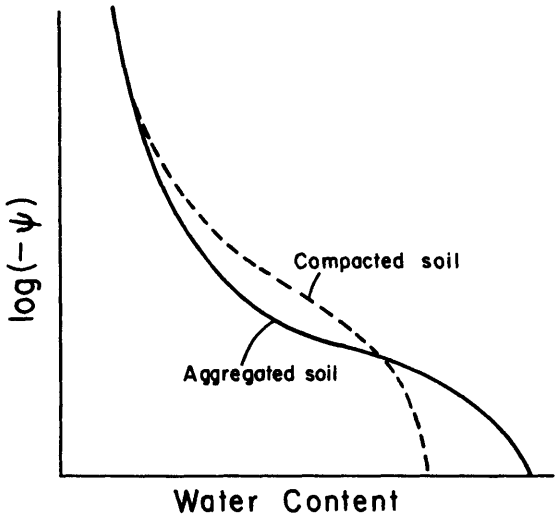


Figure 3.2

THE INFLUENCE OF SOIL STRUCTURE ON MOISTURE RETENTION
(After Hillel, 1971.)

The influence of soil structure (Figure 3.2) is limited to the capillary regime, since a rearrangement of the soil particles of a given soil cannot result in a texture change. A well-aggregated soil behaves somewhat like a sand in the capillary regime, while a compacted, structureless version of the same soil exhibits a more gradual slope of the moisture characteristic.

The value of ψ at the boundary between the capillary and adsorption regimes, if indeed such a boundary can be defined, has not been clearly determined. Miller and Miller (1955) suggest that the capillary theory of soil water is valid "at least in the coarse-silt-to-sand range, assuming agriculturally useful levels of available water are present." This qualification implies that ψ is above the wilting point, about -15,000 cm. This corresponds to a pF of 4.2, where pF is defined by

$$pF = \log_{10}(-\psi), \quad \psi \text{ in cm.} \tag{3.3}$$

Buckman and Brady (1969) place the division between capillary and hygroscopic (adsorbed) water at about pF = 4.5. Hillel (1971) states that capillarity is dominant below pF = 3.0, while adsorption is increasingly important at higher pF.

Carman and Raal (1951) observed that the condensation and adsorption of freon gas on packed (porous) powders exceeded that on loose powders for pF up to around 6.0. They interpreted the data as implying that capillarity must also play a role in retention even at such large

pF. Philip and de Vries (1957) extended this conclusion, assuming that the results held for water and that they implied that capillarity fully determines moisture retention for pF below about 6.0.

McQueen and Miller (1974) argue that the probable absence of tensile strength, especially in natural water, is reason to limit the validity of capillary theory to positive absolute pressures. The probability of vaporization at low, positive absolute pressures further limits the theory. The resultant conclusion is that capillarity is important only below pF $\simeq$ 3.0.

In actuality, the interaction of capillarity and adsorption is usually important (Edlefson and Anderson, 1943). Close to the soil particles, the adsorptive force fields are strong and, as a result, the pressure is "hydrostatically" distributed. The total (capillary plus adsorption) potential is constant locally, though the individual components vary over short (microscale) distances. Pressure will therefore be lowest at the air-water interface (i.e., farthest from the solid surface), and this is where the capillary formula is most likely to hold.

Since it is not clear, say, what range of pF may be treated using capillary theories, we shall be careful not to assume a particular model of soil behavior without attempting to find experimental justification. It will be seen, however, that the lack of knowledge concerning real soil properties necessitates the extrapolation of theoretical models beyond the ideal conditions of their formulation.

3.2.2 Hysteresis in the Moisture Retention Process

Hysteresis of the soil moisture characteristic is a non-uniqueness of the "functions" $\psi(\theta)$ and $\theta(\psi)$ for a particular soil at fixed temperature. When a completely water-saturated porous medium is dried continuously to complete dryness, the matric potential decreases monotonically from zero to some value near $\psi = -10^7$ (pF = 7). (Although the exact maximum pF may depend on the particular soil and the definition of complete dryness, we may arbitrarily define $\theta = 0$ at pF = 7.0 and consider any liquid that is more tightly bound to be part of the solid phase of the medium. This approach is justified since the soil is never subjected to stresses greater than those equivalent to pF = 7, corresponding to a relative humidity, by Equation (2.8) of about 10^{-3}.) This initial drying process is depicted by curve A in Figure 3.3.

Upon re-wetting (curve B), it is observed that curve A is not re-traced. The liquid content during re-wetting is less than that during drying for a large range of ψ. In particular, for $\psi = 0$, the soil does not return to full saturation, as the presence of entrapped air reduces the pore space available for water. Only after a long period of time (on the order of weeks) does the entrapped air dissolve and diffuse out of the system, allowing complete water-saturation (Adam and Corey, 1968).

If the soil is re-dried before the air is removed, a curve like C is traced. Subsequent complete drying and wetting proceeds along the cycle of curves B and C, which are termed the main wetting and drying curves, respectively. When wetting reversals occur anywhere other than at the common endpoints of curves B and C, they result in scanning curves,

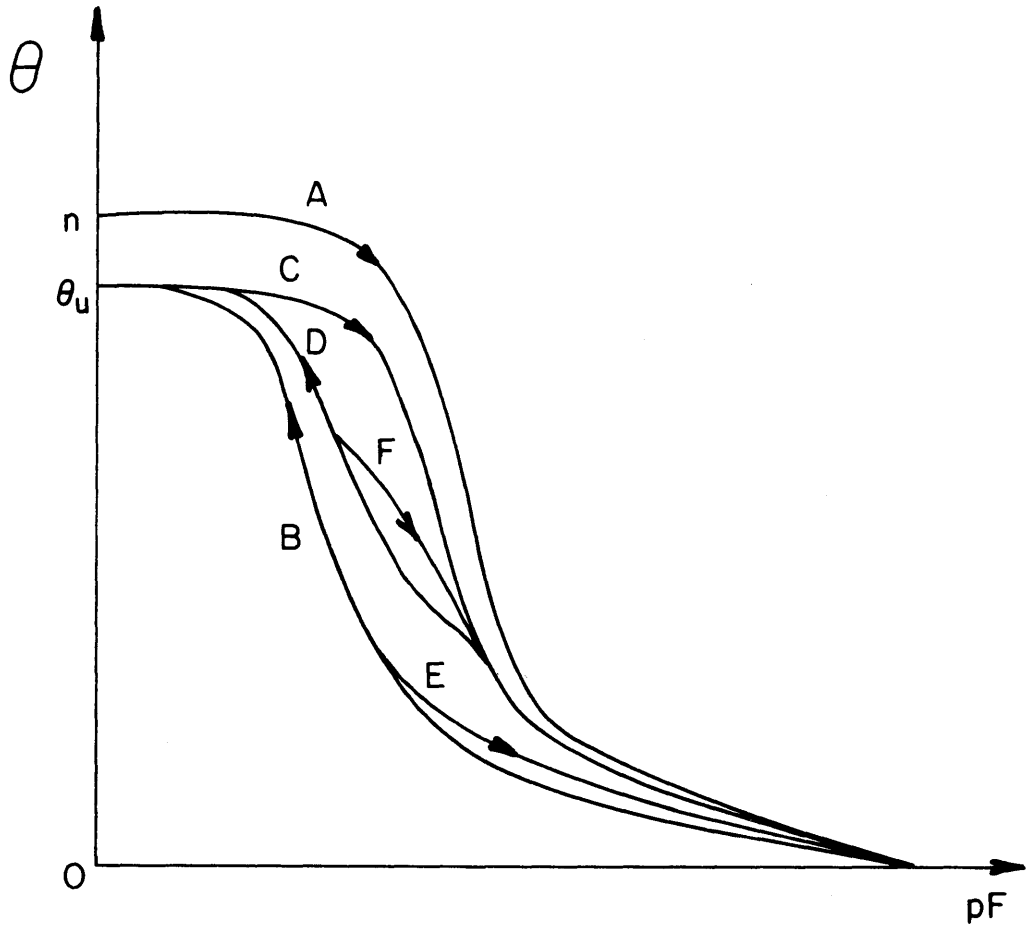


Figure 3.3

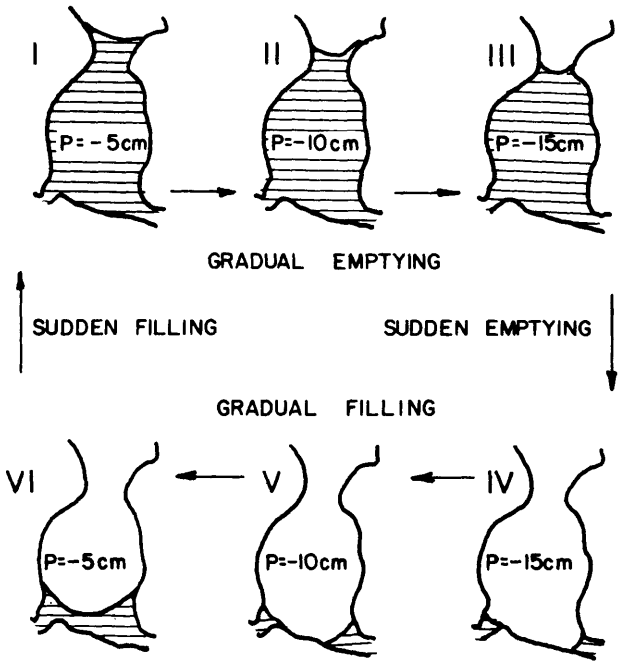
THE HYSTERETIC SOIL-WATER RETENTION PROCESS

A first drying curve
B main wetting curve
C main drying curve
D primary wetting curve
E primary drying curve
F secondary drying curve

such as D, E, and F in Figure 3.3. Curves D and E are primary wetting and drying (scanning) curves, while F is a secondary drying (scanning) curve. It is apparent from Figure 3.3 that the relation between moisture content, θ, and matric potential, ψ, at any time is dependent upon the wetting history of the medium.

Several explanations for the hysteretic behavior of the soil moisture characteristic have been suggested. The most popular model, which has been quite successful, explains the phenomenon in terms of capillarity and the so-called ink-bottle effect. In simplest terms, it states that at least some pores (relatively large intergranular voids connected by smaller passages) drain and refill at different capillary pressures (hence, matric heads). Miller and Miller (1956) recognized this effect as a natural implication of the capillary theory of moisture retention in soils.

Figure 3.4a shows how hysteresis in a single pore could occur, and Figure 3.4b exhibits a hypothetical pair of wetting and drying curves for the pore. The radius of the pore corresponds to $\psi = -5$ cm. When ψ reaches this value, the pore suddenly fills and the air-water interface migrates up into the next pore, where a new equilibrium is reached. The "neck" connecting these two pores has a radius corresponding to $\psi = -15$ cm, so a similar jump (termed a Haines jump after the investigator who observed the phenomenon experimentally) occurs when ψ is lowered to that value. The aggregated (macroscopic) interaction of all such pores then determines the behavior of the medium, as shown



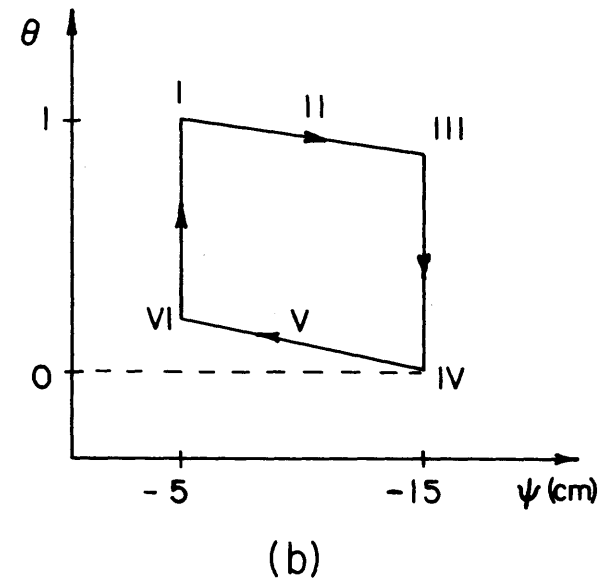


Figure 3.4

(a) "INK BOTTLE" HYSTERESIS IN A SINGLE PORE. (After Miller and Miller, 1956.); (b) THE ASSOCIATED "MOISTURE RETENTION CURVE" FOR THE SINGLE PORE.

in Figure 3.3.

Hillel (1971) mentions three other possible contributors to hysteresis. The first is the contact angle (or raindrop) effect, which is the dependency of the angle of contact between the air-water interface and the solid phase on the motion of the interface. Entrapped air and swelling and shrinking are further sources of hysteresis.

3.2.3 An Empirical Model of the Main Wetting Curve

For the purpose of simulation, we seek a representation of the hysteretic soil moisture absorption/desorption process discussed in the previous sections. In this section, a mathematical expression for the main wetting curve (curve B of Fig. 3.3) is chosen. In the next section, a model that expresses the main drying curves and all scanning curves in terms of the main wetting curve is presented. This model of Mualem (1977) is modified in this study to account for temperature effects to the first order.

Many empirically-based analytical forms for the isothermal soil moisture characteristic have been proposed. A representative, though far from exhaustive, sampling of them is reviewed here.

Brooks and Corey (1964, 1966), having analyzed the desorption curves for many consolidated rock samples, found that the relation between ψ and θ could be expressed as

$$\psi = \psi_b \left(\frac{\theta - \theta_r}{n - \theta_r} \right)^{-1/\lambda} \qquad \theta > \theta_r \qquad (3.4)$$

in which

ψ_b = the bubbling potential [cm]

θ_r = the residual moisture content

λ = a fitted parameter

The empirical constants ψ_b, θ_r, and λ have varying degrees of physical significance. The potential, ψ_b, is the value of ψ at which air is first drawn through the sample during dewatering in the laboratory. It is important to note that as $\theta \to n$, $\psi \to 0$, so Eq. (3.4) should not be used when $|\psi| < |\psi_b|$. The residual moisture content (related to θ_k of Chapter 2) is a measure of the amount of water retained at some arbitrary value of pF. In practice, the value of θ_r is simply chosen to minimize, in some sense, the error in Equation (3.4).

Since Eq. (3.4) predicts that $\psi \to -\infty$ as $\theta \to \theta_r$, this model is unusuable for $\theta \simeq \theta_r$ or smaller. For the data of Brooks and Corey (1966), this neglected water fills 10 to 30% of the pore space, a consequence of the fact that ψ was not lowered beyond -500 cm in their experiments.

Mualem (1976a) fitted the published data for 45 soils to the Brooks and Corey model. Residual moisture contents ranged from 0.01 to 0.28, but were mostly less than 0.10. In general, Mualem's data included lower values of ψ, which reduced the value of θ_r that would fit the data. Nevertheless, the selection of model parameters was apparently biased by the varying degree of data in the low θ range. Typical values of model parameters for the entire (pF < 7) characteristic are therefore not available, and Eq. (3.4) has not been extensively verified for low moisture contents.

Equation (3.4) is extremely useful nevertheless. Its simple form allows the integration of various relations (to be discussed later) in order to obtain a relative hydraulic conductivity function in a manageable mathematical form. The various analytic expressions are useful in exact solutions of the isothermal liquid flow problem. Eagleson (1978), for instance, has used a soil model based, in part, on Eq. (3.4) to approximate analytically the long-term average of moisture transfers across the ground surface.

King (1965) included the region of water content near saturation in his model of the main wetting and drying curves. The proposed relation is

$$\theta = n\delta\left[\frac{\cosh[(\psi/\psi_o)^{\beta} + \varepsilon] - \gamma}{\cosh[(\psi/\psi_o)^{\beta} + \varepsilon] + \gamma}\right] \tag{3.5}$$

in which δ, ψ_o, β, ε, and γ are the parameters. Using this formula, King (1965) fit data well for several soil types, but for $|\psi|$ only up to 100 cm. The form of Eq. (3.5) brings analytical dividends in that it is invertible and differentiable. Gillham, et al. (1976) used a modified form of Equation (3.5) in a computer simulation. Serious drawbacks of the model include the number of parameters and the difficulty of determining them for a given soil, as well as the limited range of ψ over which it has been tested.

McQueen and Miller (1974) studied the relationship between ψ and θ for pF up to 7 (completely dry media). They concluded that pF can be represented empirically as a piecewise linear function of θ for values of θ not near saturation. The three segments proposed are as

follows:

1) pF 5.0 - 7.0, "tightly adsorbed" segment

2) pF 2.5 - 5.0, adsorbed film segment

3) pF 0.0 - 3.0, capillary segment.

From statistical analyses of hundreds of samples, McQueen & Miller (1974) hypothesized the existence of common pF-axis intercepts for each linear segment for all soils. Estimated values of these intercepts are 7.0, 6.25, and 2.9 for segments 1, 2, and 3, respectively. It is also assumed that segments 1 and 2 join at pF = 5.0, where clay hydration occurs. If the absence of data for pF less than 2.5, it is suggested that segments 2 and 3 be joined at pF = 2.5. The curve near saturation is to be drawn in subjectively, using any available information about ψ_b. As presented, the model can be used to define completely the moisture characteristic on the basis of a single data point, preferably in the range between pF = 2.5 and pF = 5.0 (e.g., the permanent wilting percentage, pF = 4.2).

Disadvantages of the piecewise log-linear model are its discontinuous slope, $\frac{d\psi}{d\theta}$, and the probable over-specification of the curve that was necessary in order to achieve the goal of minimal data requirements.

The proposed form of the main wetting curve to be used in this study is based on the approach of McQueen and Miller (1974) in that it makes use of the approximate linear relation between θ and pF over certain ranges. In its simplest form, the proposed model may be written

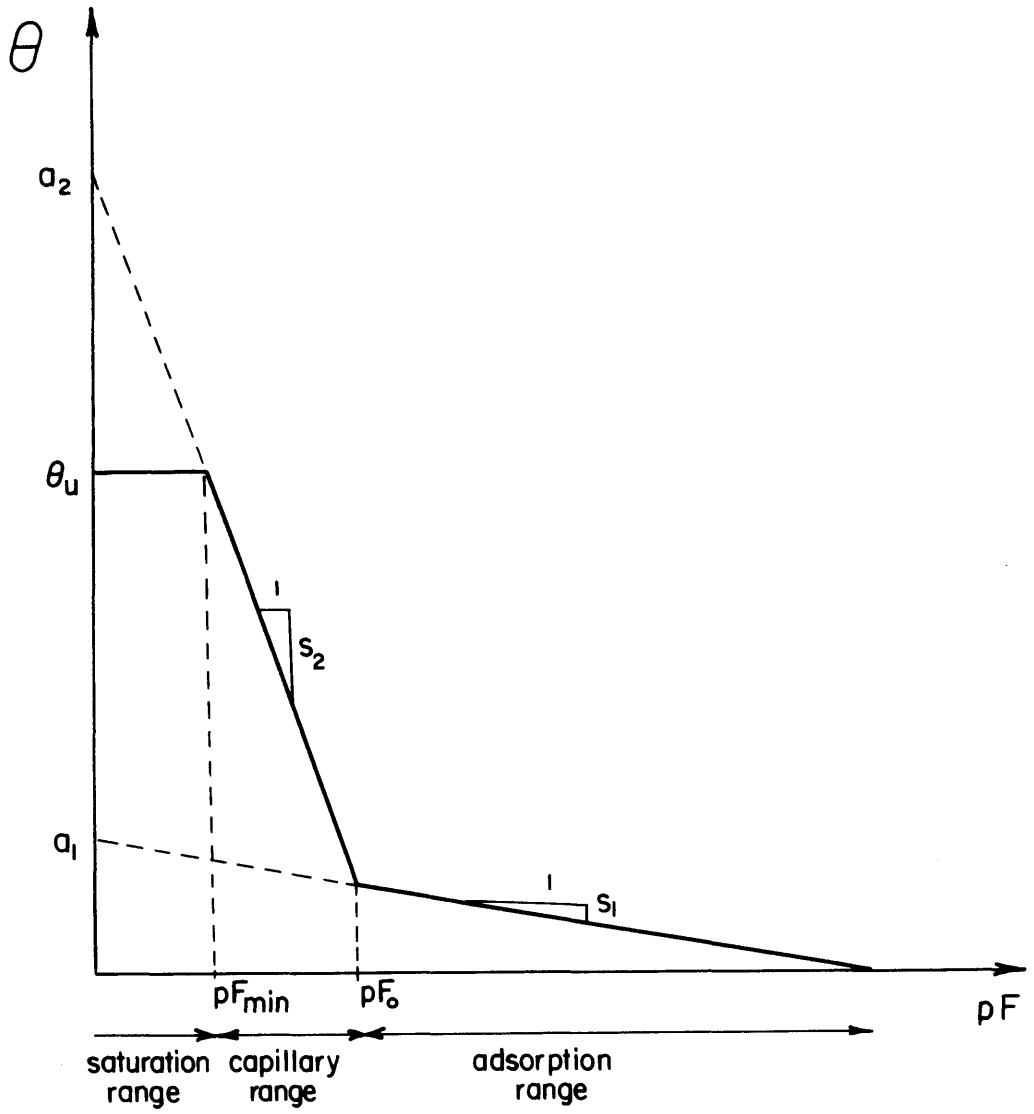


Figure 3.5

A PIECEWISE LINEAR RELATION BETWEEN θ AND pF

$$\theta_w(pF) = \begin{cases} a_1 - s_1 \cdot pF & pF_o \leq pF \leq 7 \\ a_2 - s_2 \cdot pF & pF_{min} \leq pF \leq pF_o \\ \theta_u & pF \leq pF_{min} \end{cases} \quad (3.6)$$

The significance of the various parameters is clear from Figure 3.5. Recall that θ_u is somewhat less than n (Fig. 3.3). The results of Mualem (1974) support a value of 0.9 for θ_u/n.

If desired, it is possible to introduce continuity and an arbitrary degree of curvature at the points pF_o and pF_{min} by generalizing Eq. (3.6) to the form

$$\theta_w(pF) = \frac{1}{M} \ln[e^{M(a_1 - s_1 pF)} + e^{M(a_2 - s_2 pF)}] - \frac{1}{M'} \ln[e^{M'(a_2 - s_2 pF)} + e^{M'\theta_u}] + \theta_u \quad (3.7)$$

In the limit as M and M' go to infinity, Eq. (3.7) becomes exactly equivalent to (3.6). Equation (3.7) might be preferred to (3.6) because of its increased versatility and its continuous differentiability at pF_o and pF_{min}.

The curvature at pF_{min} is a relatively small effect, and the saturated region is conveniently modeled with $\frac{\partial\theta}{\partial\psi}$ equal to zero. For these reasons, we let M' go to infinity in this study. The main wetting curve is then described by

$$\theta_w(pF) = \begin{cases} \frac{1}{M} \ln[e^{M(a_1 - s_1 pF)} + e^{M(a_2 - s_2 pF)}] & pF_{min} \leq pF \leq 7 \\ \theta_u & pF \leq pF_{min} \end{cases} \quad (3.8)$$

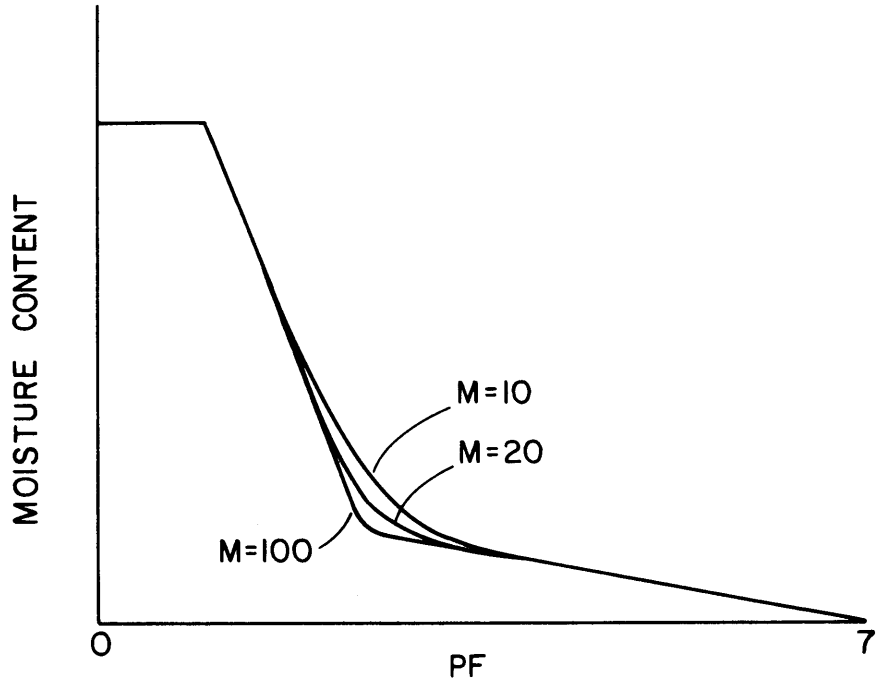


Figure 3.6

BEHAVIOR OF EQ. (3.9) FOR DIFFERENT VALUES OF THE CURVATURE PARAMETER, M

Equation (3.8) is expressible, redefining the constants, as

$$\theta_w(pF) = \begin{cases} \frac{1}{M} \ln[p_1(-\psi)^{q_1} + p_2(-\psi)^{q_2}] & pF_{min} \leq pF \leq 7 \\ \theta_u & pF \leq pF_{min} \end{cases} \quad (3.9)$$

in which

$$p_1 = e^{Ma_1} \qquad p_2 = e^{Ma_2}$$
$$q_1 = -s_1 M \log e \qquad q_2 = -s_2 M \log e \quad (3.10)$$

A typical set of curves is shown for different values of M in Figure 3.6.

Values of the parameters are determined easily from a minimal amount of information.

3.2.4 A Conceptual Model of Hysteresis

In recent years, a series of papers (Mualem, 1973, 1974, 1977; Mualem and Dagan, 1975) has described a set of models that may be used to model mathematically the phenomenon of hysteresis in the soil moisture retention process. The basic conceptual model (Mualem, 1974) accounts for the capillary hysteresis effect described in Section 3.2.2.

Mualem (1974) hypothesized that a porous medium could be modeled as a continuous set of pore groups. Each pore group is defined by r, the radius of the pore openings, and ρ, the radius of the pores themselves. The relative volume of the medium occupied by a pore group is given by the distribution function $f(r, \rho)$. That is, $f(r, \rho)\, dr d\rho$ is the proportion of the bulk medium occupied by the pore group having opening sizes between r and r + dr and having pore radii between ρ and $\rho + d\rho$.

Following Mualem (1974), we normalize r and ρ to a zero-one interval,

$$\bar{r} = \frac{r - R_{min}}{R_{max} - R_{min}} \tag{3.11}$$

$$\bar{\rho} = \frac{\rho - R_{min}}{R_{max} - R_{min}} \tag{3.12}$$

The values R_{min} and R_{max} are thus related to full saturation and complete dryness, respectively.

The behavior of a pore is taken to be fully defined by $f(\bar{r}, \bar{\rho})$. In particular, it is independent of the states of the surrounding pores. This is called the "independent domain model."

The volumetric liquid content of the medium is obtained at any time by integrating the pore group distribution function over the portion of the unit square in $\bar{r} - \bar{\rho}$ space that corresponds to the wetted pores. The extent of this region defines the wetting history of the medium.

As an example, consider wetting the medium from complete dryness (i.e., wetting along the main wetting curve). The process of wetting is defined by an increase in the equilibrium radius of curvature of the air-water interface, R. When this radius increases from R to R + dR, the groups whose pore radii are between R and R + dR are wetted. This main wetting process is represented by the Mualem diagram of Figure 3.7, in which the shaded domain represents saturated pores.

Now consider the main drying curve. When the soil drains from R + dR to R, only the groups with pore radii $\bar{\rho}$ between R and R + dR and with opening radii $\bar{r}$ less than R are emptied (Figure 3.8).

Any subsequent reversals result in more complex saturation regions. Figure 3.9 shows how two primary scanning curves and one higher order curve appear on the diagram.

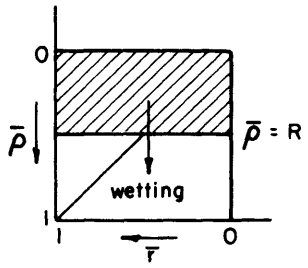


Figure 3.7

THE WETTED PORE DOMAIN IN THE MAIN WETTING PROCESS (After Mualem, 1974.)

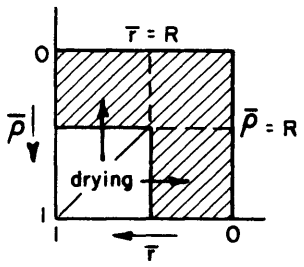


Figure 3.8

THE WETTED PORE DOMAIN IN THE MAIN DRYING PROCESS (After Mualem, 1974.)

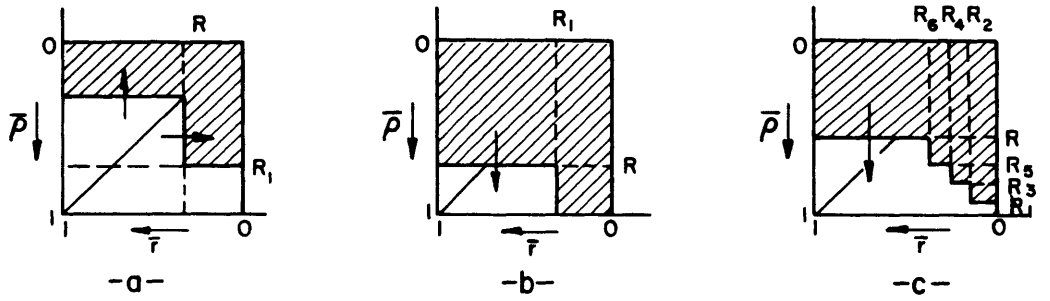


Figure 3.9

THE WETTED DOMAINS DURING SCANNING PROCESSES. (After Mualem, 1974.) (a.) PRIMARY DRYING AFTER REVERSAL AT R_1. (b.) PRIMARY WETTING AFTER REVERSAL AT R_1. (c.) SIXTH ORDER WETTING PROCESS. NOTE THAT

$$R_1 < R_4 < R_6 < R_5 < R_3 < R_1$$

Note in particular that for a high order scanning curve, alternate successive values of the reversal radius R are either monotonically increasing or decreasing, depending on the direction of the reversal. If, in the example in Figure 3.9c, wetting proceeds to a point where R exceeds R_5, then the last wet/dry loop is effectively erased from the memory of the medium. We see that the number of variables necessary to define the state of the system fluctuates. Theoretically, it is possible for the state vector to become arbitrarily large, although one might intuitively limit its size on the basis of the system dynamics.

Mualem (1974) used a simplified form of the pore group distribution function. It is

$$f(\bar{r}, \bar{\rho}) = h(\bar{r})\ \ell(\bar{\rho})$$

which constitutes the similarity hypothesis. In a subsequent paper (Mualem, 1977), he proposed an extended similarity hypothesis in the form

$$f(\bar{r}, \bar{\rho}) = h(\bar{r})\ h(\bar{\rho}) \tag{3.13}$$

Using the extended similarity hypothesis, Mualem (1977) showed that a universal hysteresis function could be derived. On the basis of one main curve, the other main curve and all scanning curves can be defined. The advantage of this model is that it greatly reduces the amount of information (data or parameters) necessary to define fully the water retention behavior of a soil.

The independent domain models described above fail when pore behavior is governed by surrounding pores. This occurs when there is significant hysteresis for $\psi > \psi_b$. The dependent domain model (Mualem and Dagan, 1975) was developed and tested as a generalization of the independent domain model in order to deal with this problem. Unfortunately, the extended similarity hypothesis, which reduces the amount of information needed to describe hysteresis, has not yet been validated for the dependent domain model. In this study, the independent domain model is used.

The use of a conceptual model based on the capillary model of moisture retention to predict the behavior of hysteresis in the "adsorption regime," where pF is, say, greater than 4, is open to question. Nevertheless, Mualem (1977) has found that this model is very good for pF up to 6, the highest value with which he worked. Whatever the physical explanation, it appears that the independent domain theory should be quite adequate for low moisture contents.

The cumulative integral of h(R) is defined as

$$H(R) = \int_0^R h(\bar{r})\, d\bar{r} \tag{3.14}$$

Since, by definition, $h(\bar{r})$ is non-negative, the function H(R) is monotonically increasing. When the medium is fully wetted,

$$\theta = \theta_u = \int_0^1 \int_0^1 f(\bar{r}, \bar{\rho})\, d\bar{r}d\bar{\rho}$$

$$= \int_0^1 \int_0^1 h(\bar{r})\, h(\bar{\rho})\, d\bar{r}d\bar{\rho}$$

$$= [H(1)]^2$$

whence

$$H(1) = \theta_u^{1/2} \tag{3.15}$$

In general, the main wetting process is described (see Figure 3.7) by

$$\theta_w(R) = \int_0^R h(\bar{\rho})\, d\bar{\rho} \int_0^1 h(\bar{r})\, d\bar{r}$$

$$= H(R)\, H(1) \tag{3.16}$$

whence

$$H(R) = \theta_u^{-1/2}\, \theta_w(R) \tag{3.17}$$

Similarly, the main drying curve (Figure 3.8 and curve C of Figure 3.3) is given by

$$\theta_d(R) = H(1)\, H(R) + H(R)[H(1) - H(R)] = [2\theta_u^{1/2} - H(R)]\, H(R) \tag{3.18}$$

Mualem (1977), assuming the ψ-R relation to be a one-to-one function, replaces R in the above equations by ψ. But constancy of R implies only constant moisture percentage, thus in a system in which the temperature varies, this simple substitution is improper. In this work,

we shall generalize Mualem's model to account for temperature effects. We assume (see the discussion of temperature effects in Section 3.3.2) that the temperature dependence of ψ has the form

$$-\frac{1}{\psi}\frac{\partial\psi}{\partial T}\bigg|_{\theta} = -\frac{1}{\psi}\frac{\partial\psi}{\partial T}\bigg|_{R} = a \equiv -C_{\psi} = \text{constant} \tag{3.19}$$

Integration of Equation (3.19) yields the relation

$$\Psi \equiv \psi\, e^{a(T-T_o)} = \text{constant for fixed R} \tag{3.20}$$

We may thus substitute Ψ for R in Equations (3.17) and (3.18) to obtain

$$H(\Psi) = \theta_u^{-1/2}\, \theta_w(\Psi) \tag{3.21}$$

and

$$\theta_d(\Psi) = [2\theta_u^{1/2} - H(\Psi)]\, H(\Psi) = [2 - \theta_u^{-1}\, \theta_w(\Psi)]\, \theta_w(\Psi) \tag{3.22}$$

A primary drying curve (Figure 3.9a and curve E of Fig. 3.3) is described by

$$\theta(_0\ ^{R_1}\ _R) = H(1)\, H(R) + H(R)[H(R_1) - H(R)] \tag{3.23}$$

where the notation indicates that the normalized capillary radius increased (wetting) from 0 to R_1, and decreased (drying) to R. Using Eqs. (3.16) and (3.17), and replacing R by Ψ, we obtain

$$\theta(_{\Psi_{min}}\ ^{\Psi_1}\ _{\Psi}) = \theta_w(\Psi)\, \{1 + \theta_u^{-1}[\theta_w(\Psi_1) - \theta_w(\Psi)]\} \tag{3.24}$$

In a similar manner, the primary wetting curve (Fig. 3.9b and curve D of Fig. 3.3) is derived. It is

$$\theta(\overset{\Psi_{max}}{\Psi_1}\overset{\Psi}{}) = H(1)\,H(R) + H(R_1)[H(1) - H(R)]$$
$$= \theta_w(\Psi) + \theta_w(\Psi_1)[1 - \theta_u^{-1}\,\theta_w(\Psi)] \qquad (3.25)$$

Routine application of the theory yields all of the higher order scanning curves (e.g., Fig. 3.9c and curve F of Fig. 3.3). As noted earlier, however, the introduction of higher order scanning curves poses serious operational problems as a result of the variable and potentially large state set. As a first approximation to the simulation of the second and higher order scanning curves, the Mualem model will be simplified in this study. As an expedient, it will be assumed that the higher order curves are coincident with the primary curves. Although this contradicts the theory, it allows quite accurate modeling of the most important effects of hysteresis without the computational difficulties inherent in describing the more detailed behavior. We treat high order scanning curves by assuming them to lie on a primary curve for the appropriate process (drying or wetting). For wetting, then,

$$\theta(\cdots\overset{\Psi}{\Psi_1}) \simeq \theta(\overset{\Psi_{max}}{\Psi_o}\overset{\Psi}{}) \qquad (3.26)$$

The value of Ψ_o is defined by the constraint that

$$\theta(\overset{\Psi_{max}}{\Psi_o}\overset{\Psi_1}{}) = \theta_1 \qquad (3.27)$$

where θ_1 is the value of θ (corresponding to Ψ_1) when the last reversal occurred. Using Equation (3.25) to solve for $\theta_w(\Psi_o)$ (Ψ_o itself is not

needed), we have

$$\theta_w(\Psi_o) = \theta_u[\theta_1 - \theta_w(\Psi_1)][\theta_u - \theta_w(\Psi_1)]^{-1} \tag{3.28}$$

The approximation for the higher order drying scanning curves is

$$\theta(\ldots\ {}^{\Psi_1}_{\ \ \Psi}) \simeq \theta({}^{\Psi_o}_{\Psi_{min}\ \ \Psi}) \tag{3.29}$$

where, using Eq. (3.24),

$$\theta_w(\Psi_o) = \theta_w(\Psi_1) + \theta_u[\theta_1\ \theta_w(\Psi_1)^{-1} - 1] \tag{3.30}$$

We now summarize the proposed model of the soil moisture retention process. Although Eq. (3.9) was introduced for isothermal conditions, we may generalize it, using the argument preceding Eq. (3.20), to obtain

Main Wetting Curve

$$\theta_w(\Psi) = \begin{cases} \frac{1}{M} \ln[p_1(-\Psi)^{q_1} + p_2(-\Psi)^{q_2}] & pF_{min} \leq PF \leq 7 \\ \theta_u & PF \leq pF_{min} \end{cases} \tag{3.31}$$

where

$$PF = \log(-\Psi) = pF + a(T - T_o)\ \log e \tag{3.32}$$

and the reference temperature, T_o, in Eq. (3.20) is the temperature at which the parameters of Eq. (3.9) are defined. We may take it to be the same T_o as that introduced in Eq. (2.18).

The derived curves are repeated below:

Main Drying Curve

$$\theta_d(\Psi) = [2 - \theta_u^{-1}\, \theta_w(\Psi)]\, \theta_w(\Psi) \tag{3.33}$$

Primary Drying Scanning Curve

$$\theta(\Psi_{min}\ ^{\Psi_1}\ \Psi) = \theta_w(\Psi)\ \{1 + \theta_u^{-1}[\theta_w(\Psi_1) - \theta_w(\Psi)]\} \tag{3.34}$$

Primary Wetting Scanning Curve

$$\theta(^{\Psi_{max}}\ _{\Psi_1}\ ^{\Psi}) = \theta_w(\Psi) + \theta_w(\Psi_1)[1 - \theta_u^{-1}\, \theta_w(\Psi)] \tag{3.35}$$

Higher Order Drying Scanning Curve

$$\theta(\ldots\ ^{\Psi_1}\ \Psi) = \theta(\Psi_{min}\ ^{\Psi_o}\ \Psi) \tag{3.36}$$

where

$$\theta_w(\Psi_o) = \theta_w(\Psi_1) + \theta_u[\theta_1\, \theta_w(\Psi_1)^{-1} - 1]$$

Higher Order Wetting Scanning Curve

$$\theta(\ldots\ _{\Psi_1}\ ^{\Psi}) = \theta(^{\Psi_{max}}\ _{\Psi_o}\ ^{\Psi}) \tag{3.37}$$

where

$$\theta_w(\Psi_o) = \theta_u[\theta_1 - \theta_w(\Psi_1)][\theta_u - \theta_w(\Psi_1)]^{-1}$$

We now clarify the meaning of R, the wetting history, which was introduced in Chapter 2. As mentioned earlier, the dimension of R

increases as the wetting history becomes more complex, assuming the general model of Mualem (1974). With the approximation that high order scanning curves lie on primary curves, the state set is truncated. Given the present value of ψ and one other number, it is possible to compute the present value of θ. This single state variable is the value Ψ_o at which the last reversal from a main curve occurred. For a higher order curve, Ψ_o is artificial and is calculated as discussed earlier. For a main curve, Ψ_o is either Ψ_{min} or Ψ_{max}. Thus, any curve can be represented as a primary scanning curve. It is a wetting curve if $\Psi > \Psi_o$ and drying if $\Psi < \Psi_o$.

The complete state set may now be considered to consist of ψ (or Ψ), T and Ψ_o. The vector $\underline{R}$, in this study, therefore has only one component - Ψ_o.

3.3 Hydraulic Conductivity

3.3.1 Introductory Remarks

Equation (2.3) states that when a head gradient is applied to a partially saturated porous medium, the resulting liquid flow rate is directly proportional to the magnitude of the applied gradient. In fact, many deviations from linearity have been reported. Swartzendruber (1969) reviews in detail the experimental observations, their suggested causes, and the implications for problem-solving, noting that some flow systems may be quite insensitive to the observed non-proportionalities. Furthermore, the Buckingham-Darcy equation (2.3) is the foundation of virtually all modern, physically-based studies of liquid transport in

the soil. It is not the purpose of this study to break new ground on the non-proportionality question.

The hydraulic conductivity, K, is a function of both θ and T (or ψ and T). The temperature effect is apparently attributable to the temperature dependence of viscosity, as will be discussed in Section 3.3.3. The isothermal hydraulic conductivity exhibits little or no hysteresis when expressed as a function of θ. Obviously, the hysteresis in the relation between θ and ψ results in significant hysteresis of K when it is expressed as a function of ψ.

For a fixed temperature, K has its maximum value at saturation and decreases monotonically over many orders of magnitude upon desaturation. Hillel (1971) gives typical saturation K's of 10^{-2} to 10^{-3} cm s^{-1} for sandy soils and 10^{-4} to 10^{-7} cm s^{-1} for clayey soils.

The conductivity effectively reaches zero when only a small amount of adsorbed liquid remains. At these very low moisture contents, vapor diffusion takes over as the dominant water transport mechanism. Rose (1963), offering data from several media, suggests that the value of moisture content at which liquid flow becomes negligible, which we have denoted by θ_k, corresponds to a relative humidity of h = 0.6, i.e.,

$$\theta_k = \theta(pF \cong 5.85)$$

The capillary model of water retention has been applied by several soil scientists and petroleum engineers to the calculation of hydraulic conductivity. The basic approach is to extract pore-size distribution information from the soil moisture characteristic, then to

apply viscous flow theory, and finally to aggregate the behavior of individual pores to describe the overall medium. Variations on this theme result in the models of Childs and Collis-George (1950), Burdine (1953), Marshall (1958), Millington and Quirk (1959), Kunze, et al. (1968), and Mualem (1976a). Mualem and Dagan (1978) present a unified derivation of several of these models.

These models of hydraulic conductivity, which are based on surface tension, viscous flow theory, are open to the same criticism as the capillary hysteresis conceptual model already presented. At low moisture contents, adsorption forces become important, so the capillary theory is incomplete. Furthermore, the physical properties of tightly-bound water may differ significantly from those of free water (Carman, 1953), resulting in such effects as non-linearity in the flux-gradient relation (Miller and Low, 1963). Hillel (1971) writes

> "The results of these [surface tension, viscous flow] theories, while more generally applicable than those based on earlier models, still appear to be valid only for certain coarse materials in which capillary hysteresis phenomena predominate."

Nevertheless, the experimental results of those investigators mentioned above, together with the evidence from other tests (Brooks and Corey, 1964, 1966; Jackson, et al., 1965; Green and Corey, 1971; Jackson, 1972) support the general use of such models in practical applications. As Jackson, et al. (1965) note,

> "The agreement between the experimental and theoretical values at these low water contents may be fortuitous, but the fact that they are of the same order of magnitude lends confidence to the calculation of conductivities at water contents where measurement is extremely difficult."

In the next section, a method for calculating the relative hydraulic conductivity at constant temperature is presented. The subsequent section deals with the temperature effect and summarizes the model of hydraulic conductivity used in this work.

3.3.2 Calculation of Relative Hydraulic Conductivity

Mualem (1976a) compared the measured conductivities for 45 soils with the values calculated by several methods. The best overall performance was yielded by a model proposed in that paper. It is

$$K_r(\theta) = S_e^{1/2}\left[\int_0^{S_e} \frac{dS}{\psi(S)}\right]^2\left[\int_0^{1} \frac{dS}{\psi(S)}\right]^{-2} \tag{3.38}$$

$$K_r(\theta) = \frac{K(\theta)}{K(\theta_u)} = \text{relative hydraulic conductivity}$$

$$S_e = \frac{\theta - \theta_k}{\theta_u - \theta_k} = \text{"effective" saturation} \tag{3.39}$$

$$S = \text{dummy integration variable for } S_e$$

It should be noted that the exponent of S_e, left undetermined in the derivation, has been chosen to give the best fit for the 45 soils. A similar advantage was not granted, for instance, to the Millington-Quirk method, which nevertheless performed rather well. Since the most systematic test and validation of any model appear to be that of Mualem (1976a), Eq. (3.38) will be used in this analysis.

As mentioned earlier, the hysteresis in the $K(\theta)$ relation is usually small (Talsma, 1970; Rogers and Klute, 1971; Nielsen and Biggar,

1961; Topp and Miller, 1966; Youngs, 1964; Elrick and Bowman, 1964). Mualem (1976b) examines the hysteresis of the hydraulic conductivity both theoretically and experimentally. In both cases, the results indicate only a small non-uniqueness of $K(\theta)$. Since hysteresis is negligible, it must be possible to integrate Eq. (3.38) using any curve of the hysteretic $\psi(\theta)$ family. For convenience, the main wetting curve, Eq. (3.31), is used.

In order to evaluate (3.38) for fixed temperature, it is necessary to invert (3.31) and to perform the indicated integration. For M going to infinity, these operations can be done analytically. Otherwise, a numerical scheme must be used.

3.3.3 Temperature Effects

The temperature dependencies of K and ψ, for fixed θ, are implied by the surface tension, viscous flow model (Miller and Miller, 1956; Childs and Collis-George, 1950) of soil water as

$$C_\psi \equiv \frac{1}{\psi}\frac{\partial \psi}{\partial T}\bigg|_\theta = \frac{1}{\sigma}\frac{d\sigma}{dT} - \frac{1}{\rho_\ell}\frac{d\rho_\ell}{dT} \tag{3.49}$$

and

$$C_K \equiv \frac{1}{K}\frac{\partial K}{\partial T}\bigg|_\theta = -\frac{1}{\mu}\frac{d\mu}{dT} + \frac{1}{\rho_\ell}\frac{d\rho_\ell}{dT} \tag{3.50}$$

where

C_ψ = the temperature coefficient of ψ [$^\circ K^{-1}$]

C_K = the temperature coefficient of K [$^\circ K^{-1}$]

μ = the dynamic viscosity [dyne s cm^{-2}]

For pure, free water at 20°C, the quantities in (3.49) and (3.50), estimated from data on $\sigma(T)$, $\mu(T)$, and $\rho_\ell(T)$ (Eagleson, 1970), are

$$\frac{1}{\sigma}\frac{d\sigma}{dT} \simeq -1.9 \times 10^{-3}\ {}^\circ K^{-1}$$

$$\frac{1}{\mu}\frac{d\mu}{dT} \simeq -2.5 \times 10^{-2}\ {}^\circ K^{-1}$$

$$\frac{1}{\rho_\ell}\frac{d\rho_\ell}{dT} \simeq -2.1 \times 10^{-4}\ {}^\circ K^{-1}$$

According to (3.49) and (3.50), these would imply

$$C_\psi \simeq -1.7 \times 10^{-3}\ {}^\circ K^{-1} \tag{3.51}$$

and

$$C_K \simeq 2.5 \times 10^{-2}\ {}^\circ K^{-1} \tag{3.52}$$

Experimental validation of (3.51) and (3.52) is far from complete. Equation (3.52) appears valid for saturated sands (Swartzendruber, 1969). For unsaturated media, it has been easier to study the temperature dependence of the isothermal liquid diffusivity defined by

$$D = K \left.\frac{\partial\psi}{\partial\theta}\right|_T$$

We would expect

$$C_D \equiv \frac{1}{D}\left.\frac{\partial D}{\partial T}\right|_\theta = C_K + C_\psi \simeq 2.3 \times 10^{-2}\ {}^\circ K^{-1} \tag{3.53}$$

assuming that (3.51) and (3.52) hold. Jackson (1963) presents data that support (3.53) for three moderately wet, heavy-textured soils. A

further investigation (Jackson, 1965), which carefully separates out the simultaneous vapor transport, shows that this dependence holds approximately for very dry media containing little more than an adsorbed monolayer of water (pF $\simeq$ 6.3). For even drier soil, the value of C_D increases three- or four-fold.

The experiments on D essentially verify the equation

$$C_\psi + C_K \simeq \frac{1}{\sigma}\frac{d\sigma}{dT} - \frac{1}{\mu}\frac{d\mu}{dT} \tag{3.54}$$

Assuming that there are not, purely by coincidence, significant equal and opposite errors in (3.49) and (3.50), we have an experimental validation of

$$C_K \simeq -\frac{1}{\mu}\frac{d\mu}{dT}$$

Since the surface tension effect is much smaller, significant (but less than a factor of ten) departures from (3.49) may occur.

Many investigators have attempted to determine the temperature dependence of ψ. The results of virtually every one refute (3.49), and yet the observed data suggest no consistent alternative to it. Jury and Miller (1974) give data suggesting a C_ψ on the order of -10^{-2} $°K^{-1}$ for a wet sand, as do Taylor and Stewart (1960) for a silt loam. The data of Taylor (1958) give even larger magnitudes. Campbell and Gardner (1971), reporting on four heavy-textured soils, give data corresponding to magnitudes of C_ψ ranging from 10^{-3} to 10^{-2} $°K^{-1}$, but having positive and negative signs, even for a single soil at different ψ and T. The most negative values of C_ψ for a given soil, in that study, are obtained at

the lower heads (pF > 4), while positive or less negative values result for pF < 4. Among different soils, correlation to soil type appears rather weak. Kijne and Taylor (1964) give data for a silt loam that suggest consistently positive C_ψ, on the order of 10^{-1} $°K^{-1}$, for pF from 3 to 5.

Faced with the inconsistencies of the $\psi(T)$ data, as well as their apparent contradiction of the diffusivity data, we shall tentatively accept the hypothesis expressed in (3.49) and (3.50). Further careful experimental and theoretical analyses of these temperature effects are needed. Inasmuch as C_ψ is uncertain, the sensitivity of transport processes in the soil to its value should be examined.

We may now express the hydraulic conductivity as a function of and T. It is

$$K(\theta, T) = K_s \, K_r(\theta) \, K_T(T)$$

in which

$K_s = K(\theta = \theta_u, T = T_o)$ = saturated reference conductivity [cm s^{-1}]

$K_T(T) = \left[\frac{\mu(T_o)}{\mu(T)}\right]$ = temperature correction of hydraulic conductivity

where the density effect is considered negligible compared to the viscosity effect.

3.4 Tortuosity

Due to the presence of the soil matrix, the effective diffusion coefficient of a gas in a porous medium is lower than that in free air. The reduction is a result both of the reduced cross-sectional area for flow and of the tortuosity of the diffusion path. Lai, et al. (1976) discussed previous studies of gas diffusion in porous media and performed field measurements of the process. Their data support the use of a reduction factor, $\alpha\ \theta_a$ in Eq. (2.12), given by

$$\alpha\ \theta_a = \theta_a^{5/3}$$

Thus,

$$\alpha = \theta_a^{2/3} \tag{3.55}$$

3.5 Thermal Properties

3.5.1 Heat Capacity

The (bulk) heat capacity of a unit volume of soil is the weighted average of the capacities of the constituents. Thus, in the definition of C following Eq. (2.24), we have

$$C = C_d + c_p \rho_v \theta_a + c_\ell \rho_\ell \theta \tag{3.56}$$

According to de Vries (1966), the dry heat capacity is estimated well for most soils from the relation

$$C_d = 0.46\ \theta_m + 0.60\ \theta_o \quad (\text{cal } g^{-1}\ C^{-1}) \tag{3.57}$$

in which

θ_m is the volume fraction of soil minerals

θ_o is the volume fraction of organic matter

Note that

$$1 - n = \theta_m + \theta_o$$

so we may eliminate θ_m from (3.57) to obtain

$$C_d = 0.46(1-n) + 0.14\ \theta_o \qquad (3.58)$$

3.5.2 Thermal Conductivity

It can be shown that the two conductivities, λ^{ψ} and λ^{θ}, defined in Chapter 2, are virtually the same; the minor distinction need not be discussed here. We shall henceforth use the symbol λ without a superscript, with the understanding that it may be used in either the (ψ, T) or the (θ, T) system.

A physical theory for the calculation of the effective thermal conductivity of a partially saturated porous medium has been developed by de Vries (1966). The theory has its roots in the work of Maxwell and Lord Rayleigh, who calculated the effective electrical conductivity of a non-homogeneous medium. The details of this model will not be presented here. It has been applied by several workers with varying degrees of success (Jury and Miller, 1974; Wierenga, et al., 1969; Hadas, 1977; Sepaskhah and Boersma, 1979;

Kimball, et al., 1976). The seemingly ad hoc introduction of various corrections to the theory makes it undesirable for this study, in which a simple, but general, representation of λ is sought. It should be remembered, however, that some elements of this theory appear in this paper, e.g., the factor ζ in Eq. (2.10).

Both experimental and theoretical work have established the general nature of the dependence of λ on θ and T. For constant temperature, the thermal conductivity increases rapidly as the medium is initially wetted. The formation of water rings between adjacent soil particles accounts for this rapid increase in thermal conductivity. At some moisture content around the field capacity (pF ~ 2.5), the rate of increase becomes significantly lower, assuming the temperature has a value normally encountered in natural environments. Conductivity is a maximum at saturation.

The temperature dependence of λ enters through the effect of vapor diffusion (and transport of latent heat). This effect becomes significant above 40°C, where the saturated vapor pressure becomes sufficiently large to affect the overall heat transport process. We shall neglect it here.

As an approximate representation of the thermal behavior of most soils, we shall adopt the following piecewise linear relation:

$$\lambda(\theta) = \begin{cases} \lambda_o + (\lambda_1 - \lambda_o)\left(\frac{\theta}{\theta_f}\right) & 0 \le \theta \le \theta_f \\ \lambda_1 + (\lambda_2 - \lambda_1)\left(\frac{\theta-\theta_f}{n-\theta_f}\right) & \theta_f < \theta \le n \end{cases} \tag{3.59}$$

in which

θ_f = θ at field capacity = θ (PF = 2.5)

λ_o = dry thermal confuctivity of medium (cal cm^{-1} s^{-1} $°C^{-1}$)

λ_2 = saturated thermal conductivity (cal cm^{-1} s^{-1} $°C^{-1}$)

λ_1 = an intermediate value of λ.

Typical values of λ_o are in the range of 3×10^{-4} to 7×10^{-4} cal cm^{-1} $°C^{-1}$, while λ_2 is about an order of magnitude larger. The value λ_1 is about midway between the other two.

3.5.3 The Thermal Gradient Ratio, ζ

The value of ζ is a by-product of the theory of de Vries (1966) for the calculation of λ. Philip and de Vries (1957) give values of ζ at 20°C for various values of n and θ, and for quartz and non-quartz soil minerals. (Quartz is the only common mineral with a thermal conductivity different from the norm.) The values range from 1.3 (n = 0.7, θ = 0.1) to 3.2 (n = 0.3, θ = 0). It appears that a value of ζ = 2 will suffice as an approximate value for most situations.

3.5.4 The Heat of Wetting

The heat of wetting is given (Edlefson and Anderson, 1943, p. 237; de Vries, 1958) by

$$W = -j^{-1} g(\psi - T \frac{\partial \psi}{\partial T}) \qquad (3.60)$$

in which

j = mechanical equivalent of heat = 4.18×10^7 erg/cal.

Chapter 4

NUMERICAL SOLUTION OF THE MASS AND HEAT CONSERVATION EQUATIONS

4.1 Introduction

This chapter outlines a numerical scheme for the solution, in one dimension, of the conservation equations derived in Chapter 2, subject to given initial and boundary conditions. The application of Galerkin's method of weighted residuals converts the partial differential equations to a system of non-linear ordinary differential equations whose unknowns are the values of the state variables at a finite number of points. Boundary conditions are easily applied at this stage. A finite difference approximation is introduced to evaluate the time derivatives, resulting in a completely algebraic system of equations. This final system, which is still non-linear, is solved by an iterative series of successive linearizations.

4.2 Application of Galerkin's Method of Weighted Residuals

4.2.1 Preliminaries

In Chapter 2, several expressions of mass and heat conservation were presented. Those chosen for use in this study are based on the dependent (state) variables ψ and T. The mass conservation equation is (2.32) and the heat equation is (2.33). The versatility of the (ψ, T) system has already been discussed. The particular choice of Equation (2.33), rather than (2.34) or (2.35), is based both on the symmetric nature of the matrices resulting from the numerical methods, and on its direct physical significance.

Introducing new notation for the coefficients in the continuity equations, we have, from (2.32),

$$M(\psi,T) = c_1 \frac{\partial \psi}{\partial t} + c_2 \frac{\partial T}{\partial t} - \frac{\partial}{\partial z} \left(c_3 \frac{\partial \psi}{\partial z} + c_4 \frac{\partial T}{\partial z} + c_5\right) + c_6 = 0 \qquad (4.1)$$

and, from (2.33),

$$H(\psi,T) = d_1 \frac{\partial \psi}{\partial t} + d_2 \frac{\partial T}{\partial t} - \frac{\partial}{\partial z} \left(d_3 \frac{\partial \psi}{\partial z} + d_4 \frac{\partial T}{\partial z} + d_5\right) = 0 \qquad (4.2)$$

where the one-dimensional equations have been adopted. $M(\cdot,\cdot)$ and $H(\cdot,\cdot)$ are differential operators. The coefficients in (4.1) and (4.2) are defined below:

$$c_1 = \left(1 - \frac{\rho_v}{\rho_\ell}\right) \frac{\partial \theta}{\partial \psi}\bigg|_T + \frac{\theta_a}{\rho_\ell} \frac{\partial \rho_v}{\partial \psi}\bigg|_{\dot{T}} \qquad (4.3a)$$

$$c_2 = \left(1 - \frac{\rho_v}{\rho_\ell}\right) \frac{\partial \theta}{\partial T}\bigg|_\psi + \frac{\theta_a}{\rho_\ell} \frac{\partial \rho_v}{\partial T}\bigg|_\psi \qquad (4.3b)$$

$$c_3 = K + D_{\psi v} \qquad (4.3c)$$

$$c_4 = D_{Tv}^{\psi} \qquad (4.3d)$$

$$c_5 = K \qquad (4.3e)$$

$$c_6 = S \qquad (4.3f)$$

$$d_1 = [L_o + c_p (T - T_o)]\, \theta_a \left.\frac{\partial \rho_v}{\partial \psi}\right|_T + [c_\ell \rho_\ell (T - T_o) - \rho_\ell W - c_p \rho_v (T - T_o) - L_o \rho_v] \left.\frac{\partial \theta}{\partial \psi}\right|_T \tag{4.4a}$$

$$d_2 = C + [L_o + c_p (T - T_o)]\, \theta_a \left.\frac{\partial \rho_v}{\partial T}\right|_\psi + [c_\ell \rho_\ell (T - T_o) - \rho_\ell W - c_p \rho_v (T - T_o) - L_o \rho_v] \left.\frac{\partial \theta}{\partial T}\right|_\psi \tag{4.4b}$$

$$d_3 = \rho_\ell L D_{\psi v} + c_\ell \rho_\ell (T - T_o) (K + D_{\psi v}) \tag{4.4c}$$

$$d_4 = \lambda^\psi + c_\ell \rho_\ell (T - T_o) D_{Tv}^\psi \tag{4.4d}$$

$$d_5 = c_\ell \rho_\ell (T - T_o) \left[K - \int_{-L}^{z} S(z')\, dz'\right] \tag{4.4e}$$

The coefficients d_3, d_4, and d_5 are obtained from (2.33) by use of the one-dimensional versions of (2.14) and (2.25). The depth L is at least as great as the depth of the root zone.

Before applying the finite element method, we shall define the solution domain and its discretization. We are concerned with the flow of moisture and heat inside the region bounded by the land surface ($z = 0$) and some arbitrary depth, $z = -L$ (Figure 4.1).

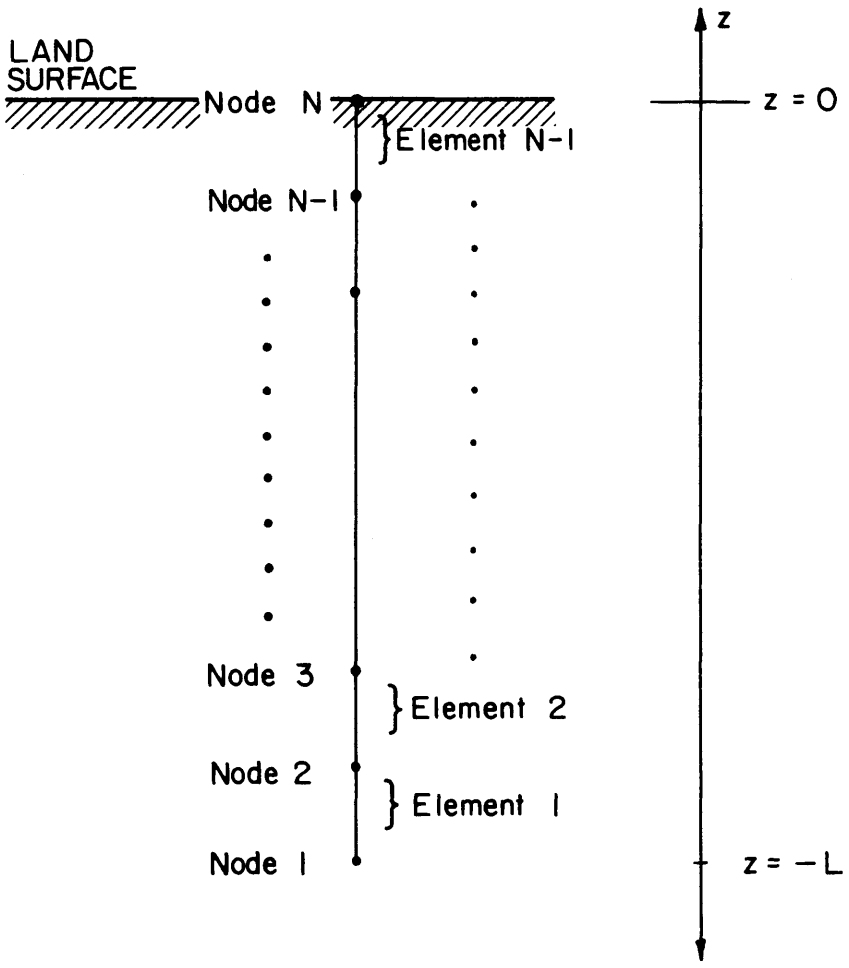


Figure 4.1

FINITE ELEMENT GRID FOR ANALYSIS OF ONE-DIMENSIONAL MASS AND HEAT TRANSPORT

We divide this domain into N-1 elements (sub-domains), bounded by N nodes. An arbitrary numbering system to be adopted here assigns the index 1 to the lowest node, 2 to the second, and so on up to the top node, N. The element numbering system is the same, except that the top element is, of course, number N-1.

We shall seek approximate solutions of (4.1) and (4.2) having the form (Neuman, et al., 1975)

$$\hat{\psi}(z,t) = \sum_{j=1}^{N} \psi_j(t) \underset{e}{U} \omega_j^e (z) \tag{4.5}$$

and

$$\hat{T}(z,t) = \sum_{j=1}^{N} T_j (t) \underset{e}{U} \omega_j^e (z) \tag{4.6}$$

where $\hat{\psi}$ and $\hat{T}$ are the approximate solutions. The moisture potential and temperature at the j'th node are given by ψ_j and T_j respectively. The trial function, $\omega_j^e (z)$, is non-zero only on element e. Its value varies linearly from unity at node j to zero at the other end of the element (Figure 4.2).

4.2.2 Application to a Single Element

Ordinarily, one applies Galerkin's method directly over the entire region of interest. In the present development, we will apply Galerkin's method instead to each element separately, and then combine the local, or element, equations by matching boundary conditions among the elements, obtaining a global system of equations.

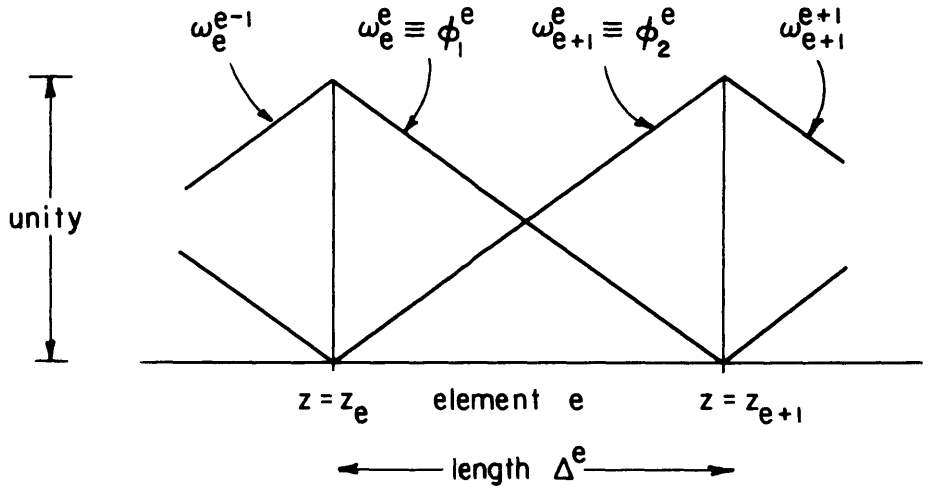


Figure 4.2

THE TRIAL FUNCTIONS EMPLOYED IN THE GALERKIN

FINITE ELEMENT FORMULATION

This is done for two reasons. The first reason is the inapplicability of the governing equations at various discontinuities. This issue is discussed in Section 2.3.4. Secondly, this approach clarifies the evaluation of the global coefficient matrices, a task that is often a notational nightmare. It can be shown that the equations resulting from either approach are identical.

Let us define some notation to facilitate manipulations on the element level. We denote the trial functions locally by

$$\phi_1^e \equiv \omega_e^e \tag{4.7}$$

$$\phi_2^e \equiv \omega_{e+1}^e \tag{4.8}$$

We therefore have

$$\begin{aligned} \phi_1^e &= \frac{z_{e+1} - z}{z_{e+1} - z_e} = \frac{z_{e+1} - z}{\Delta^e} \\ \phi_2^e &= \frac{z - z_e}{z_{e+1} - z_e} = \frac{z - z_e}{\Delta^e} \end{aligned} \tag{4.9}$$

in which Δ^e is the length of element e.

Inside a single element, (4.5) and (4.6) simplify, as a result of the definition of ω_j^e, to

$$\hat{\psi} = \bar{\psi}_1^e \phi_1^e + \bar{\psi}_2^e \phi_2^e \tag{4.10}$$

and

$$\hat{T} = \bar{T}_1^e \phi_1^e + \bar{T}_2^e \phi_2^e \tag{4.11}$$

in which $\bar{\psi}_1^e \equiv \psi_e \qquad \bar{\psi}_2^e \equiv \psi_{e+1}$

and $\bar{T}_1^e \equiv T_e \qquad \bar{T}_2^e \equiv T_{e+1}$

In the following treatment of the local equations, the superscript e may now be dropped, as it appears on all variables. The subscripts are understood to denote the local nodal numbering system of the element.

Following Galerkin's method of weighted residuals (Pinder and Gray, 1977, p. 57) for each element, we require that the residuals obtained by substituting $\hat{\psi}$ and $\hat{T}$ into M and H be orthogonal to the set of trial functions.

Mathematically,

$$\int_\Delta M(\hat{\psi}, \hat{T})\, \phi_i \, dz = 0 \qquad i = 1,2 \tag{4.12}$$

$$\int_\Delta H(\hat{\psi}, \hat{T})\, \phi_i \, dz = 0 \qquad i = 1,2 \tag{4.13}$$

The subsequent steps necessary to convert (4.12) and (4.13) into computationally practical forms will be illustrated here only for the mass equation. The heat equation is, by analogy, similar.

Let us substitute (4.1) into (4.12), using (4.10) and (4.11). This yields

$$\int_{\Delta} \left[c_1 \frac{\partial \hat{\psi}}{\partial t} + c_2 \frac{\partial \hat{T}}{\partial t} - \frac{\partial}{\partial z}\left(c_3 \frac{\partial \hat{\psi}}{\partial z} + c_4 \frac{\partial \hat{T}}{\partial z} + c_5\right) + c_6\right] \phi_i \, dz = 0 \qquad i = 1, 2 \qquad (4.14)$$

We next apply integration by parts to the third term, which may be recognized as the flux divergence.

$$\int_{\Delta} \left(c_1 \frac{\partial \hat{\psi}}{\partial t} + c_2 \frac{\partial \hat{T}}{\partial t}\right) \phi_i \, dz + \int_{\Delta} \left(c_3 \frac{\partial \hat{\psi}}{\partial z} + c_4 \frac{\partial \hat{T}}{\partial z} + c_5\right) \phi_i' \, dz + \int_{\Delta} c_6 \phi_i \, dz = \left[\left(c_3 \frac{\partial \hat{\psi}}{\partial z} + c_4 \frac{\partial \hat{T}}{\partial z} + c_5\right) \phi_i\right]_{z_1}^{z_2} = \left[-Q_m \phi_i\right]_{z_1}^{z_2} \quad i = 1, 2 \qquad (4.15)$$

in which the z's are the element end points, subscripted according to the local numbering system. Checking the definitions of c_3, c_4 and c_5, we see that (4.15) defines Q_m implicitly as

$$Q_m = (\underline{q}_\ell + \underline{q}_v) \cdot \underline{k}/\rho_\ell \qquad (4.16)$$

The primed trial function denotes the z-derivative.

We substitute $\hat{\psi}$ and $\hat{T}$ into (4.15). This yields the following pair of equations:

$$\sum_{j=1}^{2} \overline{\psi}_j' \int c_1 \phi_j \phi_i \, dz + \sum_{j=1}^{2} \overline{T}_j' \int c_2 \phi_j \phi_i \, dz + \sum_{j=1}^{2} \overline{\psi}_j \int c_3 \phi_j' \phi_i' \, dz + \sum_{j=1}^{2} \overline{T}_j \int c_4 \phi_j' \phi_i' \, dz + \int c_5 \phi_i' \, dz + \int c_6 \phi_i \, dz = \left[-Q_m \phi_i\right]_{z_1}^{z_2} \quad i = 1, 2 \qquad (4.17)$$

In Equation (4.17), the primed state variables are time derivatives. Equations (4.17), with i = 1, 2, can be interpreted as approximate mass conservation relations for nodes 1 and 2. Their sum is a mass conservation condition for the element.

4.2.3 Evaluation of Element Matrix Integrals

The integrals in (4.17) remain to be evaluated. If the 'c' coefficients were constant in an element, this would be a trivial operation, since the ϕ's are simple linear functions. In general, however, the c's are functions of the state variables. A standard integration procedure, which will be used here, is the functional coefficient scheme (Pinder and Gray, 1977, p. 132). The coefficient is assumed to vary linearly within the element. Its value at the nodes is given by

$$\left.\begin{aligned} c_{m1} &= c_m(\overline{\psi}_1, \overline{T}_1) \\ c_{m2} &= c_m(\overline{\psi}_2, \overline{T}_2) \end{aligned}\right\} \quad m = 1, 2, \ldots, 6 \qquad (4.18)$$

If c is discontinuous at a node, it is evaluated as the limit of c that is approached from inside the element.

Given this form for the variation of the c's, we may proceed to evaluate the integrals in (4.17), using the definitions (4.9). Taking the first one as an example, we have

$$\int c_1 \phi_j \phi_i \, dz = c_{11} \int \phi_1 \phi_j \phi_i \, dz + c_{12} \int \phi_2 \phi_j \phi_i \, dz \qquad (4.19)$$

where, in general,

$$\int_{\Delta} \phi_k \, \phi_j \, \phi_i \, dz = \begin{cases} \frac{\Delta}{4} & i = j = k \\ \\ \frac{\Delta}{12} & \text{otherwise} \end{cases} \tag{4.20}$$

The result is similar for the integrals containing c_2.

Now consider the second type of integral appearing in (4.17). It is

$$\int c_3 \, \phi_j' \, \phi_i' \, dz = \sum_{k=1}^{2} c_{3k} \int \phi_k \, \phi_j' \, \phi_i' \, dz \tag{4.21}$$

in which

$$\int_{\Delta} \phi_k \, \phi_j' \, \phi_i' \, dz = \phi_j' \, \phi_i' \, \frac{\Delta}{2} = \begin{cases} \frac{1}{2\Delta} & i = j \\ \\ -\frac{1}{2\Delta} & i \neq j \end{cases} \tag{4.22}$$

The integrals containing c_4 are similar.

The gravity flow term is given by

$$\int c_5 \, \phi_i' \, dz = \sum_{k=1}^{2} c_{5k} \int \phi_k \, \phi_i' \, dz \tag{4.23}$$

with

$$\int_{\Delta} \phi_k \, \phi_i' \, dz = \frac{\Delta}{2} \, \phi_i' = \begin{cases} -\frac{1}{2} & i = 1 \\ \\ \frac{1}{2} & i = 2 \end{cases} \tag{4.24}$$

The sink term is

$$\int c_6 \, \phi_i \, dz = \sum_{k=1}^{2} c_{6k} \int \phi_k \, \phi_i \, dz \tag{4.25}$$

where

$$\int_{\Delta} \phi_k \, \phi_i \, dz = \begin{cases} \frac{\Delta}{3} & i = k \\ \\ \frac{\Delta}{6} & i \neq k \end{cases} \tag{4.26}$$

Finally, we may evaluate the right-hand-side term. It is

$$\left[-Q_m \, \phi_i\right]_{z_1}^{z_2} = \begin{cases} Q_m\Big|_{z_1} & i = 1 \\ \\ -Q_m\Big|_{z_2} & i = 2 \end{cases} \tag{4.27}$$

Having evaluated the coefficients in (4.17), we now rewrite that system of equations in matrix form:

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} \bar{\psi}_1' \\ \bar{\psi}_2' \end{bmatrix} + \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} \begin{bmatrix} \bar{T}_1' \\ \bar{T}_2' \end{bmatrix} + \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \begin{bmatrix} \bar{\psi}_1 \\ \bar{\psi}_2 \end{bmatrix}$$

$$+ \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} \bar{T}_1 \\ \bar{T}_2 \end{bmatrix} + \begin{bmatrix} E_1 \\ E_2 \end{bmatrix} + \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} = \begin{bmatrix} Q_m\Big|_{z_1} \\ -Q_m\Big|_{z_2} \end{bmatrix} \tag{4.28}$$

These equations, combined with the corresponding heat equations, constitute a set of four ordinary differential equations for the four state variables - $\bar{\psi}_1$, $\bar{\psi}_2$, $\bar{T}_1$, and $\bar{T}_2$.

4.2.4 Application of Boundary Conditions

In a problem where flux boundary conditions are prescribed, the relevant Q's are substituted into the right-hand-side vector, and the four unknown states are found by integrating the system. On the

other hand, if one or more of the states is fixed at a node (first-type boundary condition), the equation(s) containing the (unknown) flux through that node is (are) eliminated from the system of equations, keeping the solution uniquely determined. After solution, the neglected equations may be used to determine boundary fluxes.

4.2.5 Linking Elements Together

We now describe how the subdomains are linked together to obtain a global system of equations. Recall that the boundaries separating individual elements are points at which discontinuities may occur. Coupling will be accomplished through the matching of boundary conditions at the discontinuities (Section 2.3.4). We will illustrate this linkage for a pair of elements. The extension to several elements should be obvious by induction.

First of all, we note that Equation (4.28) will hold for each element. Since we are now dealing with more than one element, the element superscript will be reintroduced. It will take on values of 1 and 2. Recall that both node (local or global) and element indices increase in the positive z direction.

The coupling conditions (2.73a) and (2.73b) give us

$$\bar{\psi}_2^1 = \bar{\psi}_1^2$$

$$\bar{T}_2^1 = \bar{T}_1^2$$

while condition (2.73d), given continuity of $\underline{q}_r$, yields

$$Q_m\Big|_{z_2^1} = Q_m\Big|_{z_1^2}$$

Reintroducing the global index for the coordinate z and for the state variables, and adding together the second row of (4.28) for element 1 and the first row for element 2, we obtain

$$\begin{bmatrix} A_{11}^1 & A_{12}^1 & 0 \\ A_{21}^1 & A_{22}^1 + A_{11}^2 & A_{12}^2 \\ 0 & A_{21}^2 & A_{22}^2 \end{bmatrix} \begin{bmatrix} \psi_1' \\ \psi_2' \\ \psi_3' \end{bmatrix} + \begin{bmatrix} B_{11}^1 & B_{12}^1 & 0 \\ B_{21}^1 & B_{22}^1 + B_{11}^2 & B_{12}^2 \\ 0 & B_{21}^2 & B_{22}^2 \end{bmatrix} \begin{bmatrix} T_1' \\ T_2' \\ T_3' \end{bmatrix}$$

$$+ \begin{bmatrix} C_{11}^1 & C_{12}^1 & 0 \\ C_{21}^1 & C_{22}^1 + C_{11}^2 & C_{12}^2 \\ 0 & C_{21}^2 & C_{22}^2 \end{bmatrix} \begin{bmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \end{bmatrix} + \begin{bmatrix} D_{11}^1 & D_{12}^1 & 0 \\ D_{21}^1 & D_{22}^1 + D_{11}^2 & D_{12}^2 \\ 0 & D_{21}^2 & D_{22}^2 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix}$$

$$+ \begin{bmatrix} E_1^1 \\ E_2^1 + E_1^2 \\ E_2^2 \end{bmatrix} + \begin{bmatrix} F_1^1 \\ F_2^1 + F_1^2 \\ F_2^2 \end{bmatrix} = \begin{bmatrix} Q_m\Big|_{z_1} \\ 0 \\ -Q_m\Big|_{z_3} \end{bmatrix} \qquad (4.29)$$

Once again, we see that the total number of unknowns (6) is equal to the number of equations (3 each from the mass and heat conservation conditions). The boundary conditions may be applied as described in the previous section.

The extension of this procedure to cover N nodes (N-1 elements) is straightforward. The final pair of matrix equations is

$$\underline{\underline{A}}_1 \underline{\psi}' + \underline{\underline{B}}_1 \underline{T}' + \underline{\underline{C}}_1 \underline{\psi} + \underline{\underline{D}}_1 \underline{T} + \underline{E}_1 + \underline{F}_1 = \underline{Q}_1 \qquad (4.30)$$

and

$$\underline{\underline{A}}_2 \underline{\psi}' + \underline{\underline{B}}_2 \underline{T}' + \underline{\underline{C}}_2 \underline{\psi} + \underline{\underline{D}}_2 \underline{T} + \underline{E}_2 = \underline{Q}_2 \qquad (4.31)$$

The elements of the heat equation, (4.31), are implied by analogy with those defined for the mass equation.

Note that the matrices appearing in (4.30) and (4.31) are all symmetric and tridiagonal. This will be an important factor in choosing a solution strategy. Recall also that there are strong non-linearities embodied in the coefficient matrices and in the vectors. The problem of integrating (4.30) and (4.31) is thus non-trivial.

In the following developments, we shall not deal explicitly with the boundary conditions. They are easily incorporated as described earlier.

4.3 Approximation of the Time Derivatives

The time derivatives in (4.30) and (4.31) are evaluated by finite difference. In this work, a fully implicit, backward difference scheme is used. This means that all terms other than the time derivative are evaluated at the end of the time step:

$$\underline{\underline{A}}_1^k \frac{\underline{\psi}^k - \underline{\psi}^{k-1}}{\Delta t} + \underline{\underline{B}}_1^k \frac{\underline{T}^k - \underline{T}^{k-1}}{\Delta t} + \underline{\underline{C}}_1^k \underline{\psi}^k + \underline{\underline{D}}_1^k \underline{T}^k + \underline{E}_1^k + \underline{F}_1^k = \underline{Q}_1^k \qquad (4.32)$$

in which Δt is the time increment. An implicit integration scheme is usually much more stable than an explicit one. The heat equation is treated in the same fashion.

Let us rewrite (4.32) considering $\underline{\psi}^k$ to be the unknown. This yields

$$(\frac{1}{\Delta t}\,\underline{\underline{A}}_1^k + \underline{\underline{C}}_1^k)\,\underline{\psi}^k = -\,(\frac{1}{\Delta t}\,\underline{\underline{B}}_1^k + \underline{\underline{D}}_1^k)\,\underline{T}^k + \frac{1}{\Delta t}\,\underline{\underline{A}}_1^k\,\underline{\psi}^{k-1} + \frac{1}{\Delta t}\,\underline{\underline{B}}_1^k\,\underline{T}^{k-1}$$

$$- \underline{E}_1^k - \underline{F}_1^k + \underline{Q}_1^k \qquad (4.33)$$

Similarly, for the heat equation, we have

$$(\frac{1}{\Delta t}\,\underline{\underline{B}}_2^k + \underline{\underline{D}}_2^k)\,\underline{T}^k = -\,(\frac{1}{\Delta t}\,\underline{\underline{A}}_2^k + \underline{\underline{C}}_2^k)\,\underline{\psi}^k + \frac{1}{\Delta t}\,\underline{\underline{A}}_2\,\underline{\psi}^{k-1}$$

$$+ \frac{1}{\Delta t}\,\underline{\underline{B}}_2^k\,\underline{T}^{k-1} - \underline{E}_2^k + \underline{Q}_2^k \qquad (4.34)$$

4.4 Iterative Solution Strategy

The following iterative strategy is proposed for the solution of (4.33) and (4.34):

1. Extrapolate the solutions for the last two time steps (k-2 and k-1) forward to obtain an estimate of $\underline{\psi}^k$ and $\underline{T}^k$. (If this is the first time step, only the k-1 "solution" is available - it is the initial condition. In this case, we can assume $\underline{\psi}^k$ and $\underline{T}^k$ are given by the initial conditions, for a first guess.)

2. Use the estimated states to evaluate all components (except $\underline{\psi}^k$) in (4.33). Solve the resulting tridiagonal matrix equation for $\underline{\psi}_k$.

3. Use the latest estimates of $\underline{\psi}^k$ and $\underline{T}^k$ to evaluate (4.34) and solve for $\underline{T}^k$.

4. Repeat steps 2 and 3 until some convergence criterion is met.

With this algorithm, we never need to solve a matrix equation any more complex than one in which the matrix is tridiagonal. A very

fast procedure exists for the solution of such equations (Pinder and Gray, p. 23, 1977).

Note that it is possible to incorporate any of the boundary conditions, including the non-linear ones, described in Section 2.3.1, into this iterative procedure, using the methods of Section 4.2.4.

4.5 Treatment of Hysteresis and Soil Discontinuities

The use of Equations (4.1) and (4.2) has been restricted (Section 2.2.3) to domains on which $\frac{\partial \theta}{\partial \psi}$ and other storage parameters are continuous in time, and on which the various conductivities and diffusivities are continuous in space. In general, this will require that no discontinuities in the relevant soil properties occur inside a particular domain, and that no temporal reversals in wetting history occur during the time period of interest. These restrictions were illustrated in Figure 2.1. To them we add the convenient modeling assumption that the wetting history is spatially continuous within the domain. Although this is theoretically inconsistent with the approximations used for ψ and T, it appears justifiable if the length of the domain is small compared to the scale of the phenomenon under study. Alternatively, one could allow wetting discontinuities inside elements, but this would complicate the evaluation of the integrals in the element matrices as the interior discontinuities would invalidate the direct use of the functional coefficient scheme described in Section 4.2.3.

The "domains" on which the conservation equations are to be applied may now be identified. Spatially, they are simply the individual elements of the finite element discretization. As demonstrated in general

terms in Section 2.3.4 and for the finite element model in Section 4.2.5, these domains are coupled through matching of boundary conditions. Temporal discontinuities are allowed to occur between time steps.

The hysteresis model described in Section 3.2.4 is applied at each node of each element in order to determine the values of θ and $\frac{d\theta}{d\Psi}$ at all time. Since we have chosen to keep the wetting history spatially continuous, we will require that a wetting reversal occur instantaneously (between time steps) throughout an entire element. The reversal moisture content may vary continuously in the element. A wetting reversal will be assumed to occur at time t^k in element e if the (space) average moisture content in the element at time t^k is greater (less) than the average content at time t^{k-1}, where the element was previously drying (wetting). The reversal actually computed, then, will precede the adoption of a new scanning curve by one time step.

An alternative to this procedure would be to choose the proper scanning curve each iteration on the basis of the most recently calculated moisture contents for the present time step. The lag would then be avoided. Preliminary numerical experiments with this more rigorous procedure yielded approximately the same results. Apparently, the rather samll time step required by the system non-linearities also minimizes the error in the time-lagged hysteresis procedure.

4.6 Lumping the Storage Matrices

Several investigators (e.g., Neuman, et al., 1975; Mercer and Faust, 1976) have found that the form of the storage matrices (i.e., $\underline{\underline{A}}$

and $\underline{\underline{B}}$ in Eqs. (4.30) and (4.31)) generated by the Galerkin method often lead to numerical difficulties. A commonly accepted means of overcoming these problems is to diagonalize, or lump, the storage matrix. Preliminary results of this study confirmed the superiority of a diagonalized storage matrix, which has therefore been adopted in this work.

Consider the first matrix of Eq. (4.28). Using (4.17), (4.19) and (4.20), we obtain

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \Delta \begin{bmatrix} \frac{c_{11}}{4} + \frac{c_{12}}{12} & \frac{c_{11}}{12} + \frac{c_{12}}{12} \\ \frac{c_{11}}{12} + \frac{c_{12}}{12} & \frac{c_{11}}{12} + \frac{c_{12}}{4} \end{bmatrix}$$

In the present work, the following modification is introduced:

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \Delta \begin{bmatrix} \frac{c_{11}}{2} & 0 \\ 0 & \frac{c_{12}}{2} \end{bmatrix} \tag{4.35}$$

A similar modification is used in evaluating the other storage matrices.

4.7 Mass and Energy Balances; Evaluation of $\frac{d\theta}{d\Psi}$

4.7.1 General Balance Considerations

Consider now the finite element equations resulting from the application of Galerkin's method to a single element. In matrix form, these equations (for mass or heat) are given by (4.28). Using (4.17) and the results of Section 4.2.3, together with the lumped storage assumption, we obtain

$$\begin{bmatrix} \frac{\Delta c_{11}}{2} & 0 \\ 0 & \frac{\Delta c_{12}}{2} \end{bmatrix} \begin{bmatrix} \frac{\bar{\psi}_1^k - \bar{\psi}_1^{k-1}}{\Delta t} \\ \frac{\bar{\psi}_2^k - \bar{\psi}_2^{k-1}}{\Delta t} \end{bmatrix} + \begin{bmatrix} \frac{\Delta c_{21}}{2} & 0 \\ 0 & \frac{\Delta c_{22}}{2} \end{bmatrix} \begin{bmatrix} \frac{\bar{T}_1^k - \bar{T}_1^{k-1}}{\Delta t} \\ \frac{T_2^k - T_2^{k-1}}{\Delta t} \end{bmatrix}$$

$$+ \begin{bmatrix} \frac{c_{31} + c_{32}}{2\Delta} & -\frac{c_{31} + c_{32}}{2\Delta} \\ -\frac{c_{31} + c_{32}}{2\Delta} & \frac{c_{31} + c_{32}}{2\Delta} \end{bmatrix} \begin{bmatrix} \bar{\psi}_1^k \\ \bar{\psi}_2^k \end{bmatrix} + \begin{bmatrix} \frac{c_{41} + c_{42}}{2\Delta} & -\frac{c_{41} + c_{42}}{2\Delta} \\ -\frac{c_{41} + c_{42}}{2\Delta} & \frac{c_{41} + c_{42}}{2\Delta} \end{bmatrix} \begin{bmatrix} \bar{T}_1^k \\ \bar{T}_2^k \end{bmatrix}$$

$$+ \begin{bmatrix} -\frac{1}{2}(c_{51} + c_{52}) \\ \frac{1}{2}(c_{51} + c_{52}) \end{bmatrix} + \begin{bmatrix} \frac{\Delta}{6}(2c_{61} + c_{62}) \\ \frac{\Delta}{6}(c_{61} + 2c_{62}) \end{bmatrix} = \begin{bmatrix} Q_m\big|_{z_1} \\ -Q_m\big|_{z_2} \end{bmatrix} \qquad (4.36)$$

Addition of the two lines of (4.36) results in the cancellation of the internal flux terms, yielding

$$\Delta\left[\frac{c_{11}}{2}(\bar{\psi}_1^k - \bar{\psi}_1^{k-1}) + \frac{c_{12}}{2}(\bar{\psi}_2^k - \bar{\psi}_2^{k-1}) + \frac{c_{21}}{2}(\bar{T}_1^k - \bar{T}_1^{k-1}) + \frac{c_{22}}{2}(\bar{T}_2^k + \bar{T}_2^{k-1})\right]$$

$$+ \Delta t \cdot \Delta\left[\frac{1}{2}(c_{61} + c_{62})\right] = \left[Q_m\big|_{z_1} - Q_m\big|_{z_2}\right] \cdot \Delta t \qquad (4.37)$$

In physical terms, this is equivalent to a mass balance for the element over one time step:

(length of element) x (average storage change)

+ (duration of time step) x (length of element) x (average sink strength)

= (duration of time step) x (sum of inward flux rates)

Assume that the sink terms and boundary fluxes are defined as constants (e.g., time averages) for the duration of the time step. Then, the only mass balance error resulting from Eq. (4.37) will result from the failure of the first term to describe exactly the storage change in the element. We proceed to illustrate the origin of this error.

We define the water mass stored in an element at time step k as follows:

$$\sum^k \equiv \frac{\Delta}{2} (S_{m1}^k + S_{m2}^k) \tag{4.38}$$

where S_m has been defined in Eq. (2.2). The subscripts 1 and 2 denote, as usual, the local (element) node index. Using Eq. (2.2) together with the values $\bar{\psi}_1^k$, $\bar{\psi}_2^k$, $\bar{T}_1^k$, and $\bar{T}_2^k$, it is possible to evaluate (4.38).

We now write an expression for the change in storage during a time step. It is

$$\sum^k - \sum^{k-1} = \frac{\Delta}{2} (S_{m1}^k - S_{m1}^{k-1} + S_{m2}^k - S_{m2}^{k-1}) \tag{4.39}$$

With our definition of element storage, then, the change in storage is a simple linear combination of the changes in the nodal values of S_m.

Comparing (4.37) and (4.39), we identify

$$S_{mi}^k - S_{mi}^{k-1} \cong c_{1i}(\bar{\psi}_i^k - \bar{\psi}_i^{k-1}) + c_{2i}(\bar{T}_i^k - \bar{T}_i^{k-1}) \tag{4.40}$$

where we recognize the possibility that the expression in (4.37) may not be exactly equivalent to (4.39), the "actual" storage change defined here. Recall that the c coefficients are defined, essentially, as

$$c_1 = \left.\frac{\partial S_m}{\partial \psi}\right|_T$$

$$c_2 = \left.\frac{\partial S_m}{\partial T}\right|_\psi \qquad (4.41)$$

(The node index will be omitted in the following development.) Thus, if c_1 and c_2 were constant in time, (4.40) would be an exact relation. Due to the non-linearity of storage as a function of ψ and T, however, a finite mass balance error will, in general, result. The magnitude of this error for a given problem can be estimated by the use of a Taylor series.

4.7.2 An Improved Estimate of $\frac{d\theta}{d\Psi}$

In the present problem, we have chosen (Section 4.3) to evaluate the coefficients in (4.36), including c_1 and c_2, at the k time level, i.e., at the end of the time step. In most situations, the quantity $\frac{d\theta}{d\Psi}$, which appears in both c_1 and c_2 (see (4.3a), (4.3b) and (3.20)), is by far the largest source of non-linearity and, hence, of error in (4.40). This is also true in the heat equation. By designing a special scheme to evaluate $\frac{d\theta}{d\Psi}$, we should be able to reduce the balance error significantly.

Let us for the moment treat c_1 and c_2 as though their variability were fully attributable to $\frac{d\theta}{d\Psi}$, i.e.

$$c_1 = \hat{c}_1 \frac{d\theta}{d\Psi}$$

$$c_2 = \hat{c}_2 \frac{d\theta}{d\Psi}$$

in which $\hat{c}_1$ and $\hat{c}_2$ are "relatively constant." Then (from (4.41)),

$$S_m^k = S_m^{k-1} + \int_{t^{k-1}}^{t_k} \left(\hat{c}_1 \frac{d\theta}{d\Psi} \frac{\partial \psi}{\partial \tau} + \hat{c}_2 \frac{d\theta}{d\Psi} \frac{\partial T}{\partial \tau} \right) d\tau \qquad (4.42)$$

According to our finite difference approximations,

$$\frac{\partial \psi}{\partial t} = \frac{\psi^k - \psi^{k-1}}{\Delta t}$$

$$\frac{\partial T}{\partial t} = \frac{T^k - T^{k-1}}{\Delta t}$$

i.e., the derivatives are constant. Then, (4.42) yields

$$S_m^k \cong S_m^{k-1} + \left[\hat{c}_1 \frac{\bar{\psi}^k - \bar{\psi}^{k-1}}{\Delta t} + \hat{c}_2 \frac{\bar{T}^k - \bar{T}^{k-1}}{\Delta t} \right] \int_{t^{k-1}}^{t_k} \frac{d\theta}{d\Psi} d\tau \qquad (4.43)$$

where the approximation sign results from the removal of $\hat{c}_1$ and $\hat{c}_2$ (which are not entirely constant) from the integral. The integral is

$$\int_{t^{k-1}}^{t^k} \frac{d\theta}{d\Psi} d\tau = \int_{\Psi^{k-1}}^{\Psi^k} \frac{d\theta}{d\Psi} \left(\frac{d\Psi}{d\tau} \right)^{-1} d\Psi$$

$$\cong \frac{\Delta t}{\Psi^k - \Psi^{k-1}} \int_{\Psi^{k-1}}^{\Psi^k} \frac{d\theta}{d\Psi} d\Psi$$

$$= \frac{\Delta t}{\Psi^k - \Psi^{k-1}} (\theta^k - \theta^{k-1})$$

It follows that a more exact expression than (4.40) with the c's simply evaluated at time t^k can be used. It is

$$S_m^k - S_m^{k-1} \cong \hat{c}_1 \left(\frac{\theta^k - \theta^{k-1}}{\Psi^k - \Psi^{k-1}} \right) (\bar{\psi}^k - \bar{\psi}^{k-1}) + \hat{c}_2 \left(\frac{\theta^k - \theta^{k-1}}{\Psi^k - \Psi^{k-1}} \right) (\bar{T}^k - \bar{T}^{k-1})$$

We have thus determined an "optimal" expression for $\frac{d\theta}{d\Psi}$ during a time

step. The coefficients $\hat{c}_1$ and $\hat{c}_2$ are still evaluated at the k time level, while the effective value of $\frac{d\theta}{d\Psi}$ is an average over the time step before time t^k. As before, the coefficients must be evaluated on the basis of a last estimate of the system state at time t^k in the iterative process outlined in Section 4.4.

Where $\frac{d\theta}{d\Psi}$ appears in d_1 and d_2, the heat storage coefficients, it will be evaluated by means of the same procedure.

4.7.3 Saturation Conditions

Following the lead of Neuman, et al. (1975), we recognize that the saturated regions will respond instantaneously to their boundary conditions, the storage coefficient being zero. In order to deal with nodes that might de-saturate during a given time step, it is convenient to lower Ψ^{k-1} to the bubbling pressure at saturated nodes at the beginning of a time step. Neuman, et al. (1975) show that this modification results in an equivalent statement of the original problem, and is therefore justifiable. In the present work, it was found that this procedure stabilized an otherwise non-converging iteration cycle in the numerical algorithm.

4.8 A FORTRAN Code for Execution of the Numerical Model

The numerical method described in this chapter has been coded in the FORTRAN language for computer execution. The program, SPLaSHWaTr1, is documented in Appendix A. The FORTRAN listing appears in Appendix B.

Chapter 5

TESTS OF THE NUMERICAL METHOD

5.1 Introduction

The purpose of this chapter is to evaluate the performance of the numerical procedure outlined in Chapter 4. A related purpose is to examine the validity of certain models of the soil properties proposed in Chapter 3.

We may identify three criteria for the evaluation of the numerical procedure. The first is accuracy, which we shall loosely define as the ability of the numerical model to reproduce satisfactorily the true solution to a mathematical statement of a problem. More specific definitions of accuracy are stability -- the ability of a procedure to dampen errors in a solution as computations progress -- and convergence -- the tendency toward perfect accuracy that results from finer spatial and temporal discretization. For the complex, non-linear problem treated here, analytical demonstrations of stability and convergence appear difficult, thus we shall infer that the method is stable and convergent if it is accurate in a representative set of applications.

A second criterion is consistency, defined herein as the successful preservation of total mass and energy by the numerical scheme. This issue has already been addressed theoretically in Section 4.7. The method proposed there for evaluation of $\frac{d\theta}{d\Psi}$ will

be evaluated in this chapter.

Convergence of the iterative solution procedure at each time step will be defined as the automatic termination of the iterative scheme presented in Section 4.4, i.e., the decay toward zero of the difference between successive iterations. Though this is not the traditional usage of "convergence" (see above), we shall employ it here. Successful completion of a simulation run implies convergence for that problem.

The validity of the numerical model, and of the soil property representations, will be evaluated in terms of the criteria set forth above. An attempt is made to isolate the question of validity of the numerical model from that of validity of the physical theory. The latter is judged sufficiently well-established for application. This isolation is accomplished by comparing the numerical solutions to solutions obtained analytically or quasi-analytically, thereby avoiding the complication of data uncertainty. The exception is the example of hysteretic redistribution given in Section 5.3, a problem for which no analytic solution has yet been given in the literature.

Since analytic solutions of the entire set of equations are not available, it is necessary to test different features of the numerical model separately. Having established the validity of the model for simulating various processes independently, it will be assumed that the model may be used for more complex situations. The demonstrated ability to handle strong non-linearities (Section 5.2) and highly-coupled problems (Section 5.5) helps justify this assumption.

5.2 Isothermal Infiltration into Yolo Light Clay

The problem of infiltration into Yolo light clay was solved by Philip (1957) using a "quasi-analytic" solution procedure. His classic example has since become a standard against which many subsequent solutions have been compared.

This problem is solved here neglecting the vapor and thermal effects, in order to be consistent with Philip's solution. The governing equation is thus

$$\frac{d\theta}{d\psi}\frac{\partial\psi}{\partial t} = \frac{\partial}{\partial z}\left[K\left(\frac{\partial\psi}{\partial z} + 1\right)\right]$$

with the following boundary and initial conditions:

$\psi = -600$ cm	$t = 0$	$0 \geq z \geq -50$ cm
$\psi = 0$ cm	$t > 0$	$z = 0$
$\psi = -600$ cm	$t > 0$	$z = -50$ cm

Although Philip (1957) considers a semi-infinite medium, we may use a finite column for times before the wetting front nears the lower boundary.

Haverkamp et al. (1977) have fitted equations closely to the data describing the Yolo light clay. Their expression for the moisture retention curve is

$$\theta = \begin{cases} 0.124 + \dfrac{274.2}{739 + (\ln(-\psi))^4} & \psi < -1 \text{ cm} \\ 0.495 & \psi \geq -1 \text{ cm} \end{cases} \qquad (5.1)$$

The relative hydraulic conductivity is given as

$$K_r(\psi) = \frac{124.6}{124.6 + (-\psi)^4} \tag{5.2}$$

These functions are illustrated by the circles in Figures 5.1 and 5.2. Saturated conductivity is given as 1.23×10^{-5} cm s^{-1}.

Equation (3.8) was also fitted to the $\theta(\psi)$ data used by Haverkamp et al. (1977). The parameters used are

$a_1 = 0.690$ $\quad$ $s_1 = 0.172$

$a_2 = 0.375$ $\quad$ $s_2 = 0.0536$

$M = 38$ $\quad$ $\theta_u = 0.495$

This relation is plotted in Figure 5.1 for comparison with Eq. (5.1). Note that the main difference is due to the inability of (3.8) to reproduce the observed curve near saturation.

Using Eq. (3.38) and Eq. (3.8) with the parameters above, we may derive numerically a relative hydraulic conductivity function. This is plotted in Fig. 5.2 for comparison with (5.2). Note the rather large discrepancy in the drier region. This is not unusual for a derived conductivity function.

The numerical model was used to simulate this isothermal infiltration problem. The top 5 cm. was divided into ten equal-sized elements; the next 25 cm. into twenty-five elements; and the bottom 20 cm. into ten elements. The time step size in this and subsequent examples was controlled automatically by a rule that sought to keep the

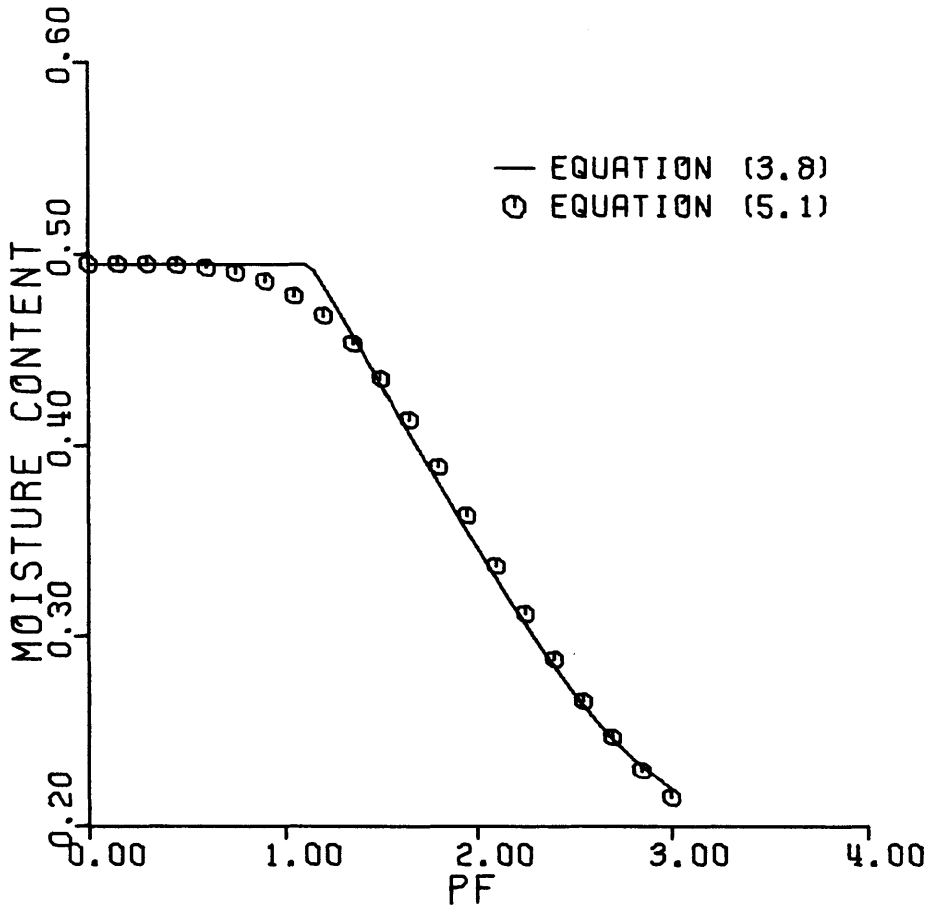


FIGURE 5.1. MAIN WETTING CURVE OF YOLO LIGHT CLAY.

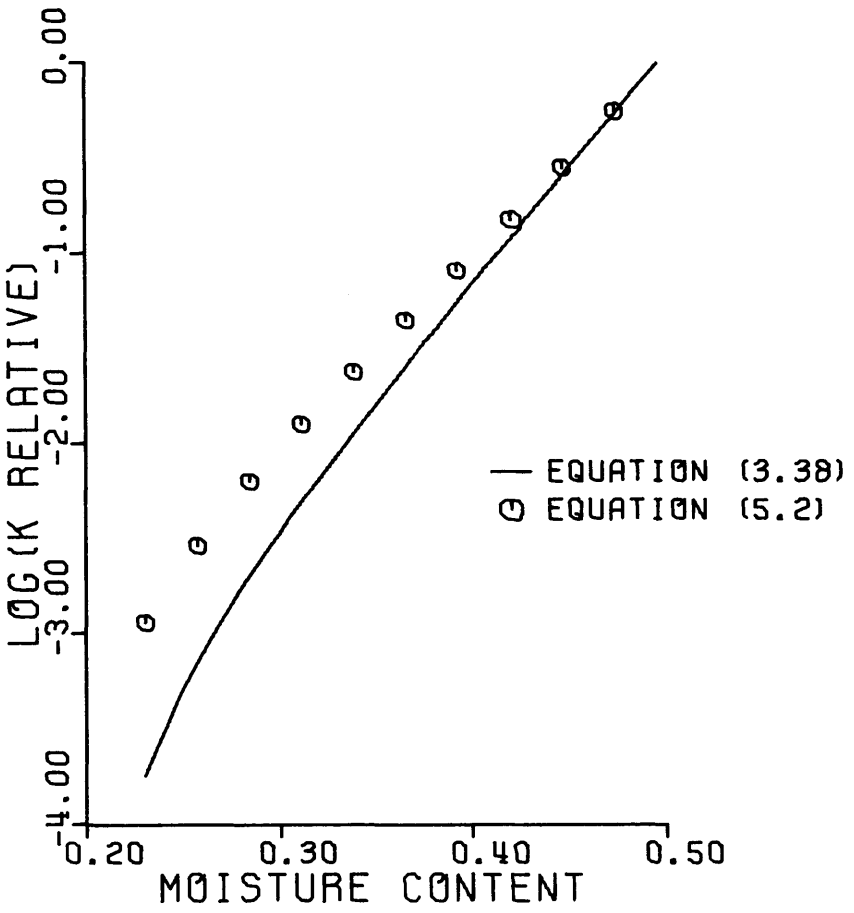


FIGURE 5.2. RELATIVE HYDRAULIC CONDUCTIVITY OF YOLO LIGHT CLAY.

number of iterations required for convergence near some pre-specified number from 2 to 10.

The solution computed using (5.1) and (5.2) is plotted in Figure 5.3. The agreement with Philip's solution is very good. Computation time was about 38 seconds on an IBM 370/168 machine for the entire simulation. The mass balance error (defined as the relative difference between cumulative inflow and total storage change) was about 0.5 per cent. The consistency could be improved by reducing the convergence criterion, with a consequent increase in computation time.

This problem was also solved without the special scheme for evaluation of $\frac{d\theta}{d\Psi}$. Given the same total execution time, the mass balance error increased by almost an order of magnitude.

The problem was also solved using the soil properties defined by (3.8) and (3.38). The solution, plotted in Figure 5.4, is rather good, especially when one considers the error in the calculated relative conductivity function shown in Figure 5.2. Note that the numerical solution incorrectly predicts saturation to a significant depth; this is a result of the "corner" at saturation given by (3.8). Despite this error in the shape of the front, the depth of the front is rather well predicted.

Finally, we consider the problem of ponded water at the surface, corresponding to a boundary condition of positive matric potential at the surface. A solution is again given by Philip (1958). For this problem, we use Equations (5.1) and (5.2) to represent the

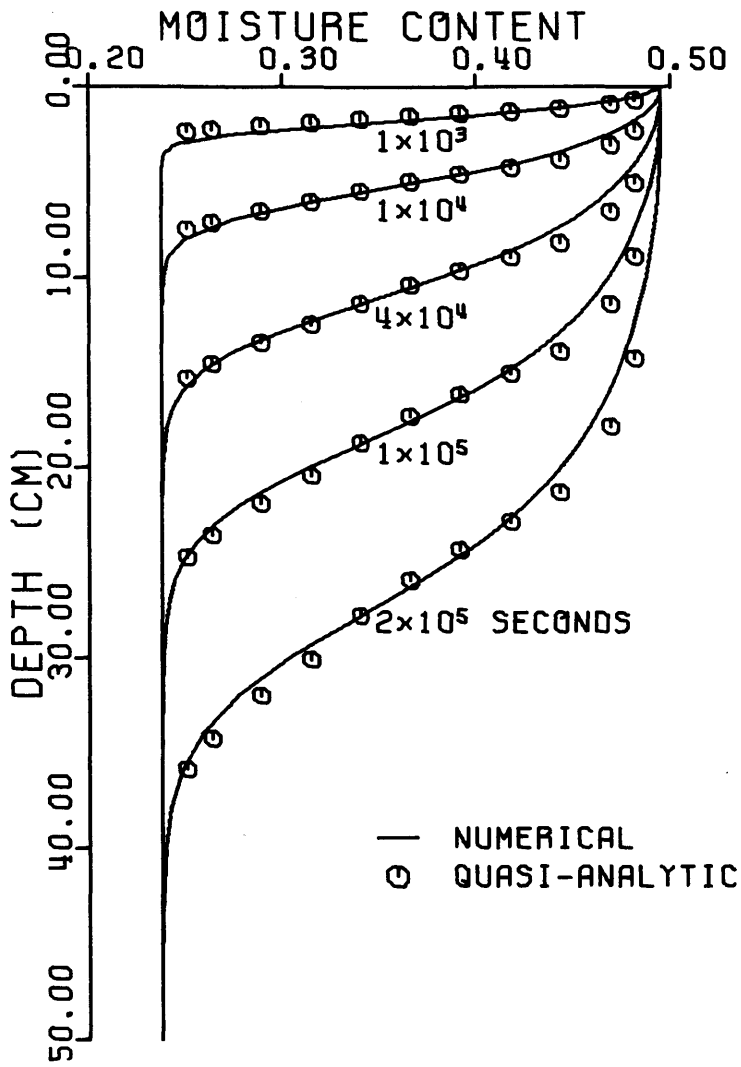


FIGURE 5.3. INFILTRATION INTO YOLO LIGHT CLAY BASED ON EQS. (5.1) AND (5.2).

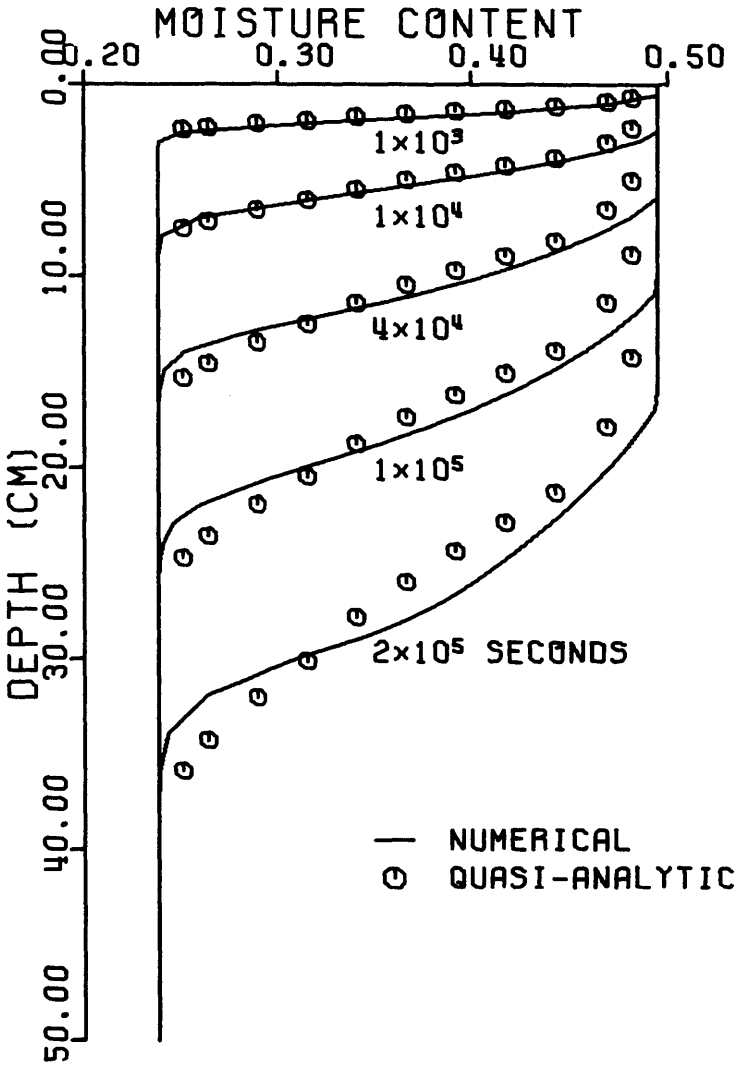


FIGURE 5.4. INFILTRATION INTO YOLO LIGHT CLAY BASED ON EQS. (3.8) AND (3.38).

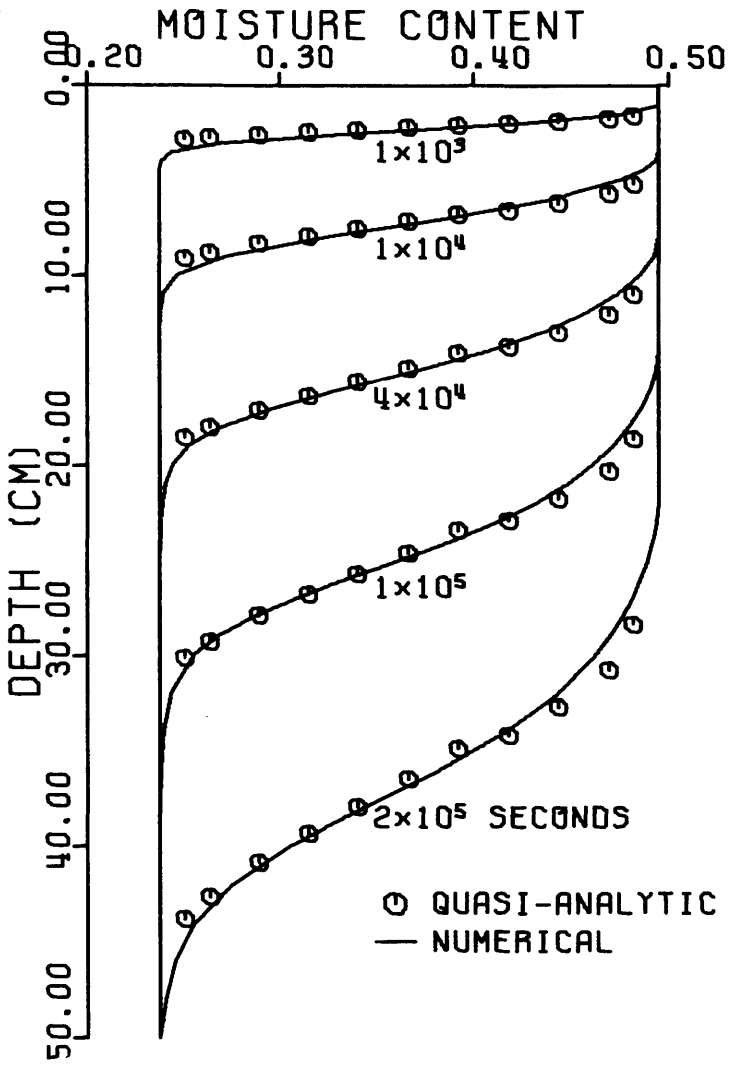


FIGURE 5.5. INFILTRATION INTO YOLO LIGHT CLAY WITH PONDING, BASED ON EQS. (5.1) AND (5.2).

soil properties. The surface boundary condition is changed to

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

The solution is plotted in Figure 5.5. Agreement with the quasi-analytic solution is very good.

5.3 Isothermal Infiltration and Redistribution in Uplands Sand

Staple (1969) studied the problem of soil moisture redistribution following infiltration into an initially-dry sand both experimentally and numerically. His main wetting curve for Uplands sand was fitted to Equation (3.8). Using (3.8), the scanning curves were derived using the model presented in Section 3.2.4, and the relative conductivity function was derived using Equation (3.38). These soil properties are plotted in Figure 5.6. Two of the primary drying scanning curves are shown. Saturated conductivity is $2.15 \times 10^{-3} \text{cm s}^{-1}$.

The experiment performed by Staple (1969) was to allow one inch of water to infiltrate "with a head not exceeding a depth of 1 cm" into sand that was very dry initially. Infiltrated water was then allowed to percolate downward in the column and measurements were made. (Unfortunately, the time of the infiltration phase was not given.) The following initial and boundary conditions were employed here:

$\psi = -2.25 \times 10^6$ cm	$t = 0$	$0 \geq z \geq -30$ cm
$\psi = 0.5$ cm	$t_i \geq t > 0$	$z = 0$
$q_\ell = 0$	$t > t_i$	$z = 0$
$\psi = -2.25 \times 10^6$ cm	$t > 0$	$z = -30$ cm

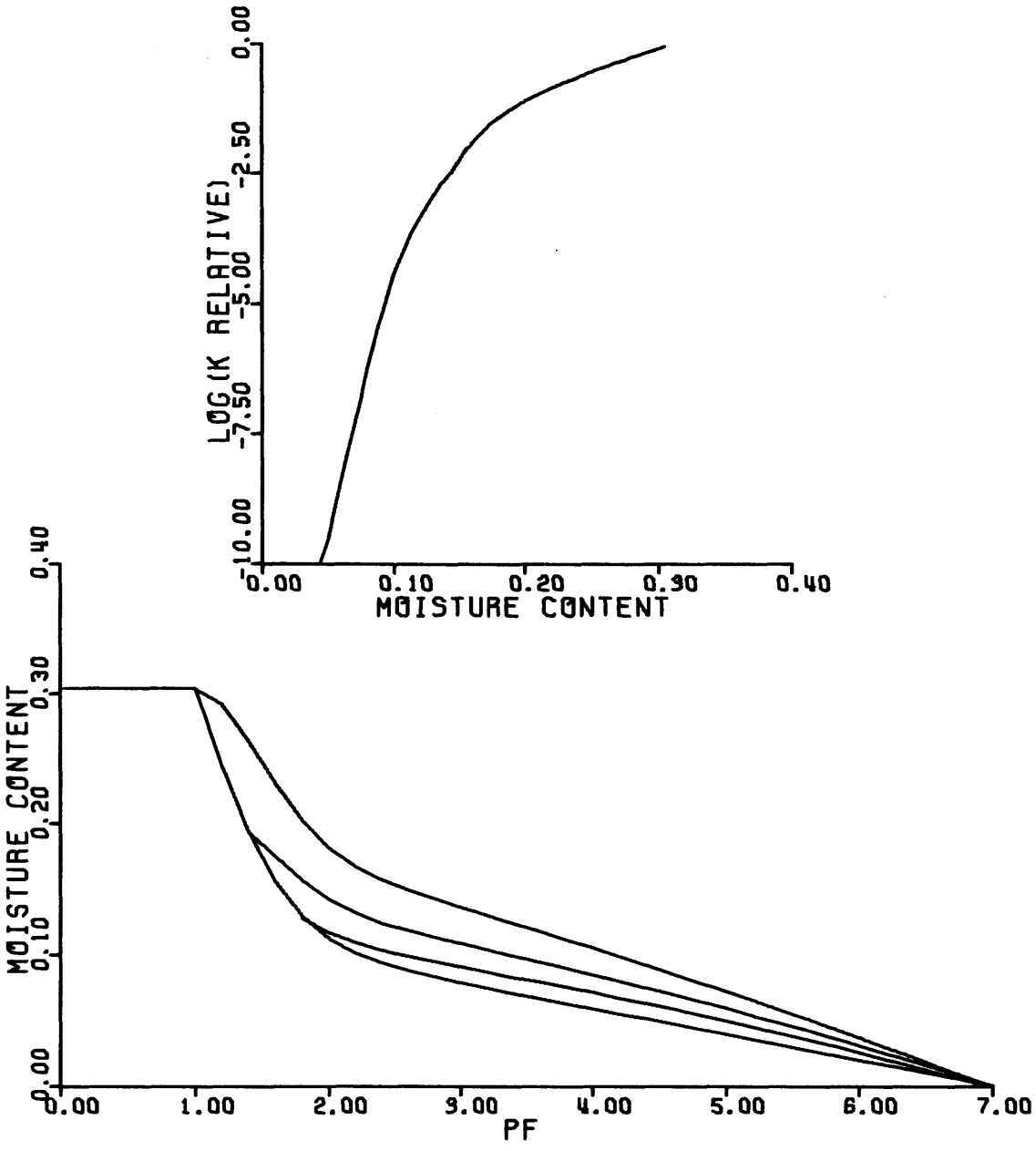


FIGURE 5.6. HYDRAULIC PROPERTIES OF UPLANDS SAND.

where t_i is the time at which the cumulative infiltration is equal to 2.54 cm of water. A uniform finite element discretization of 1 cm. was employed.

The solution to this problem, using the properties in Figure 5.6, is plotted in Figure 5.7 for two values of the redistribution time, $t - t_i$. (The value of t_i was 222 seconds.) The profiles are in general agreement, though the shapes of the fronts are not predicted especially well. The anomalously high moisture content at the surface at 90 seconds is apparently related to swelling of the soil, a phenomenon that was neglected in the formulation of the model. The other errors are probably attributable to approximations in the soil property data.

The same problem was also solved by assuming that drainage occurred along the main wetting curve, i.e., hysteresis was neglected. The results appear in Figure 5.8, where it is apparent that redistribution proceeds much too quickly. Some insight into the difference between Figures 5.7 and 5.8 may be gained by examining the moisture retention curves in Figure 5.6. When hysteresis is neglected, drainage of the wetted region proceeds along the lower curve, the main wetting curve, and much water is released readily to seep into the lower depths. When hysteresis is properly modeled, drainage occurs along the drying curves, which specify more moisture retention at a given value of pF. The wetted region thus drains more slowly.

It should be noted that infiltration into a very dry sand

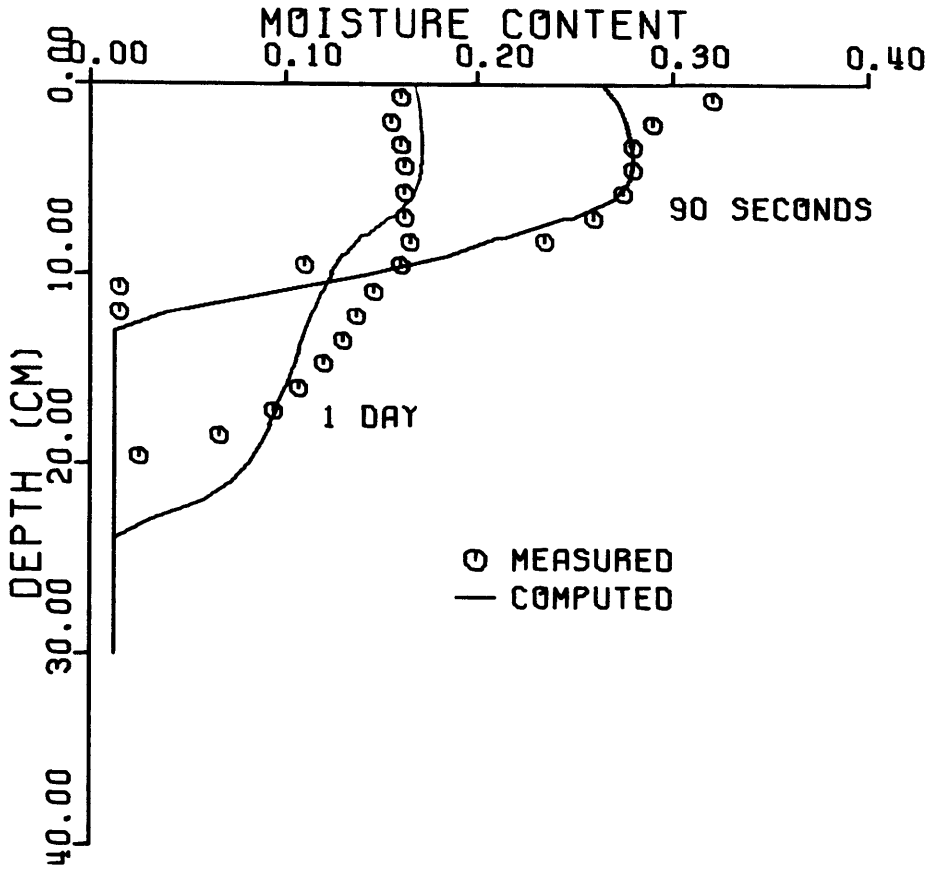


FIGURE 5.7. REDISTRIBUTION IN UPLANDS SAND, CONSIDERING HYSTERESIS.

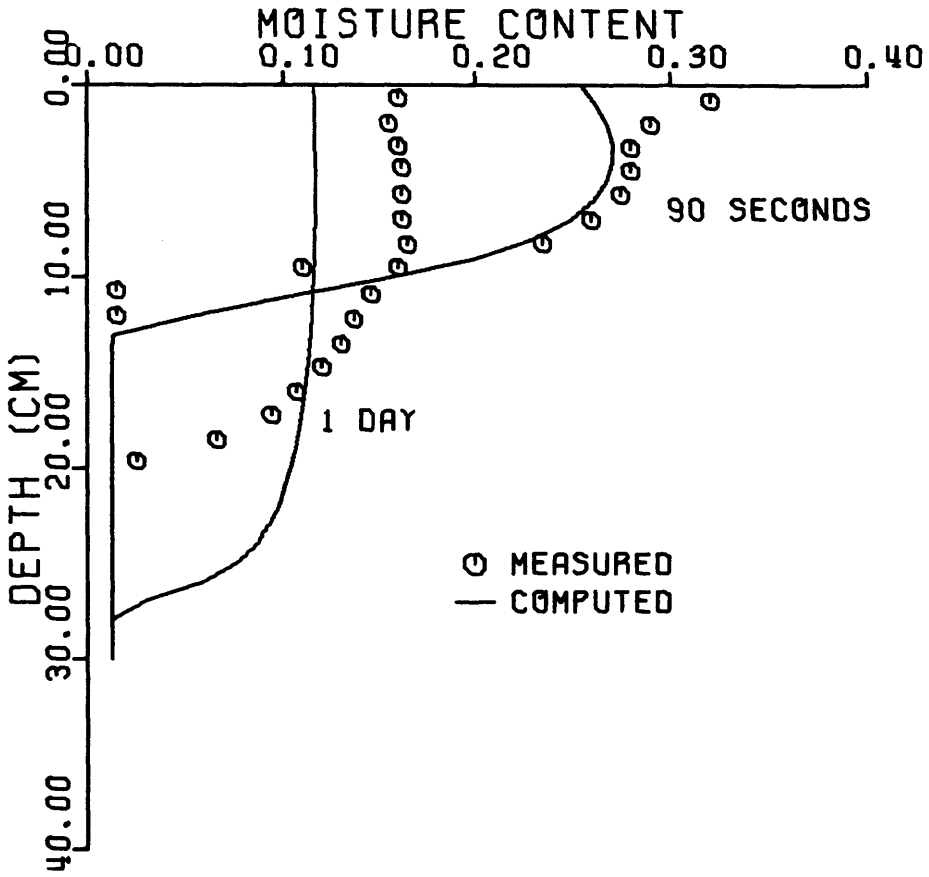


FIGURE 5.8. REDISTRIBUTION IN UPLANDS SAND, NEGLECTING HYSTERESIS.

is one of the more difficult problems in numerical modeling of soil water dynamics, often presenting serious problems with respect to accuracy and stability (Narasimhan and Witherspoon, 1978). Nevertheless, the present model handled the infiltration stage in this example quite well.

5.4 Advection and Dispersion of Heat in a Uniform Moisture Flow

Having looked at some problems in which the moisture field is subjected to large changes in boundary conditions, or stresses, we now turn to a problem in which the thermal regime is highly stressed. We consider the problem of advection and dispersion of heat in a saturated porous medium with a large flow rate. Though the rather extreme example to be proposed is not likely to occur in a natural environment, it does serve as a test of the model's validity.

In a non-elastic, homogeneous, saturated porous medium the one-dimensional heat conservation equation reduces to

$$C\frac{\partial T}{\partial t} = \lambda\frac{\partial^2 T}{\partial z^2} - c_\ell q_\ell \frac{\partial T}{\partial z}$$

in which q_ℓ is the vertical component of $\underline{q}_\ell$, and all of the coefficients are constants. Let these coefficients take the following values:

$$C = 9.79 \times 10^{-1} \text{ cal cm}^{-3} \text{K}^{-1}$$

$$\lambda = 3.6 \times 10^{-3} \text{cal cm}^{-1}\text{s}^{-1}\text{K}^{-1}$$

$$c_\ell q_\ell = -9.98 \times 10^{-4} \text{cal cm}^{-2}\text{s}^{-1}\text{K}^{-1}$$

The thermal properties, C and λ, have values typical for soils, while

the advection term, $c_\ell q_\ell$, represents a strong liquid flow. It is well-known (e.g., Pinder and Gray, 1977, p. 148) that the advection term is the source of difficulty in solving this equation with finite difference and finite element methods, so this extreme example was chosen.

The problem statement is completed with the following initial and boundary conditions:

$$T = T_o \qquad t = 0 \qquad 0 \geq z \geq -50\text{cm}$$

$$-\lambda\frac{\partial T}{\partial z} + c_\ell q_\ell (T - T_o) = c_\ell q_\ell (T_1 - T_o) \qquad t > 0 \qquad z = 0$$

$$\frac{\partial T}{\partial z} = 0 \qquad t > 0 \qquad z = -50\text{cm}$$

in which

$$T_o = 20^oC$$
$$T_1 = 21^oC.$$

Note that the surface boundary condition is a statement of a specified heat flux at the surface. It is a non-homogeneous, mixed-type boundary condition. Physically, this problem represents a sudden rise to T_1 of the temperature of the liquid supplied to the soil at its surface.

For a semi-infinite domain, the solution to this problem can be found by application of Fourier transforms. Defining

$$D = \frac{\lambda}{C}$$

and

$$u = \frac{c_\ell q_\ell}{C}$$

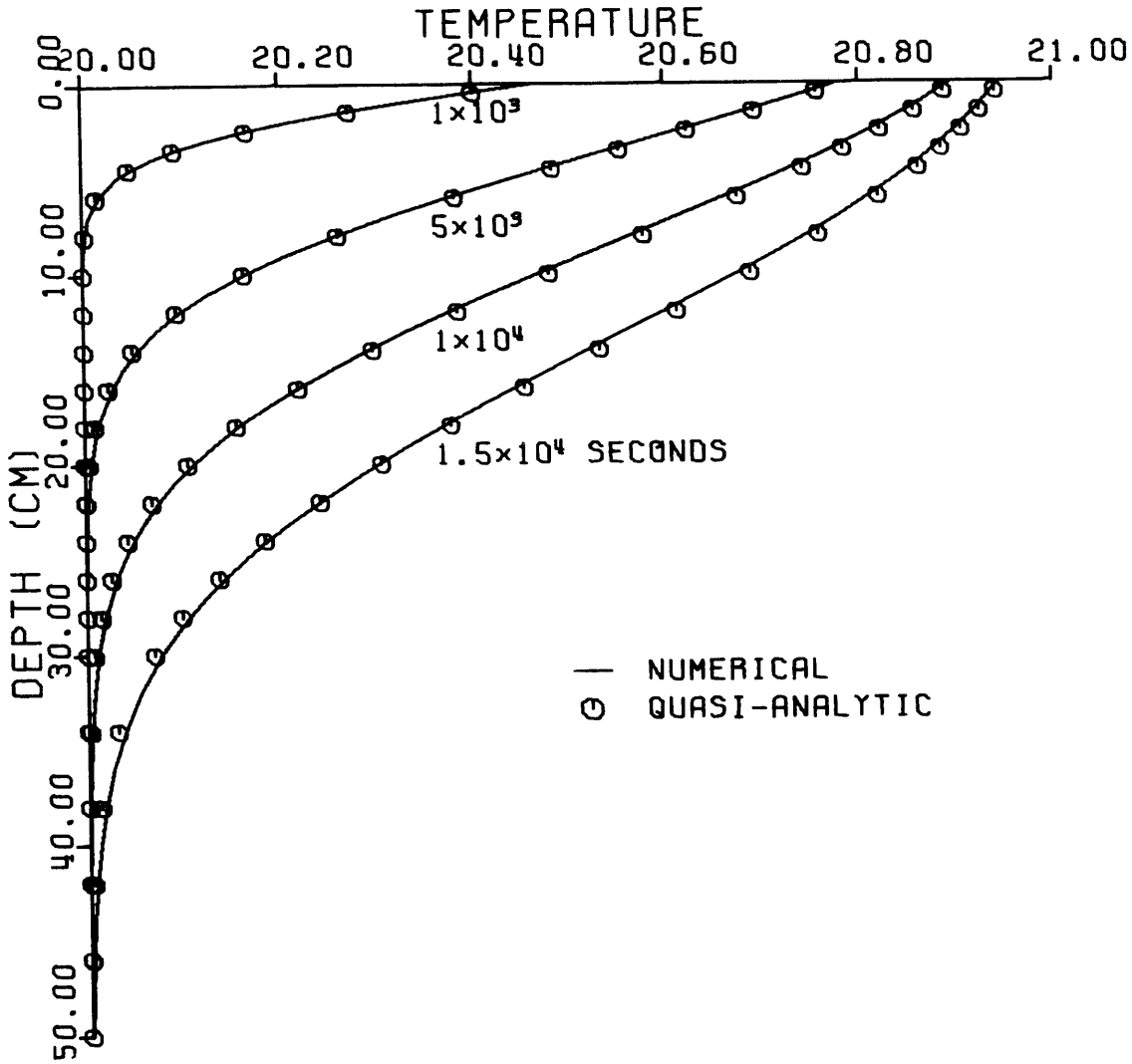


FIGURE 5.9. ADVECTION AND DISPERSION OF HEAT IN A SATURATED POROUS MEDIUM.

and recalling that we have defined z negative downward from the surface, we may write the analytic solution to this problem as

$$T = T_o + \frac{1}{2}(T_1 - T_o)\left\{\operatorname{erfc}\left(\frac{-z-ut}{2\sqrt{Dt}}\right) - \frac{u}{D}\left(-z+ut+\frac{D}{u}\right)\right.$$

$$\left.\cdot\exp\left(\frac{-zu}{D}\right)\operatorname{erfc}\left(\frac{-z+ut}{2\sqrt{Dt}}\right) + \left(\frac{4u^2t}{D\pi}\right)^{\frac{1}{2}}\exp\left[-\left(\frac{-z-ut}{2\sqrt{Dt}}\right)^2\right]\right\} \qquad (5.3)$$

in which erfc is the complementary error function and exp is the exponential function.

The problem was solved numerically using the same discretization as for the Yolo light clay examples. The time step size ranged from an initial value of 10 seconds to a final value of about 500 seconds. The solution is plotted, along with (5.3), in Fig. 5.9. The agreement is excellent. The relative error in the heat balance was less than 10^{-7} for this linear problem.

5.5 Coupled Diffusion of Heat and Vapor in a Very Dry Porous Medium

The final example presented here is designed to test the ability of the model to simulate properly the dynamics of vapor-dominated systems with strong coupling between moisture and heat fields.

A very dry soil column at some equilibrium temperature and matric potential (and therefore vapor density) is subjected to a sudden increase in vapor density at one end, while the temperature is held at its original value. The other end is closed to heat and

moisture flow. Vapor will diffuse into the column, condense, and release latent heat. There will thus be a temporary rise in the temperature of the medium, though it will eventually return to the value at the boundary as heat diffuses back out of the column.

In the absence of liquid flow, and using the "simple theory" of vapor transport described in Chapter 2, the governing equations reduce to

$$\frac{\theta_a}{\rho_\ell}\frac{\partial \rho_v}{\partial t} + \left(1 - \frac{\rho_v}{\rho_\ell}\right)\frac{\partial \theta}{\partial t} = D^* \frac{\partial^2 \rho_v}{\partial z^2} \tag{5.4}$$

$$C\frac{\partial T}{\partial t} - \rho_\ell L \frac{\partial \theta}{\partial t} = \lambda \frac{\partial^2 T}{\partial z^2} \tag{5.5}$$

in which

$$D^* = \frac{D_a}{\rho_\ell}\alpha\,\theta_a$$

The heat of wetting has not been included; it is equivalent mathematically to L. Expanding the θ derivative,

$$\frac{\partial \theta}{\partial t} = \frac{\partial \theta}{\partial \rho_v}\bigg|_T \frac{\partial \rho_v}{\partial t} + \frac{\partial \theta}{\partial T}\bigg|_{\rho_v} \frac{\partial T}{\partial t}$$

this system may be written as a pair of nonlinear diffusion equations in two unknowns, coupled through their storage terms,

$$\left[\frac{\theta_a}{\rho_\ell} + \left(1 - \frac{\rho_v}{\rho_\ell}\right)\left.\frac{\partial\theta}{\partial\rho_v}\right|_T\right]\frac{\partial\rho_v}{\partial t} + \left(1 - \frac{\rho_v}{\rho_\ell}\right)\left.\frac{\partial\theta}{\partial T}\right|_{\rho_v}\frac{\partial T}{\partial t}$$

$$= D^*\frac{\partial^2\rho_v}{\partial z^2} \tag{5.6}$$

$$\left(C - \rho_\ell L\left.\frac{\partial\theta}{\partial T}\right|_{\rho_v}\right)\frac{\partial T}{\partial t} - \rho_\ell L\left.\frac{\partial\theta}{\partial\rho_v}\right|_T\frac{\partial\rho_v}{\partial t}$$

$$= \lambda\frac{\partial^2 T}{\partial z^2} \tag{5.7}$$

For small perturbations about a basic state--the initial conditions--we may linearize Eqs. (5.6) and (5.7) by evaluating the coefficients at the basic state. Crank (1956, p. 306) has shown how a dependent variable transformation may then be used to convert these equations to a pair of independent diffusion equations. For further details, the reader is directed to that reference, which also contains the solutions to the resulting diffusion equations.

The boundary and initial conditions are

$$\left.\begin{aligned} \rho_v &= \rho^* \\ T &= T^* \end{aligned}\right\} \qquad t = 0 \qquad 0 \geq z \geq -5 \text{ cm}$$

$$\left.\begin{aligned} \rho_v &= \rho^* + \Delta\rho \\ T &= T^* \end{aligned}\right\} \qquad t > 0 \qquad z = 0$$

$$\left.\begin{aligned} \frac{\partial \rho_v}{\partial z} &= 0 \\ \frac{\partial T}{\partial z} &= 0 \end{aligned}\right\} \qquad t > 0 \qquad z = -5 \text{ cm}$$

In the numerical model, conditions on ρ_v are implemented by applying the appropriate conditions on ψ. The following numerical values were employed for the analytic solution:

$$\rho^* = 6.342 \times 10^{-6} \text{ g cm}^{-3}$$

$$\Delta\rho = 6.67 \times 10^{-7} \text{ g cm}^{-3}$$

$$T^* = 20^{\circ}\text{C}$$

General expressions described in Chapter 3 and employed in the numerical model were used to evaluate the coefficients in (5.6) and (5.7), given the initial conditions and typical parameter values, for use in the analytic solution. The same parameters were used as input to the numerical model, which did not use the linearization. A ten-element discretization was used.

The numerical and analytic solutions for temperature and vapor density are shown in Fig. 5.10. The agreement is good. The small errors are probably attributable to the spatial discretization scheme,

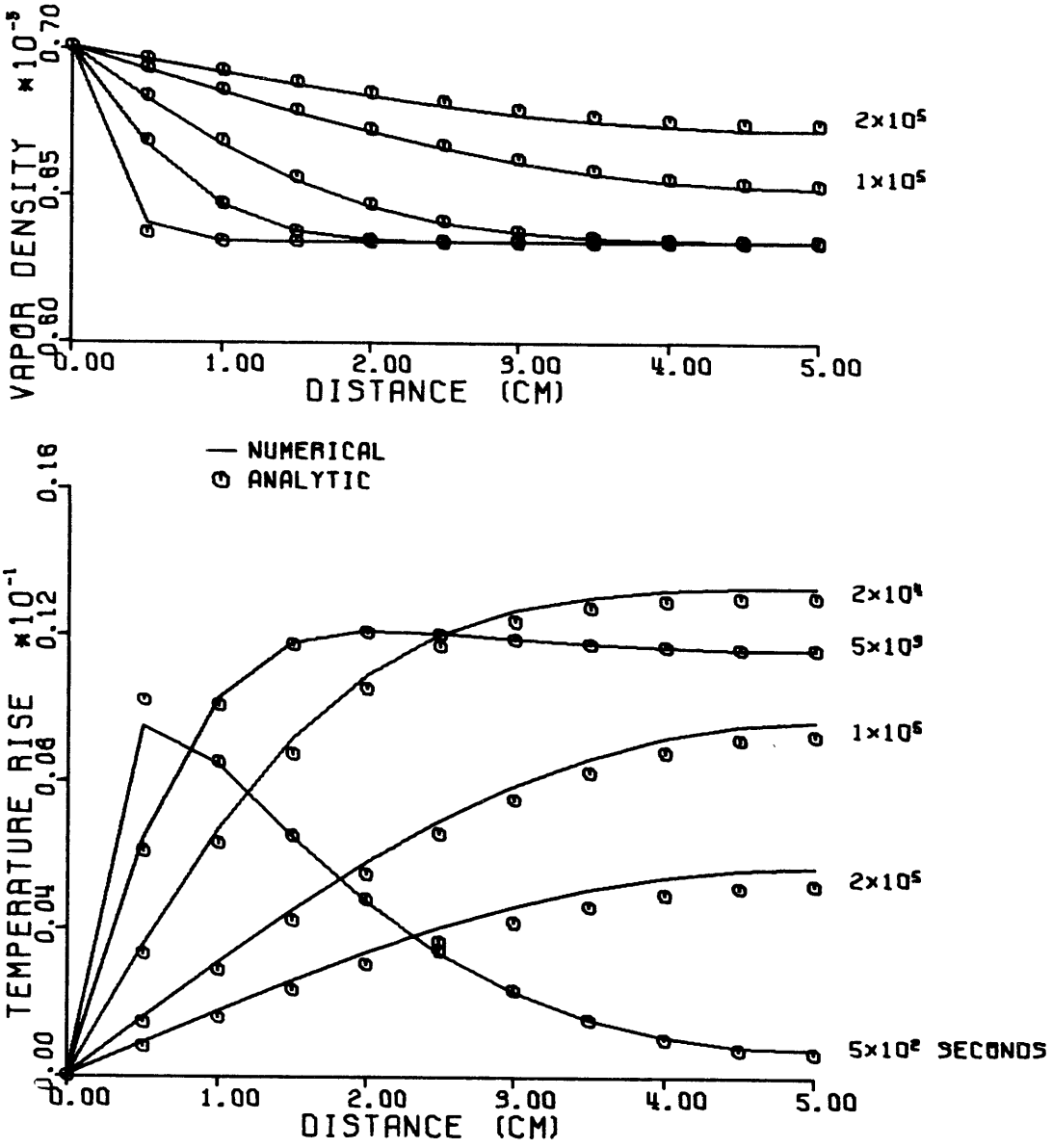


FIGURE 5.10. COUPLED DIFFUSION OF HEAT AND MOISTURE IN A VERY DRY POROUS MEDIUM.

to the non-linearities neglected in the analytic solution, and to the somewhat different treatments of vapor transport (simple theory and theory of Philip and de Vries). Mass and heat balance errors were both on the order of 10^{-4} at the end of the simulation.

The same problem was run using the standard method for evaluating $\frac{d\theta}{d\Psi}$. The energy balance errors were about the same, as might be expected, since energy storage is essentially linear in this problem. The mass balance error, however, was increased to about 10^{-2} given the same execution time. Therefore, the method proposed in Section 4.7.2 for evaluating the storage coefficient reduces the mass balance error by two orders of magnitude.

5.6 Summary

The numerical model described in Chapter 4 has been subjected to a variety of tests for validity. These tests included some extremely nonlinear problems and a problem in which the heat and moisture fields were strongly coupled. The physical processes of liquid and vapor transport, and of heat transport (sensible and latent) by conduction and by advection, were all simulated using the model. In addition, the models of soil properties developed in Chapter 3 were tested.

In all cases, the numerical model converged to solutions that preserve mass and heat well. Comparisons to several analytic and quasi-analytic solutions demonstrate that the model is very accurate.

A new method for iterative estimation of the storage coefficients in the numerical model appears to yield significant improve-

ments in mass and energy balances compared to traditional techniques. Further investigation of this procedure is warranted.

Chapter 6

SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS FOR FURTHER RESEARCH

6.1 Summary

A general mathematical description of the coupled dynamics of moisture and heat flow in porous media has been presented. The Philip and de Vries (1957) formulation has been re-cast in terms of matric potential, accounting for the complications of hysteresis and of soil heterogeneities. A practical model for simulating hysteresis, based on the work of Mualem (1977), has been proposed. In addition, a suggested extension of Mualem's theory allows incorporation of the effects of temperature on moisture retention.

A numerical (finite element) method for the solution of the general conservation equations has been outlined. It includes a new procedure for evaluating the storage coefficients that appears to offer very large improvements in mass and energy balances in nonlinear problems.

A FORTRAN computer code has been developed in order to execute the complex numerical solution procedure. The numerical method and the computer code have been shown to perform well in simulating highly-coupled, hysteresis-affected, or very nonlinear problems. Comparisons to analytic solutions verified that the model will simulate correctly the transport of mass in both the liquid and vapor phases and the transport of sensible and latent heat by conduction and by advection. The mass and energy balance properties of the model are excellent.

6.2 Conclusions

The myriad experimental and theoretical studies conducted by soil scientists, hydrologists, geologists, physicists, etc. during the past years have supplied us with much information about transport processes in porous media. Although there remain even today basic physical questions that are not entirely resolved, it appears possible to synthesize the available knowledge into a coherent theoretical framework that will allow us to study systematically the dynamics of coupled mass-energy systems in soils that interact with the wide spectrum of naturally-occurring climates and vegetal systems. The exposition of such a framework and the demonstration of its viability are the goals toward which the present work has been directed.

6.3 Recommendations for Further Research

The theory presented in this work has been precise, but cumbersome. Physically-based studies of large-scale, complex systems--such as the air-land-sea system that provides us with weather and climate--cannot be expected to incorporate the exact physical details of millions of tiny system components. Even if an "infinite" computer were available, the problem of managing data would easily overwhelm us, and the assessment of initial conditions would be hopeless.

Clearly, simplified representations of the soil column and of its aggregate areal response to atmospheric forcing must be developed. Employing the model developed in this paper as the apparatus for a potentially infinite number of experiments on the "true" physical system, it should be possible to develop parameterizations that

faithfully reproduce the behavior of the one-dimensional system.

Another major problem is that of studying the interaction of the soil with the overlying vegetation and atmosphere. Tied in with this is the question of how properly to treat the problem of areal inhomogeneity of soil and vegetal response. Analysis of these problems would probably be facilitated by the development of simplified models of the components, such as the soil parameterization that has been mentioned in the preceding paragraph.

REFERENCES

Adam, K. M. and A. T. Corey, Diffusion of entrapped gas from porous media, Colo. State Univ. Hydrology Paper No. 27, 1968.

Bear, J., Notes on Hydraulics of Groundwater in Aquifers, Lecture notes for CE523, Princeton University, 1977.

Brooks, R. H. and A. T. Corey, Hydraulic properties of porous media, Colo. State Univ. Hydrology Paper No. 3, 1964.

_____, Properties of porous media affecting fluid flow, Journal of the Irrigation and Drainage Division, ASCE, 92 (IR2), 1966.

Buckman, H. O. and N. C. Brady, The Nature and Properties of Soils, Macmillan Co., New York, 1969.

Burdine, N. T., Relative permeability calculations from pore size distribution data, Petroleum Trans., Am Inst. of Mining, Metallurgical and Petroleum Engineers, 198, 71-77, 1953.

Campbell, G. S. and W. H. Gardner, Psychrometric measurement of soil water potential: Temperature and bulk density effects, Soil Sci. Soc. Am. Proc., 35 (1), 8-12, 1971.

Carman, P. C., Properties of capillary-held liquids, J. Phys. Chem., 57, 56-64, 1953.

Carman, P. C. and F. A. Raal, Physical adsorption of gases on porous solids. I. Comparison of loose powders and porous plugs, Proc. Royal Soc. London, A:209, 59-69, 1951.

Childs, E. C., The ultimate moisture profile during infiltration in a uniform soil, Soil Sci., 97(3), 173-178, 1964.

Childs, E. C. and N. Collis-George, The permeability of porous materials, Proc. Roy. Soc. London, A:201, 392-405, 1950.

Corey, W., Mathematical study of the first stage of drying of a moist soil, Soil Sci. Soc. Am. Proc., 27(2), 130-134, 1963.

Crank, J., The Mathematics of Diffusion, Oxford University Press, New York, 1956.

Eagleson, P. S., Dynamic Hydrology, McGraw-Hill, New York, 1970.

_____, Climate, soil, and vegetation, 3. A simplified model of soil moisture movement in the liquid phase, Water Resour. Res., 14(5), 722-730, 1978.

Edlefsen, N. E. and A. B. C. Anderson, Thermodynamics of soil moisture, Hilgardia, 15(2), 1943.

Elrick, D. E. and D. H. Bowman, Note on an improved apparatus for soil moisture flow measurements, Soil Sci. Soc. Am. Proc., 28(3), 450-453, 1964.

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 Resour. Res., 5(1), 153-171, 1969.

_____, Three-dimensional, transient, saturated-unsaturated flow in a groundwater basin, Water Resour. Res., 7(2), 347-366, 1971.

Gardner, W. R., Some steady-state solutions of the unsaturated moisture flow equation with application to evaporation from a water table, Soil Sci., 85(4), 228-232, 1958.

Geiger, R., The Climate Near the Ground (translated from the fourth German edition), Harvard University Press, Cambridge, 1965.

Gillham, R. W., A. Klute, and D. F. Heermann, Hydraulic properties of a porous medium: measurement and empirical representation, Soil Sci. Soc. Am. Proc., 40(2), 203-207, 1976.

Green, R. E. and J. C. Corey, Calculation of hydraulic conductivity: A further evaluation of some predictive methods, Soil Sci. Soc. Am.Proc., 35(1), 3-8, 1971.

Hadas, A., Evaluation of theoretically predicted thermal conductivities of soils under field and laboratory conditions, Soil Sci. Soc. Am. Proc., 41(3), 460-466, 1977.

Hanks, R. J. and S. A. Bowers, Numerical solution of the moisture flow equation for infiltration into layered soils, Soil Sci. Soc. Am. Proc., 26(6), 530-534, 1962.

Haverkamp, R., M. Vauclin, J. Touma, P. J. Wierenga, and G. Vachaud, A comparison of numerical simulation models for one-dimensional infiltration, Soil Sci. Soc. Am. Proc., 41(2), 285-294, 1977.

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

_____, Computer Simulation of Soil-Water Dynamics, International Development Research Centre, Ottawa, 1977.

Jackson, R. D., Water vapor diffusion in relatively dry soil: IV. Temperature and pressure effects on sorption diffusion coefficients, Soil Sci. Soc. Am. Proc., 29(2), 144-148, 1965.

_____, Temperature and soil-water diffusivity relations, Soil Sci. Soc. Am. Proc., 27(4), 363-366, 1963.

_____, On the calculation of hydraulic conductivity, Soil Sci. Soc. Am. Proc., 36(2), 380-382, 1972.

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

Jury, W. A. and E. E. Miller, Measurement of the transport coefficients for coupled flow of heat and moisture in a medium sand, Soil Sci. Soc. Am. Proc., 38(4), 551-557, 1974.

Kijne, J. W. and S. A. Taylor, The temperature dependence of soil water vapor pressure, Soil Sci. Soc. Am. Proc., 28(5), 595-599, 1964.

Kimball, B. A., R. D. Jackson, R. J. Reginato, F. S. Nakayama,and S. B. Idso, Comparison of field-measured and calculated soil-heat fluxes, Soil Sci. Soc. Am. Proc., 40(1), 18-25, 1976.

King, L. G., Description of soil characteristics for partially saturated flow, Soil Sci. Soc. Am. Proc., 29(4), 359-362, 1965.

Klute, A., Some theoretical aspects of the flow of water in unsaturated soils, Soil Sci. Soc. Am. Proc., 16(2), 144-148, 1952.

Kunze, R. J., G. Uehara, and K. Graham, Factors important in the calculation of hydraulic conductivity, Soil Sci. Soc. Am. Proc., 32(6), 760-765, 1968.

Lai, S., J. M. Tiedje, and A. E. Erickson, In situ measurement of gas diffusion coefficient in soils, Soil Sci. Soc. Am. Proc., 40(1), 3-6, 1976.

Marshall, T. J., A relation between permeability and size distribution of pores, J. Soil Sci., 9(1), 1-8, 1958.

Mercer, J. W. and C. R. Faust, The application of finite-element techniques to immiscible flow in porous media, in International Conference on Finite Elements in Water Resources, Part II, Water Resources Program, Princeton University, 1976.

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

Miller, E. E. and R. D. Miller, Theory of capillary flow: I. Practical implications, Soil Sci. Soc. Am. Proc., 19(3), 267-271, 1955.

_____, Physical theory for capillary flow phenomena, J. Appl. Phys., 27(4), 324-332, 1956.

Miller, R. J. and P. F. Low, Threshold gradient for water flow in clay systems, Soil Sci. Soc. Am. Proc., 27(6), 605-609, 1963.

Millington, R. J., and J. P. Quirk, Permeability of porous media, Nature, 183, 387-388, 1959.

Milly, P. C. D., Comment on "Analysis of water and heat flow in unsaturated - saturated porous media" by M. Sophocleous, (unpublished manuscript), M. I. T., Cambridge, Massachusetts, 1979.

Milly, P. C. D., Documentation of a numerical model for the simulation of moisture and heat transport in soils, M. I. T. Technical Report in preparation, Cambridge, 1980.

Mualem, Y., Modified approach to capillary hysteresis based on a similarity hypothesis, Water Resour. Res., 9(5), 1324-1331, 1973.

_____, A conceptual model of hysteresis, Water Resour. Res., 10(3), 514-520, 1974.

_____, A new model for predicting the hydraulic conductivity of unsaturated porous media, Water Resour. Res., 12(3), 513-522, 1976a.

_____, Hysteretical models for prediction of the hydraulic conductivity of unsaturated porous media, Water Resour. Res., 12(6), 1248-1254, 1976b.

_____, Extension of the similarity hypothesis used for modeling the soil water characteristics, Water Resour. Res., 13(4), 773-780, 1977.

Mualem, Y. and G. Dagan, A dependent domain model of capillary hysteresis, Water Resour. Res., 11(3), 452-460, 1975.

_____, Hydraulic conductivity: Unified approach to the statistical models, Soil Sci. Soc. Am. Proc., 42(3), 392-395, 1978.

Narasimhan, T. N. and P. A. Witherspoon, Numerical model for saturated - unsaturated flow in deformable porous media I. Theory, Water Resour. Res., 13(3), 657-664, 1977.

_____, Numerical model for saturated - unsaturated flow in deformable porous media 3. Applications, Water Resour. Res., 14(6), 1017-1034, 1978.

Neuman, S. P., R. A. Feddes and E. Bresler, Finite element analysis of two-dimensional flow in soils considering water uptake by roots, I. Theory, Soil Sci. Soc. Am. Proc., 39(2), 224-230, 1975.

Nielsen, D. R. and J. W. Biggar, Measuring capillary conductivity, Soil Sci., 92(3), 192-193, 1961.

Philip, J. R., The theory of infiltration: 1. The infiltration equation and its solution, Soil Sci., 83(5), 345-357, 1957.

_____, The theory of infiltration: 6. Effect of water depth over soil, Soil Sci., 85(5), 278-286, 1958.

_____, Energy dissipation during absorption and infiltration: I, Soil Sci., 89(3), 132-136, 1960.

_____, Similarity hypothesis for capillary hysteresis in porous materials, J. Geophys. Res., 69(8), 1553-1562, 1964.

Philip, J. R. and D. A. de Vries, Moisture movement in porous materials under temperature gradients, Trans. Amer. Geophys. Union, 38(2), 222-232, 1957.

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

Poulovassilis, A., Hysteresis of pore water, an application of the concept of independent domains, Soil Sci., 93(6), 405-412, 1962.

Rogers, J. S. and A. Klute, The hydraulic conductivity - water content relationship during nonsteady flow through a sand column, Soil Sci. Soc. Am. Proc., 35(5), 695-700, 1971.

Rose, D. A., Water movement in porous materials: Part 2 - The separation of the components of water movement, Brit. J. Appl. Phys., 14, 491-496, 1963.

Rubin, J., Numerical method for analyzing hysteresis-affected, post-infiltration redistribution of soil moisture, Soil Sci. Soc. Am. Proc., 31(1), 13-20, 1967.

_____, Theoretical analysis of two-dimensional, transient flow of water in unsaturated and partly unsaturated soils, Soil Sci. Soc. Am. Proc., 32(5), 607-615, 1968.

Sasamori, T., A numerical study of atmospheric and soil boundary layers, J. Atmos. Sci., 27(8), 1122-1137, 1970.

Sepaskhah, A. R. and L. Boersma, Thermal conductivity of soils as a function of temperature and water content, Soil Sci. Soc. Am. Proc., 43(3), 439-444, 1979.

Sophocleous, M., Analysis of water and heat flow in unsaturated - saturated porous media, Water Resour. Res., 15(5), 1195-1206, 1979.

Staple, W. J., Comparison of computed and measured moisture redistribution following infiltration, Soil Sci. Soc. Am. Proc., 33(6), 840-847, 1969.

Swartzendruder, D., The flow of water in unsaturated soils, in Flow Through Porous Media, ed. R. J. M. De Wiest, Academic Press, New York, 1969.

Talsma, T., Hysteresis in two sands and the independent domain model, Water Resour. Res., 6(3), 964-970, 1970.

Taylor, S. A., The activity of water in soils, Soil Sci., 86(2), 83-90, 1958.

Taylor, S. A. and G. L. Stewart, Some thermodynamic properties of soil water, Soil Sci. Soc. Am. Proc., 24(4), 243-247, 1960.

Topp, G. C. and E. E. Miller, Hysteretic moisture characteristics and hydraulic conductivities for glass-bead media, Soil Sci. Soc. Am Proc., 30(2), 156-162, 1966.

Vauclin, M., G. Hamon, and G. Vachaud, Simulation of coupled flow of heat and water in a partially saturated soil. Determination of the surface temperature and evaporation rate from a bare soil, in Influence des Gradients Thermiques sur les Transferts d'Humidité dans la Zone non Saturée, A. T. P. Hydrogeologie 1652, Centre National de la Recherche Scientifique, Grenoble, 1977.

de Vries, D. A., Simultaneous transfer of heat and moisture in porous media, Trans. Amer. Geophys. Union, 39(5), 909 - 916, 1958.

_____, Thermal properties of soils, in Physics of Plant Environment, ed. W. R. Van Wijk, North-Holland Publ. Co., Amsterdam, 1966.

Whisler, F. D. and A. Klute, The numerical analysis of infiltration, considering hysteresis, into a vertical soil column at equilibrium under gravity, Soil Sci. Soc. Am. Proc., 29(5), 1965.

Wierenga, P. J., D. R. Nielsen, and R. M. Hagan, Thermal properties of a soil based upon field and laboratory measurements, Soil Sci. Soc. Am. Proc., 33(3), 354-360, 1969.

Youngs, E. G., An infiltration method of measuring the hydraulic conductivity of unsaturated porous materials, Soil Sci., 97(5), 307-311, 1964.

APPENDIX A

DOCUMENTATION OF SPLaSHWaTr1

A.1 Introduction and Contents

This appendix provides documentation of the computer code for execution of the numerical algorithm outlined in Chapter 4. The computer program is named "SPLaSHWaTr1", an acronym for "Simulation Program for Land Surface Heat and Water Transport, Edition 1."

We first present the large-scale organization of SPLaSHWaTr1, which is defined roughly by the main program of the code. The concept of a simulation period is introduced as a fundamental element in the main program.

In order that various computational tasks be executed, the main program calls many subroutines. Ideally, each subroutine has a well-defined role in the algorithm. The purpose of each subroutine is described in Section A.3.

Execution of SPLaSHWaTr1 requires the definition of an input file containing all of the system parameters, initial conditions and boundary conditions. The format of the input file is presented in Section A.4.

Finally, some listings of sample input and output files are given. Such examples can be helpful when a user must debug the program, possibly after a change to a new computer system or when modifications of the code have been made. These listings constitute Section A.5.

An extensive list of definitions of the FORTRAN variables used in SPLaSHWaTrl is given in the main program listing (Appendix B). For the user's convenience, much of the documentation contained in this appendix is also found in the comment statements of the program listing.

A.2 Structure of SPLaSHWaTrl - An Overview

A.2.1 The Simulation Period

The fundamental time period for specification of boundary conditions in a simulation is the "simulation period." By definition, a simulation period is a time interval during which the specified surface boundary conditions are constant. In the current version, the user may specify either the average mass flux rate or the matric potential at the surface during the simulation period. Similarly, the other condition may be expressed either as a given heat flux or as a known temperature.

In general, a simulation period will consist of more than one time step. The time step size is controlled automatically by the program. Considerations of "accuracy" and numerical "convergence," as defined in Chapter 5, limit the size of the time step.

A.2.2 Flow of Control in the Main Program

The structure of the main program of SPLaSHWaTrl is illustrated in Fig. A.1. After initial input and computations, the specified simulation periods are modeled sequentially by looping through the remainder of the program, once for each period. Inside a given simulation period, a time step loop is executed as many times as necessary to reach the end

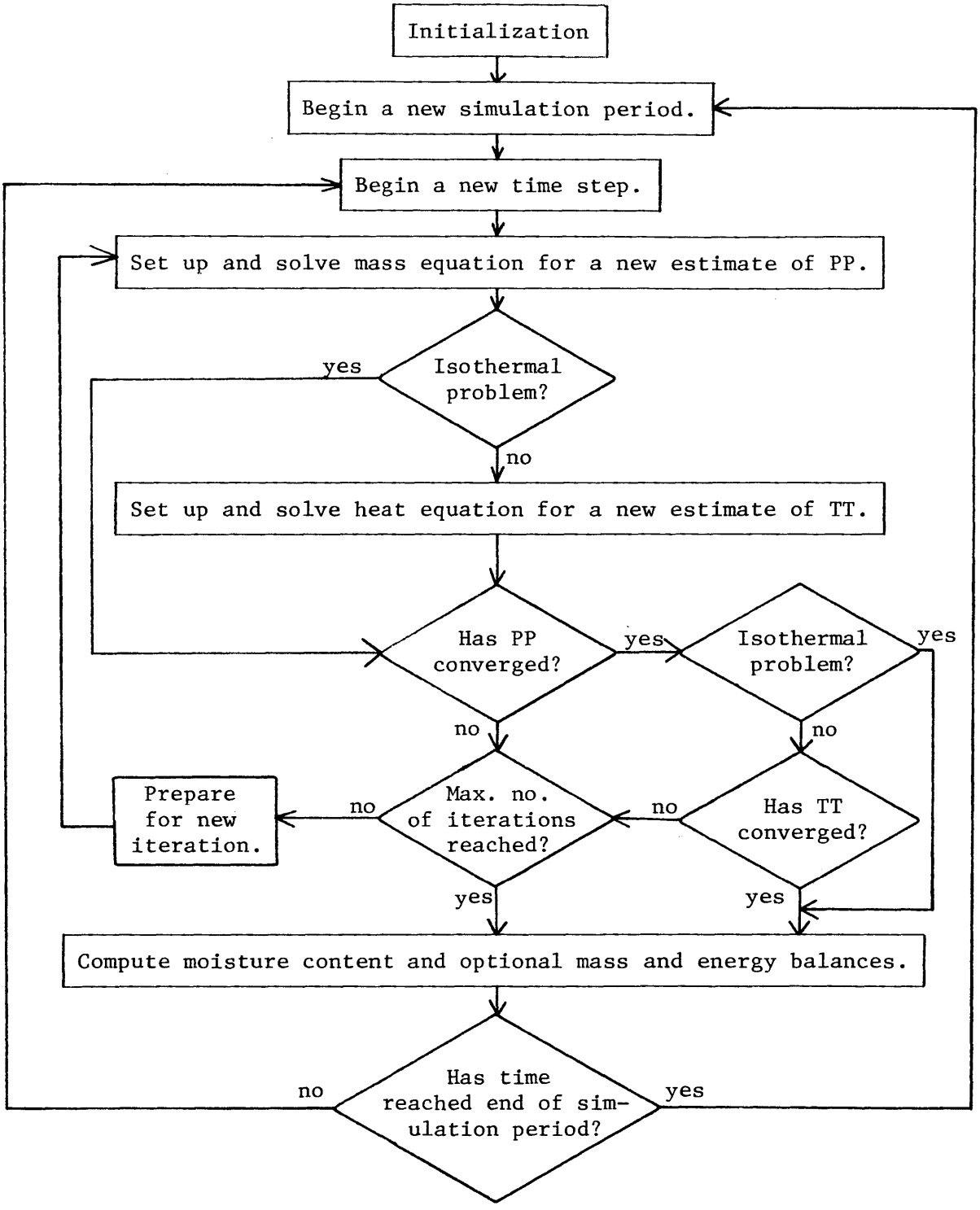


Figure A.1. Flowchart for SPLaSHWaTr1 Main Program.

of the period, given the constraints on the time step size.

For a given time step (Section 4.4), the mass and energy equations are solved alternately to obtain successive estimates of matric potential and temperature at the end of the current time step. For isothermal problems of moisture transport, the heat equation is omitted.

When convergence of the calculated solution is obtained (or if a specified maximum number of iterations is exceeded), the new moisture contents are calculated and mass and energy balances are calculated, if desired. If the time has reached the end of the current simulation period, a new period is begun. Otherwise, a new time step follows.

Further details of the operation of the main program may be found in the program listing. Documentation of the various subprograms is presented in the following section and in the program listing.

A.3 SPLaSHWaTr1 Subprograms

A.3.1 INIT2

This subroutine is called only once, at the start of execution, in order to perform the initial calculations and input and output operations. Specifically, it performs the following operations:

1. Set the values of constants.
2. Read the option codes, simulation parameters, initial conditions, element information, etc.
3. Write initial output as requested.
4. Call SOILI2 to get soil parameters.
5. Initialize PP, TT, X, XOLD; initialize mass and energy

balance variables, time variables.

A.3.2 NEWPR2

This subroutine begins a new simulation period by reading the duration and the boundary conditions. Optionally, an initial time step length for the period may be specified (see Section A.3.3). If NBCP (NBCT) equals one, a specified matric head (temperature) equal to BCP (BCT) is applied as the surface boundary condition. If NBCP (NBCT) equals two, a surface mass (heat) flux of BCP (BCT) is indicated.

Program execution is terminated in this subroutine when a negative duration is specified for a simulation period.

The simulation period number, NPER, is incremented during each call, and the time at the end of the period, TEND, is computed. The time step counter is set to zero.

A.3.3 NWSTP2

This subroutine is called by the main program to begin a new time step. It computes a new time step length, updates the wetting history, computes estimates (by extrapolation) of the solution for the new time step, and updates various vectors. ENTRY NEWIT2, contained in this subprogram, is called at the start of subsequent iterations in the same time step. It increments KIT, the iteration counter, and updates PPOLD and TTOLD.

The length of the first time step in a simulation period may be specified as DLT in input to NEWPR2 (Section A.3.2). For other time steps, or if DLT is less than zero, the new value is calculated by an

algorithm that tends to maintain the number of iterations per time step near a specified value, ITDES. If KIT from the last time step exceeds ITDES, the value of DELT is reduced by a factor of the square of their ratio. If KIT is less than ITDES, the time step length is increased by a factor of 1.2. If necessary, the value of DELT is reduced to prevent the time step from overshooting the end of the simulation period.

A.3.4 PPRAM2

For both nodes of each element, PPRAM2 computes the values of the coefficients and the gravity term in the mass conservation equation. This will allow the various discontinuities that may occur across element boundaries. The components of the coefficients that depend only on ψ and T are continuous at node points and may therefore be calculated using the global node index; these computations are inside the first DO loop. In the second loop, the mass equation coefficients are calculated using the element and local node indices.

The evaluation of $\frac{d\theta}{d\Psi}$ may be performed in two ways. If J9 has been given as zero, it is simply computed as the derivative of the current scanning curve, evaluated at the end of the time step (using the latest estimates of ψ and T). If J9 is equal to one and a significant change in ψ occurs, the iterative method described in Section 4.7 is employed.

If the isothermal liquid flow option is invoked (I24 = 3), many unnecessary computations in PPRAM2 are skipped.

A.3.5 TPRAM2

This subroutine calculates the coefficients of the heat equation at both nodes of each element. It is analogous to PPRAM2.

Note that some variables calculated in PPRAM2 (e.g., RHOV) are saved in storage for use in TPRAM2 so that they need not be recalculated.

A.3.6 MAT2

MAT2 assembles the global coefficient matrices of the Galerkin expressions for the mass or energy conservation equations.

Three weighting options are provided for evaluation of the storage matrices. If J3 (J4) is equal to zero, the mass (energy) equation is evaluated according to the expression generated by the Galerkin method (e.g., the matrix two lines above (4.35)). When J3 (J4) is equal to two, the lumping scheme of (4.35) is used. For J3 (J4) equal to one, the following expression is employed:

$$\begin{vmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{vmatrix} = \Delta \begin{vmatrix} \frac{C_{11}}{3} + \frac{C_{12}}{6} & 0 \\ 0 & \frac{C_{11}}{6} + \frac{C_{12}}{3} \end{vmatrix}$$

Element conductivity terms are evaluated using the functional coefficient scheme (which results in an arithmetic average) when J8 is equal to one. A geometric mean is used for J8 equal to two.

A.3.7 EQN2

This subroutine performs the finite difference of the time derivatives and sets up the matrix equation for the matrix solver (Section 4.3). Note that the ordering of the arguments in the CALL statements in the main program depends on which equation is being solved. The variable names used in EQN2, therefore, do not always correspond to those in the

COMMON block, their definition being determined by the CALL statement.

The matrix "SAVE" retains the lines of the matrix equation corresponding to the end nodes for later calculation of the end fluxes.

A.3.8 PBC2

This subroutine adjusts the matrix equation to account for boundary conditions. (The boundary conditions are specified by input in NEWPR2.) PBC2 incorporates the mass boundary condition, and ENTRY TBC2 applies the heat boundary condition. The method for application of boundary conditions is described in Section 4.2.4.

A type I boundary condition (specified ψ or T) at node i is incorporated as follows:

1. Set i'th element of right-hand side vector equal to known value of state variable.
2. Set i'th diagonal element of matrix equal to unity. Other elements in i'th row of matrix set to zero.
3. Use known value of i'th state to evaluate any terms involving it in other lines of the matrix equation. Move these known terms to the right-hand-side vector, and replace the relevant matrix elements with zero. In this way, the matrix will remain symmetric.

A flux boundary condition is incorporated simply by adding to or subtracting from the right-hand-side vector the specified flux into the system at the relevant node. See, for example, Equation (4.29).

In the current version of SPLaSHWaTr1, a first-type B.C. is applied to matric head at the bottom of the column by allowing no change in the initial condition. Heat flux at the bottom of the column is zero except for advection by moisture flow; there is no temperature gradient.

The surface boundary conditions are applied according to the rules discussed in Section A.3.2.

A.3.9 SOLVE2

The Thomas algorithm for solution of a tridiagonal matrix equation is used to solve the mass or energy equation, yielding the values of matric head or temperature at each node. The algorithm is presented by Pinder and Gray (p. 23, 1977).

Note that the second argument in SOLVE2 contains the right-hand-side of the equation when the subroutine is called, while the subroutine returns the solution in the same vector.

A.3.10 CHK2

This subroutine checks for convergence of the mass (K = 1) or energy (K = 2) solution. It returns a value of J = 1 if convergence is obtained and J = 0 otherwise.

For K = 1, convergence is obtained when the relative difference between PP and PPOLD,

$$\frac{PP(I) - PPOLD(I)}{PP(I)}$$

is less than PERR for all I. For K = 2, the criterion is that TT and TTOLD differ by no more than TERR for all I.

A.3.11 RHOZZ2

This subprogram defines three functions of temperature - saturation vapor density and its derivative and the correction of hydraulic conductivity for temperature effects.

The vector RZ contains the saturation vapor density values for integer values of temperature from 0°C to 80°C. Actual values for any T are found by linear interpolation. The slope, DRZDT, is estimated as the difference between the two closest values of RZ tabulated (since the corresponding difference in T is unity).

The temperature correction for hydraulic conductivity is

$$\text{CKTT2} = \frac{\mu(T_o)}{\mu(T)} = (\text{CKTN} \cdot \text{VISCC})^{-1}$$

The value of CKTN is calculated by an initial call to this function from INIT2 (with CKTN = 1). The viscosity is calculated by one of two formulas depending on whether T is greater than or less than 20°C (CRC Handbook of Chemistry and Physics, 1979, p. F-51).

A.3.12 BAL2

This subroutine need not be called to complete the execution of any simulation. If desired, it is called after each time step to produce an accounting of the mass and energy budget for the simulation. The total storage of mass and energy is calculated, assuming linear variation of storage inside elements. Changes in storage for the most recent time step and for the entire simulation are compared to the corresponding net fluxes, and errors are computed. All balance information is printed out.

A.3.13 SOILI2

This subroutine performs calculations involving the hydraulic properties of soils. It uses the models of moisture retention and hydraulic conductivity presented in Chapter 3. Alternatively, the user may specify J5 = 1 and supply his own FORTRAN statements at the appropriate lines in the code; in this case, hysteresis is not allowed.

The first entry point, SOILI2, is used in a one-time call from INIT2 to initialize various soil variables. For each soil type, a set of parameters that determine its hydraulic behavior is read. The parameters of Eq. (3.8) are derived from the input variables by a fitting procedure.

The input variables are θ_u, the negative of the wetting equivalent of the bubbling potential, the moisture content at the wilting point, the parameter M, the PF intercept of the capillary segment of the main wetting curve, and the saturated hydraulic conductivity at reference temperature T_o. The PF-intercept of the adsorption segment of the main wetting curve is taken as 7.0, and the PF at the wilting point is 4.2. PFK, the maximum value of PF at which the liquid phase is discontinuous, is set equal to 5.85.

Given the input parameters, SOILI2 makes initial estimates of a_1, a_2, s_1, and s_2, assuming that M is infinite (Figs. 3.5 and 3.6). The parameters a_1 and a_2 are then adjusted iteratively to fit the main wetting curve at saturation and at the wilting point, using the finite value of M.

In order that the relative hydraulic conductivity integral (3.38) may be evaluated, the main wetting curve is approximated by a

piecewise linear relation between θ_w and PF. Thus,

$$PF = PFR_{j,k} - SS_{j,k}(\theta_w - XR_{j,k}) \quad \begin{cases} XR_{j,k} < \theta_w \leq XR_{j,k+1} \\ PFR_{j,k+1} \leq PF < PFR_{j,k} \end{cases} \quad k = 1, 2, \ldots, 8$$

for soil type j, where

$$\left.\begin{aligned} \theta_w(PFR_{j,k}) &= XR_{j,k} \\ SS_{j,k} &= \frac{PFR_{j,k} - PFR_{j,k+1}}{XR_{j,k+1} - XR_{j,k}} \end{aligned}\right\} \quad k = 1, 2, \ldots, 9$$

The $PFR_{j,k}$ and $XR_{j,k}$ are chosen such that

$$R_k = \frac{a_{1j} - s_{1j}\, PFR_{j,k+1}}{\theta_w(PFR_{j,k+1})} \quad k = 1, 2, \ldots, 7$$

in which a_{1j} and s_{1j} are the soil parameters a_1 and s_1 for soil type j. The values of R_k are 0.01, 0.1, 0.3, 0.5, 0.7, 0.9, and 0.99 for k = 1, 2, ..., 7, respectively. In addition,

$$XR_{j,1} = \theta_{kj} \qquad PFR_{j,1} = PFK$$

$$XR_{j,9} = \theta_{uj} \qquad PFR_{j,9} = \log_{10}(PB_j)$$

where θ_{kj}, θ_{uj} and PB_j are θ_k, θ_u and PB for soil j.

Using this representation for $\Psi(\theta_w)$, we may evaluate the integrals appearing in (3.38). In general,

$$\int_o^S \frac{dS'}{\psi(S')} = \frac{-1}{\theta_u - \theta_k} \int_{\theta_k}^{\theta_w} 10^{-PF} d\theta$$

Substituting our piecewise expression for PF and integrating, we obtain

$$\int_0^S \frac{dS'}{\psi(S')} = [(\theta_u - \theta_k)\ \ln\ 10]^{-1} \left\{ \sum_{i=1}^{m-1} SS_{j,i}^{-1} (PR_{j,i+1}^{-1} - PR_{j,i}^{-1}) + SS_{j,m}^{-1} (PWET^{-1} - PR_{j,m}^{-1}) \right\} \qquad PR_{j,m+1} \leq PWET < PR_{j,m}$$

in which

$$PR_{j,i} = -\ 10^{PFR_{j,i}}$$

and

$$PWET = \theta_w^{-1}(\theta)$$

Defining

$$ENT_{j,m} = \sum_{i=1}^{m} SS_{j,i}^{-1} (PR_{j,i+1}^{-1} - PR_{j,i}^{-1})$$

we then have, from (3.38)

$$K_r(\theta) = S_e^{1/2} \{ [ENT_{j,m-1} + SS_{j,m}^{-1} (PWET^{-1} - PR_{j,m}^{-1})]\ ENT_{j,8}^{-1} \}^2$$

$$PR_{j,m+1} \leq PWET < PR_{j,m}$$

The quantities PR, ENT, etc., are calculated, as outlined above, by a call to SOILI2 before the actual start of the simulation. Later calls to ENTRY SOIL32 are used to calculate hydraulic conductivity during a simulation (see below).

ENTRY SOIL12 is called (for J5 not equal to 1) at the beginning of a new time step. It updates the wetting history, if necessary, of any element. If the change in average moisture content during the most recent

time step was in the opposite direction of that during the previous time step, the history is updated, following the procedure described in Section 3.3.

A call to ENTRY SOIL22 will calculate moisture content and, if KODE equals 2, the slope of the current scanning curve. Equations (3.36) and (3.37), along with (3.8), are used for this purpose. They may be differentiated, using (3.34) and (3.35), to obtain the formulas for the slopes. When J5 equals unity, hysteresis is ignored, and the moisture characteristic and its slope are defined by user-supplied FORTRAN statements.

ENTRY SOIL32 calculates the relative hydraulic conductivity, given the moisture content, using the formula given above. It first determines in which interval the moisture content falls, then uses a modified Newton-Raphson algorithm to find the corresponding value of PF and, hence, PWET. The formula is then used. If J5 is equal to one, a user-supplied FORTRAN statement is used to express K_r as a function either of θ or of ψ (or Ψ).

The liquid island enhancement factor for vapor flow is calculated in SOIL32 using Eq. (2.13).

A.3.14 XWET

XWET is a function subprogram that computes the moisture content of the main wetting curve as a function of PF and the soil index. It uses Eq. (3.38).

A.4 Input Data for SPLaSHWaTr1

The input data required for running the current version of the numerical simulation model are listed in Table A.1. Definitions of most of the FORTRAN variables are given in the comment statements at the beginning of the main program (Appendix B). The remaining variables, which are all in record type 6, are defined in subroutine SOILI2.

The number of type-7 records is one more than the desired number of simulation periods. In the last record, a negative value for DURTN should be entered; this will terminate execution. (The other variables in that record will then be ignored.)

Subroutine in which READ occurs	Record type	No. of records	Format	Variable names
INIT2	1	1	25I2	I1-I25
INIT2	2	1	9I2	J1-J9
INIT2	3	1	6G10.0	NN, NS, PERR, TERR, TZERO, ITDES
INIT2	4	NN	3G10.0, 50X	Z(I), P(I), T(I)
INIT2	5	NL	4G10.0, 40X	IS(L), IH(L),XWRE(L,1), XWRE(L,2)
SOILI2	6	NS	6G10.0, 20X	J, POR(J), PB(J), XWILT(J), EM(J), CKSAT(J), PFINT1(J)
NEWPR2	7	see text	6G10.0	DURTN, DLT, NBCP, BCP, NBCT, BCT

TABLE A.1. Input Format for SPLaSHWaTr1

A.5 Two Examples: Input and Output

A.5.1 Example 1

The first example given here is the problem of coupled heat and vapor diffusion examined in Section 5.5. The input and output are given for two 500-second simulation periods. Note that, although the boundary conditions do not change, it is convenient to use separate simulation periods in order to obtain output at desired times.

Example 1 Input

```
1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 8 0 0
0 0 2 2 0 0 0 0 1
       11         1   1.0D-03   1.0D-05      20.0        5
      -5. -1379139.       20.
     -4.5 -1379139.       20.
     -4.0 -1379139.       20.
     -3.5 -1379139.       20.
     -3.0 -1379139.       20.
     -2.5 -1379139.       20.
     -2.0 -1379139.       20.
     -1.5 -1379139.       20.
     -1.0 -1379139.       20.
     -0.5 -1379139.       20.
     -0.0 -1379139.       20.
        1         1       0.0
        1         1       0.0
        1         1       0.0
        1         1       0.0
        1         1       0.0
        1         1       0.0
        1         1       0.0
        1         1       0.0
        1         1       0.0
        1         1       0.0
    0.500       10.      .056      80.0     0.001       4.
   5.D+02   5.D-01        1-1241223.3         1     20.0
   5.D+02       -1.       1-1241223.3         1     20.0
     -1.0
```

OUTPUT OPTIONS -- 1=ACTION 0=NO ACTION
****

OUTPUT OPTIONS	I1	1
SIMULATION OPTIONS	I2	1
OTHER SIMULATION SCALARS	I3	1
NODE AND ELEMENT INFORMATION	I4	1
INPUT SOIL INFORMATION	I5	1
COMPUTED SOIL PARAMETERS	I6	1
SIMULATION PERIOD PARAMETERS	I7	1
MATRIC POTENTIAL	I8	0
-1 - NEVER		
0 - EVERY SIMULATION PERIOD		
1 - EVERY TIME STEP		
2 - EVERY ITERATION		
TEMPERATURE SOLUTION	I9	0
(SAME CODE AS FOR I8)		
MOISTURE CONTENT	I10	0
-1 - NEVER		
0 - EVERY SIMULATION PERIOD		
1 - EVERY TIME STEP		
MASS BALANCE INFORMATION	I11	0
(SAME CODE AS FOR I10)		
ENERGY BALANCE INFORMATION	I12	0
(SAME CODE AS FOR I10)		
NOT IN USE	I13	0
NOT IN USE	I14	0
NOT IN USE	I15	0
NOT IN USE	I16	0
MOISTURE EQUATION COEFFICIENTS	I17	0
MOISTURE EQUATION MATRICES	I18	0
MOISTURE EQUATION	I19	0
TEMPERATURE EQUATION COEFFICIENTS	I20	0
TEMPERATURE EQUATION MATRICES	I21	0
TEMPERATURE EQUATION	I22	0
WRITE SOLUTION ON FILE I23 EACH PERIOD	I23	8
IF EQUAL TO 3, USE ISOTHERMAL EQUATION	I24	0
NOT IN USE	I25	0

SIMULATION OPTIONS
****

NOT IN USE	J1	0
NOT IN USE	J2	0
MASS LUMPING OPTION	J3	2
ENERGY LUMPING OPTION	J4	2
USE ALTERNATE RETENTION CURVE	J5	0
NOT IN USE	J6	0

NOT IN USE	J7	0
NOT IN USE	J8	0
ITERATIVE MOISTURE CAPACITY SCHEME	J9	1

OTHER SIMULATION SCALARS
========================

NUMBER OF NODES	11	
NUMBER OF ELEMENTS	10	
NUMBER OF SOIL TYPES	1	
MOISTURE CONVERGENCE CRITERION	0.1D-02	
TEMP. CONVERGENCE CRITERION	0.1D-04	DEG CELSIUS
REFERENCE TEMPERATURE	20.	DEG CELSIUS
DESIRED NO. OF ITERATIONS	5	

NODAL DATA
==========

NODE	Z	P	T
1	-5.00	-1379139.	20.0
2	-4.50	-1379139.	20.0
3	-4.00	-1379139.	20.0
4	-3.50	-1379139.	20.0
5	-3.00	-1379139.	20.0
6	-2.50	-1379139.	20.0
7	-2.00	-1379139.	20.0
8	-1.50	-1379139.	20.0
9	-1.00	-1379139.	20.0
10	-0.50	-1379139.	20.0
11	0.0	-1379139.	20.0

INPUT SOIL PARAMETERS
=====================

SOIL TYPE	POR	PB	XWILT	EM	CKSAT	PFINT1
1	0.50	10.	0.06	80.	0.10D-02	4.0000

COMPUTED SOIL PARAMETERS

SOIL TYPE	A1	A2	S1	S2	XK
1	0.66667	0.13999	0.16667	0.02000	0.02299

SOIL TYPE 1

XR	0.23D-01	0.55D-01	0.59D-01	0.61D-01	0.62D-01	0.62D-01	0.63D-01	0.63D-01	0.50D+00
PFR	0.59D+01	0.42D+01	0.41D+01	0.40D+01	0.39D+01	0.39D+01	0.39D+01	0.39D+01	0.10D+01
PR	-0.71D+06	-0.17D+05	-0.11D+05	-0.94D+04	-0.86D+04	-0.81D+04	-0.77D+04	-0.76D+04	-0.10D+02
SS	0.50D+02	0.49D+02	0.47D+02	0.45D+02	0.43D+02	0.42D+02	0.41D+02	0.66D+01	
ENT	-0.11D-05	-0.17D-05	-0.21D-05	-0.24D-05	-0.25D-05	-0.27D-05	-0.27D-05	-0.15D-01	

ELEMENT DATA

ELEMENT	IS	IH	X(L,1)	X(L,2)	XWRE(L,1)	XWRE(L,2)	XOLD
1	1	1	0.0172	0.0172	0.0	0.0	0.0172
2	1	1	0.0172	0.0172	0.0	0.0	0.0172
3	1	1	0.0172	0.0172	0.0	0.0	0.0172
4	1	1	0.0172	0.0172	0.0	0.0	0.0172
5	1	1	0.0172	0.0172	0.0	0.0	0.0172
6	1	1	0.0172	0.0172	0.0	0.0	0.0172
7	1	1	0.0172	0.0172	0.0	0.0	0.0172
8	1	1	0.0172	0.0172	0.0	0.0	0.0172
9	1	1	0.0172	0.0172	0.0	0.0	0.0172
10	1	1	0.0172	0.0172	0.0	0.0	0.0172

****

SIMULATION PERIOD NUMBER 1

DURATION	500.00	SECONDS
START TIME	0.0	SECONDS
END TIME	0.500D+03	SECONDS
NBCP	1	
BCP	-0.12D+07	
NBCT	1	
BCT	0.20D+02	

MASS BALANCE INFORMATION
=======================

DATA FOR CURRENT TIME STEP

SURFACE FLUX RATE	-0.95175D-07 CM/S
BOTTOM FLUX RATE	-0.48668D-11 CM/S
NET FLUX RATE	0.95170D-07 CM/S
RATE OF STORAGE CHANGE	0.95121D-07 CM/S
PERCENT ERROR	-0.05124

CUMULATIVE DATA

NET TOTAL FLUX	0.27902D-03 CM
TOTAL STORAGE CHANGE	0.27895D-03 CM
PERCENT ERROR	-0.02281

ENERGY BALANCE INFORMATION
==========================

DATA FOR CURRENT TIME STEP

SURFACE FLUX RATE	-0.11401D-04 CAL/CM2-S
BOTTOM FLUX RATE	-0.36776D-14 CAL/CM2-S
NET FLUX RATE	0.11401D-04 CAL/CM2-S
RATE OF STORAGE CHANGE	0.11401D-04 CAL/CM2-S
PERCENT ERROR	-0.00002

CUMULATIVE DATA

NET TOTAL FLUX	0.97591D-02 CAL/CM2
TOTAL STORAGE CHANGE	0.97567D-02 CAL/CM2
PERCENT ERROR	-0.02430

MATRIC HEAD

1	-1379139.00	2	-1379138.37	3	-1379138.14	4	-1379137.77	5	-1379137.20
6	-1379136.35	7	-1379133.94	8	-1379095.30	9	-1378251.36	10	-1364372.29
11	-1241223.30								

TEMPERATURE

1	20.00075022518	2	20.00083794392	3	20.00111127388	4	20.00159871587	5	20.00234237771
6	20.00338896120	7	20.00477655602	8	20.00651587249	9	20.00851060970	10	20.00955013690
11	20.00000000000								

MOISTURE CONTENT

1	1	0.017198004	1	2	0.017198009	2	1	0.017198009	2	2	0.017198015	3	1	0.017198015
3	2	0.017198026	4	1	0.017198026	4	2	0.017198043	5	1	0.017198043	5	2	0.017198066
6	1	0.017198066	6	2	0.017198105	7	1	0.017198105	7	2	0.017198379	8	1	0.017198379
8	2	0.017203731	9	1	0.017203731	9	2	0.017291659	10	1	0.017291659	10	2	0.018113153

****

SIMULATION PERIOD NUMBER 2

DURATION	500.00	SECONDS
START TIME	0.500D+03	SECONDS
END TIME	0.100D+04	SECONDS
NBCP	1	
BCP	-0.12D+07	
NBCT	1	
BCT	0.20D+02	

MASS BALANCE INFORMATION

DATA FOR CURRENT TIME STEP

SURFACE FLUX RATE	-0.86276D-07	CM/S
BOTTOM FLUX RATE	-0.71493D-11	CM/S
NET FLUX RATE	0.86269D-07	CM/S
RATE OF STORAGE CHANGE	0.86258D-07	CM/S
PERCENT ERROR	-0.01267	

CUMULATIVE DATA

NET TOTAL FLUX 0.32407D-03 CM
TOTAL STORAGE CHANGE 0.32401D-03 CM

PERCENT ERROR -0.02076

ENERGY BALANCE INFORMATION
==========

DATA FOR CURRENT TIME STEP

SURFACE FLUX RATE -0.76971D-05 CAL/CM2-S
BOTTOM FLUX RATE -0.21108D-13 CAL/CM2-S
NET FLUX RATE 0.76971D-05 CAL/CM2-S
RATE OF STORAGE CHANGE 0.76971D-05 CAL/CM2-S

PERCENT ERROR -0.00004

CUMULATIVE DATA

NET TOTAL FLUX 0.14294D-01 CAL/CM2
TOTAL STORAGE CHANGE 0.14292D-01 CAL/CM2

PERCENT ERROR -0.01661

MATRIC HEAD

1	-1379139.00	2	-1379136.77	3	-1379136.35	4	-1379135.91	5	-1379135.28
6	-1379133.77	7	-1379118.83	8	-1378898.38	9	-1376189.12	10	-1352587.32
11	-1241223.30								

TEMPERATURE

1	20.00295556862	2	20.00307317122	3	20.00342594424	4	20.00401146737	5	20.00482450747
6	20.00585498169	7	20.00708498899	8	20.00847181009	9	20.00977018176	10	20.00921351084
11	20.00000000000								

MOISTURE CONTENT

1	1	0.017198042	1	2	0.017198058	2	1	0.017198058	2	2	0.017198067	3	1	0.017198067

3	2	0.017198080	4	1	0.017198080	4	2	0.017198098	5	1	0.017198098	5	2	0.017198125
6	1	0.017198125	6	2	0.017198241	7	1	0.017198241	7	2	0.017199653	8	1	0.017199653
8	2	0.017216759	9	1	0.017216759	9	2	0.017367005	10	1	0.017367005	10	2	0.018113153

A.5.2 Example 2

The second example simulates isothermal infiltration into Yolo light clay using (5.1) and (5.2). This is the problem whose solution is plotted in Figure 5.3. Simulation periods ending at 10^3 sec. and 10^4 sec. are used. The computed soil parameters, although included in the output, are irrelevant, as the hydraulic functions were specified explicitly in SOILI2.

Example 2 Input

```
1 1 1 1 1 1 1 0-1 0 0-1 0 0 0 0 0 0 0 0 0 0 0 3 0
0 0 2 0 1 0 0 0 1
   46         1   5.0D-02   1.0D-01      20.0          2
 -50. -599.9743       20.
 -48. -599.9743       20.
 -46. -599.9743       20.
 -44. -599.9743       20.
 -42. -599.9743       20.
 -40. -599.9743       20.
 -38. -599.9743       20.
 -36. -599.9743       20.
 -34. -599.9743       20.
 -32. -599.9743       20.
 -30. -599.9743       20.
 -29. -599.9743       20.
 -28. -599.9743       20.
 -27. -599.9743       20.
 -26. -599.9743       20.
 -25. -599.9743       20.
 -24. -599.9743       20.
 -23. -599.9743       20.
 -22. -599.9743       20.
 -21. -599.9743       20.
 -20. -599.9743       20.
 -19. -599.9743       20.
 -18. -599.9743       20.
 -17. -599.9743       20.
 -16. -599.9743       20.
 -15. -599.9743       20.
 -14. -599.9743       20.
 -13. -599.9743       20.
 -12. -599.9743       20.
 -11. -599.9743       20.
 -10. -599.9743       20.
  -9. -599.9743       20.
  -8. -599.9743       20.
  -7. -599.9743       20.
  -6. -599.9743       20.
  -5. -599.9743       20.
 -4.5 -599.9743       20.
 -4.0 -599.9743       20.
 -3.5 -599.9743       20.
 -3.0 -599.9743       20.
 -2.5 -599.9743       20.
 -2.0 -599.9743       20.
 -1.5 -599.9743       20.
 -1.0 -599.9743       20.
 -0.5 -599.9743       20.
 -0.0 -599.9743       20.
    1         1       0.0
```

1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
1	1	0.0			
0.495	1.0	0.05	10.0	0.0000123	4.
1.D+03	0.1	1	0.00D+00	1	20.0
9.D+03	-0.1	1	0.00D+00	1	20.0
-1.					

OUTPUT OPTIONS -- 1=ACTION 0=NO ACTION
****

OUTPUT OPTIONS	I1	1
SIMULATION OPTIONS	I2	1
OTHER SIMULATION SCALARS	I3	1
NODE AND ELEMENT INFORMATION	I4	1
INPUT SOIL INFORMATION	I5	1
COMPUTED SOIL PARAMETERS	I6	1
SIMULATION PERIOD PARAMETERS	I7	1
MATRIC POTENTIAL	I8	0
-1 - NEVER		
0 - EVERY SIMULATION PERIOD		
1 - EVERY TIME STEP		
2 - EVERY ITERATION		
TEMPERATURE SOLUTION	I9	-1
(SAME CODE AS FOR I8)		
MOISTURE CONTENT	I10	0
-1 - NEVER		
0 - EVERY SIMULATION PERIOD		
1 - EVERY TIME STEP		
MASS BALANCE INFORMATION	I11	0
(SAME CODE AS FOR I10)		
ENERGY BALANCE INFORMATION	I12	-1
(SAME CODE AS FOR I10)		
NOT IN USE	I13	0
NOT IN USE	I14	0
NOT IN USE	I15	0
NOT IN USE	I16	0
MOISTURE EQUATION COEFFICIENTS	I17	0
MOISTURE EQUATION MATRICES	I18	0
MOISTURE EQUATION	I19	0
TEMPERATURE EQUATION COEFFICIENTS	I20	0
TEMPERATURE EQUATION MATRICES	I21	0
TEMPERATURE EQUATION	I22	0
WRITE SOLUTION ON FILE I23 EACH PERIOD	I23	0
IF EQUAL TO 3, USE ISOTHERMAL EQUATION	I24	3
NOT IN USE	I25	0

SIMULATION OPTIONS
****

NOT IN USE	J1	0
NOT IN USE	J2	0
MASS LUMPING OPTION	J3	2
ENERGY LUMPING OPTION	J4	0
USE ALTERNATE RETENTION CURVE	J5	1
NOT IN USE	J6	0

NOT IN USE	J7	0
NOT IN USE	J8	0
ITERATIVE MOISTURE CAPACITY SCHEME	J9	1

OTHER SIMULATION SCALARS
========================

NUMBER OF NODES	46	
NUMBER OF ELEMENTS	45	
NUMBER OF SOIL TYPES	1	
MOISTURE CONVERGENCE CRITERION	0.5D-01	
TEMP. CONVERGENCE CRITERION	0.1D+00	DEG CELSIUS
REFERENCE TEMPERATURE	20.	DEG CELSIUS
DESIRED NO. OF ITERATIONS	2	

NODAL DATA
==========

NODE	Z	P	T
1	-50.00	-600.	20.0
2	-48.00	-600.	20.0
3	-46.00	-600.	20.0
4	-44.00	-600.	20.0
5	-42.00	-600.	20.0
6	-40.00	-600.	20.0
7	-38.00	-600.	20.0
8	-36.00	-600.	20.0
9	-34.00	-600.	20.0
10	-32.00	-600.	20.0
11	-30.00	-600.	20.0
12	-29.00	-600.	20.0
13	-28.00	-600.	20.0
14	-27.00	-600.	20.0
15	-26.00	-600.	20.0
16	-25.00	-600.	20.0
17	-24.00	-600.	20.0
18	-23.00	-600.	20.0
19	-22.00	-600.	20.0
20	-21.00	-600.	20.0
21	-20.00	-600.	20.0
22	-19.00	-600.	20.0
23	-18.00	-600.	20.0
24	-17.00	-600.	20.0
25	-16.00	-600.	20.0
26	-15.00	-600.	20.0

27	-14.00	-600.	20.0
28	-13.00	-600.	20.0
29	-12.00	-600.	20.0
30	-11.00	-600.	20.0
31	-10.00	-600.	20.0
32	-9.00	-600.	20.0
33	-8.00	-600.	20.0
34	-7.00	-600.	20.0
35	-6.00	-600.	20.0
36	-5.00	-600.	20.0
37	-4.50	-600.	20.0
38	-4.00	-600.	20.0
39	-3.50	-600.	20.0
40	-3.00	-600.	20.0
41	-2.50	-600.	20.0
42	-2.00	-600.	20.0
43	-1.50	-600.	20.0
44	-1.00	-600.	20.0
45	-0.50	-600.	20.0
46	0.0	-600.	20.0

INPUT SOIL PARAMETERS

SOIL TYPE	POR	PB	XWILT	EM	CKSAT	PFINT1
1	0.49	1.	0.05	10.	0.12D-04	4.0000

COMPUTED SOIL PARAMETERS

SOIL TYPE	A1	A2	S1	S2	XK
1	0.49367	0.06202	0.12375	0.01786	-0.02822

SOIL TYPE 1

XR	-0.28D-01	-0.28D-01	-0.28D-01	-0.28D-01	-0.28D-01	-0.28D-01	-0.28D-01	-0.28D-01	0.49D+00
PFR	0.59D+01	0.59D+01	0.59D+01	0.59D+01	0.59D+01	0.59D+01	0.59D+01	0.59D+01	0.0
PR	-0.71D+06	-0.71D+06	-0.71D+06	-0.71D+06	-0.71D+06	-0.71D+06	-0.71D+06	-0.71D+06	-0.10D+01
SS	0.31D+02	0.31D+02	0.31D+02	0.31D+02	0.31D+02	0.31D+02	0.31D+02	0.11D+02	
ENT	-0.10D-16	-0.21D-16	-0.31D-16	-0.41D-16	-0.52D-16	-0.62D-16	-0.73D-16	-0.89D-01	

ELEMENT DATA

ELEMENT	IS	IH	X(L,1)	X(L,2)	XWRE(L,1)	XWRE(L,2)	XOLD
1	1	1	0.2376	0.2376	0.0	0.0	0.2376
2	1	1	0.2376	0.2376	0.0	0.0	0.2376
3	1	1	0.2376	0.2376	0.0	0.0	0.2376
4	1	1	0.2376	0.2376	0.0	0.0	0.2376
5	1	1	0.2376	0.2376	0.0	0.0	0.2376
6	1	1	0.2376	0.2376	0.0	0.0	0.2376
7	1	1	0.2376	0.2376	0.0	0.0	0.2376
8	1	1	0.2376	0.2376	0.0	0.0	0.2376
9	1	1	0.2376	0.2376	0.0	0.0	0.2376
10	1	1	0.2376	0.2376	0.0	0.0	0.2376
11	1	1	0.2376	0.2376	0.0	0.0	0.2376
12	1	1	0.2376	0.2376	0.0	0.0	0.2376
13	1	1	0.2376	0.2376	0.0	0.0	0.2376
14	1	1	0.2376	0.2376	0.0	0.0	0.2376
15	1	1	0.2376	0.2376	0.0	0.0	0.2376
16	1	1	0.2376	0.2376	0.0	0.0	0.2376
17	1	1	0.2376	0.2376	0.0	0.0	0.2376
18	1	1	0.2376	0.2376	0.0	0.0	0.2376
19	1	1	0.2376	0.2376	0.0	0.0	0.2376
20	1	1	0.2376	0.2376	0.0	0.0	0.2376
21	1	1	0.2376	0.2376	0.0	0.0	0.2376
22	1	1	0.2376	0.2376	0.0	0.0	0.2376
23	1	1	0.2376	0.2376	0.0	0.0	0.2376
24	1	1	0.2376	0.2376	0.0	0.0	0.2376
25	1	1	0.2376	0.2376	0.0	0.0	0.2376
26	1	1	0.2376	0.2376	0.0	0.0	0.2376
27	1	1	0.2376	0.2376	0.0	0.0	0.2376
28	1	1	0.2376	0.2376	0.0	0.0	0.2376
29	1	1	0.2376	0.2376	0.0	0.0	0.2376
30	1	1	0.2376	0.2376	0.0	0.0	0.2376
31	1	1	0.2376	0.2376	0.0	0.0	0.2376
32	1	1	0.2376	0.2376	0.0	0.0	0.2376
33	1	1	0.2376	0.2376	0.0	0.0	0.2376
34	1	1	0.2376	0.2376	0.0	0.0	0.2376
35	1	1	0.2376	0.2376	0.0	0.0	0.2376
36	1	1	0.2376	0.2376	0.0	0.0	0.2376
37	1	1	0.2376	0.2376	0.0	0.0	0.2376
38	1	1	0.2376	0.2376	0.0	0.0	0.2376
39	1	1	0.2376	0.2376	0.0	0.0	0.2376
40	1	1	0.2376	0.2376	0.0	0.0	0.2376
41	1	1	0.2376	0.2376	0.0	0.0	0.2376
42	1	1	0.2376	0.2376	0.0	0.0	0.2376
43	1	1	0.2376	0.2376	0.0	0.0	0.2376
44	1	1	0.2376	0.2376	0.0	0.0	0.2376
45	1	1	0.2376	0.2376	0.0	0.0	0.2376

****

SIMULATION PERIOD NUMBER 1
==========

DURATION	1000.00	SECONDS
START TIME	0.0	SECONDS
END TIME	0.100D+04	SECONDS
NBCP	1	
BCP	0.0	
NBCT	1	
BCT	0.20D+02	

MASS BALANCE INFORMATION
==========

DATA FOR CURRENT TIME STEP

SURFACE FLUX RATE	-0.19876D-03	CM/S
BOTTOM FLUX RATE	-0.18513D-07	CM/S
NET FLUX RATE	0.19875D-03	CM/S
RATE OF STORAGE CHANGE	0.19824D-03	CM/S
PERCENT ERROR	-0.25290	

CUMULATIVE DATA

NET TOTAL FLUX	0.44452D+00	CM
TOTAL STORAGE CHANGE	0.44092D+00	CM
PERCENT ERROR	-0.81632	

MATRIC HEAD

1	-599.97	2	-599.97	3	-599.97	4	-599.97	5	-599.97
6	-599.97	7	-599.97	8	-599.97	9	-599.97	10	-599.97
11	-599.97	12	-599.97	13	-599.97	14	-599.97	15	-599.97
16	-599.97	17	-599.97	18	-599.97	19	-599.97	20	-599.97
21	-599.97	22	-599.97	23	-599.97	24	-599.97	25	-599.97
26	-599.97	27	-599.97	28	-599.97	29	-599.97	30	-599.97
31	-599.97	32	-599.97	33	-599.97	34	-599.97	35	-599.97
36	-599.95	37	-599.74	38	-598.20	39	-587.48	40	-525.73
41	-332.72	42	-140.61	43	-55.41	44	-23.20	45	-8.86
46	0.0								

MOISTURE CONTENT

1	1	0.237600048	1	2	0.237600048	2	1	0.237600048	2	2	0.237600048	3	1	0.237600048
3	2	0.237600048	4	1	0.237600048	4	2	0.237600048	5	1	0.237600048	5	2	0.237600048
6	1	0.237600048	6	2	0.237600048	7	1	0.237600048	7	2	0.237600048	8	1	0.237600048
8	2	0.237600048	9	1	0.237600048	9	2	0.237600048	10	1	0.237600048	10	2	0.237600048
11	1	0.237600048	11	2	0.237600048	12	1	0.237600048	12	2	0.237600048	13	1	0.237600048
13	2	0.237600048	14	1	0.237600048	14	2	0.237600048	15	1	0.237600048	15	2	0.237600048
16	1	0.237600048	16	2	0.237600048	17	1	0.237600048	17	2	0.237600048	18	1	0.237600048
18	2	0.237600048	19	1	0.237600048	19	2	0.237600048	20	1	0.237600048	20	2	0.237600048
21	1	0.237600048	21	2	0.237600048	22	1	0.237600048	22	2	0.237600048	23	1	0.237600048
23	2	0.237600048	24	1	0.237600048	24	2	0.237600048	25	1	0.237600048	25	2	0.237600048
26	1	0.237600048	26	2	0.237600048	27	1	0.237600048	27	2	0.237600048	28	1	0.237600048
28	2	0.237600048	29	1	0.237600048	29	2	0.237600048	30	1	0.237600048	30	2	0.237600048
31	1	0.237600048	31	2	0.237600048	32	1	0.237600048	32	2	0.237600048	33	1	0.237600048
33	2	0.237600050	34	1	0.237600050	34	2	0.237600108	35	1	0.237600108	35	2	0.237601780
36	1	0.237601780	36	2	0.237618976	37	1	0.237618976	37	2	0.237745811	38	1	0.237745811
38	2	0.238641359	39	1	0.238641359	39	2	0.244282830	40	1	0.244282830	40	2	0.270117908
41	1	0.270117908	41	2	0.328994923	42	1	0.328994923	42	2	0.398499993	43	1	0.398499993
43	2	0.451660433	44	1	0.451660433	44	2	0.483968231	45	1	0.483968231	45	2	0.495000000

****

SIMULATION PERIOD NUMBER 2

DURATION	9000.00	SECONDS
START TIME	0.100D+04	SECONDS
END TIME	0.100D+05	SECONDS
NBCP	1	
BCP	0.0	
NBCT	1	
BCT	0.20D+02	

MASS BALANCE INFORMATION

DATA FOR CURRENT TIME STEP

SURFACE FLUX RATE	-0.67074D-04 CM/S
BOTTOM FLUX RATE	-0.18513D-07 CM/S
NET FLUX RATE	0.67056D-04 CM/S

RATE OF STORAGE CHANGE 0.66924D-04 CM/S

PERCENT ERROR -0.19633

CUMULATIVE DATA

NET TOTAL FLUX 0.13138D+01 CM
TOTAL STORAGE CHANGE 0.13045D+01 CM

PERCENT ERROR -0.71319

MATRIC HEAD

1	-599.97	2	-599.97	3	-599.97	4	-599.97	5	-599.97
6	-599.97	7	-599.97	8	-599.97	9	-599.97	10	-599.97
11	-599.97	12	-599.97	13	-599.97	14	-599.97	15	-599.97
16	-599.97	17	-599.97	18	-599.97	19	-599.97	20	-599.97
21	-599.97	22	-599.97	23	-599.97	24	-599.97	25	-599.97
26	-599.97	27	-599.96	28	-599.89	29	-599.56	30	-598.04
31	-591.37	32	-564.65	33	-478.68	34	-313.85	35	-162.24
36	-78.61	37	-53.68	38	-37.81	39	-27.32	40	-20.05
41	-14.76	42	-10.71	43	-7.45	44	-4.71	45	-2.27
46	0.0								

MOISTURE CONTENT

1	1	0.237600048	1	2	0.237600048	2	1	0.237600048	2	2	0.237600048	3	1	0.237600048
3	2	0.237600048	4	1	0.237600048	4	2	0.237600048	5	1	0.237600048	5	2	0.237600048
6	1	0.237600048	6	2	0.237600048	7	1	0.237600048	7	2	0.237600048	8	1	0.237600048
8	2	0.237600048	9	1	0.237600048	9	2	0.237600048	10	1	0.237600048	10	2	0.237600048
11	1	0.237600048	11	2	0.237600048	12	1	0.237600048	12	2	0.237600048	13	1	0.237600048
13	2	0.237600048	14	1	0.237600048	14	2	0.237600048	15	1	0.237600048	15	2	0.237600048
16	1	0.237600048	16	2	0.237600048	17	1	0.237600048	17	2	0.237600048	18	1	0.237600048
18	2	0.237600048	19	1	0.237600048	19	2	0.237600048	20	1	0.237600048	20	2	0.237600048
21	1	0.237600048	21	2	0.237600048	22	1	0.237600048	22	2	0.237600049	23	1	0.237600049
23	2	0.237600057	24	1	0.237600057	24	2	0.237600096	25	1	0.237600096	25	2	0.237600308
26	1	0.237600308	26	2	0.237601410	27	1	0.237601410	27	2	0.237606955	28	1	0.237606955
28	2	0.237633848	29	1	0.237633848	29	2	0.237759005	30	1	0.237759005	30	2	0.238314139
31	1	0.238314139	31	2	0.240626903	32	1	0.240626903	32	2	0.249236664	33	1	0.249236664
33	2	0.273712816	34	1	0.273712816	34	2	0.318481380	35	1	0.318481380	35	2	0.372825898
36	1	0.372825898	36	2	0.400747672	37	1	0.400747672	37	2	0.424261999	38	1	0.424261999
38	2	0.443295867	39	1	0.443295867	39	2	0.458426553	40	1	0.458426553	40	2	0.470381922
41	1	0.470381922	41	2	0.479777386	42	1	0.479777386	42	2	0.487002868	43	1	0.487002868
43	2	0.492134519	44	1	0.492134519	44	2	0.494772514	45	1	0.494772514	45	2	0.495000000

Appendix B

Listing of SPLaSHWaTrl

```
C*****************************************************************MAI00010
C                                                                 MAI00020
C                        SPLASHWATR1                              MAI00030
C                        -----------                              MAI00040
C                                                                 MAI00050
C                                                                 MAI00060
C   SIMULATION PROGRAM FOR LAND SURFACE HEAT AND WATER TRANSPORT. MAI00070
C                         EDITION 1.                              MAI00080
C                                                                 MAI00090
C                        23 JUNE 1980                             MAI00100
C                                                                 MAI00110
C*****************************************************************MAI00120
C                                                                 MAI00130
C                                                                 MAI00140
C       THIS COMPUTER PROGRAM SIMULATES THE ONE-DIMENSIONAL FLOW OF MAI00150
C   MOISTURE AND HEAT IN A VERTICAL SOIL COLUMN BY SOLVING THE    MAI00160
C   PRIMITIVE EQUATIONS OF MASS AND HEAT CONSERVATION USING THE   MAI00170
C   GALERKIN FINITE ELEMENT METHOD. THIS PRELIMINARY VERSION OF   MAI00180
C   THE PROGRAM WAS USED TO SOLVE THE EXAMPLE PROBLEMS IN CHAPTER MAI00190
C   5 OF P.C.D. MILLY, "THE COUPLED TRANSPORT OF WATER AND HEAT IN MAI00200
C   A VERTICAL SOIL COLUMN UNDER ATMOSPHERIC EXCITATION," S.M. THESIS. MAI00210
C   M.I.T. DEPARTMENT OF CIVIL ENGINEERING, 1980. ALL REFERENCES CON- MAI00220
C   TAINED IN THIS DOCUMENTATION PERTAIN TO THAT THESIS, WHICH CONTAINS MAI00230
C   AN OUTLINE OF THE NUMERICAL METHOD EMPLOYED HERE (CHAPTER 4).  MAI00240
C                                                                 MAI00250
C                                                                 MAI00260
C*****************************************************************MAI00270
C*****************************************************************MAI00280
C                                                                 MAI00290
C   LIST OF VARIABLES CONTAINED IN UNNAMED COMMON BLOCK           MAI00300
C                                                                 MAI00310
C*****************************************************************MAI00320
C   NAME          DESCRIPTION                                     MAI00330
C*****************************************************************MAI00340
C   C(NN,2)       CONDUCTIVITY MATRIX ASSOCIATED WITH MATRIC HEAD MAI00350
C                 (DIAGONAL ELEMENTS ARE STORED IN C(I,1). ABOVE- MAI00360
C                 DIAGONAL ELEMENTS IN C(I,2). NO OTHER ELEMENTS OF MAI00370
C                 THIS OR OTHER SYMMETRIC, TRIDIAGONAL MATRICES NEED TOMAI00380
C                 BE STORED.                                      MAI00390
C   D(NN,2)       CONDUCTIVITY MATRIX ASSOCIATED WITH TEMPERATURE MAI00400
C   B1(NL,2)      VOLUMETRIC AIR CONTENT OF MEDIUM                MAI00410
C   B7(NL,2)      "HYDRAULIC CONDUCTIVITY" ASSOCIATED WITH VAPOR PHASE MAI00420
C   CK(NL,2)      HYDRAULIC CONDUCTIVITY FOR LIQUID FLOW          MAI00430
C   C1(NL,2)      STORAGE COEFFICIENT ASSOCIATED WITH MATRIC HEAD MAI00440
C   C2(NL,2)      STORAGE COEFFICIENT ASSOCIATED TEMPERATURE      MAI00450
C   C3(NL,2)      CONDUCTIVITY TERM ASSOCIATED WITH MATRIC HEAD   MAI00460
C   C4(NL,2)      CONDUCTIVITY TERM ASSOCIATED WITH TEMPERATURE   MAI00470
C   C5(NL,2)      "CONSTANT" VECTOR TERM                          MAI00480
C   DXDP(NL,2)    RATE OF CHANGE OF MOISTURE CONTENT WITH RESPECT TO MAI00490
C                 CAPITAL "PSI"                                   MAI00500
```

```
C   EF(NL,2)        LIQUID ISLAND ENHANCEMENT FACTOR FOR VAPOR FLOW      MAI00510
C   X(NL,2)         MOISTURE CONTENT AT BEGINNING OF NEW TIME STEP       MAI00520
C   XX(NL,2)        MOISTURE CONTENT AT END OF CURRENT TIME STEP         MAI00530
C                   (UPDATED ITERATIVELY)                                MAI00540
C   XWRE(NL,2)      VALUE OF THE MAIN WETTING FUNCTION EVALUATED AT      MAI00550
C                   THE EFFECTIVE REVERSAL VALUE OF CAPITAL "PSI"        MAI00560
C   B3(NN)          DERIVATIVE OF VAPOR DENSITY WITH RESPECT TO MATRIC   MAI00570
C                   HEAD, THE TEMPERATURE BEING HELD CONSTANT            MAI00580
C   B4(NN)          DERIVATIVE OF VAPOR DENSITY WITH RESPECT TO          MAI00590
C                   TEMPERATURE, THE MATRIC HEAD BEING HELD CONSTANT     MAI00600
C   COR(NN)         THE TEMPERATURE CORRECTION FACTOR THAT CONVERTS      MAI00610
C                   MATRIC HEAD TO CAPITAL "PSI"                         MAI00620
C   P(NN)           THE MATRIC HEAD AT THE BEGINNING OF THE CURRENT      MAI00630
C                   TIME STEP                                            MAI00640
C   PFU(NN)         LOGARITHM TO THE BASE 10 OF CAPITAL "PSI"            MAI00650
C   PP(NN)          THE MATRIC HEAD AT THE END OF THE CURRENT TIME STEP. MAI00660
C                   CALCULATED ITERATIVELY                               MAI00670
C   PPOLD(NN)       VALUE OF PP FROM THE LAST ITERATION                  MAI00680
C   RHOV(NN)        VAPOR DENSITY                                        MAI00690
C   T(NN)           TEMPERATURE AT THE START OF THE CURRENT TIME STEP    MAI00700
C   TT(NN)          TEMPERATURE AT THE END OF THE CURRENT TIME STEP.     MAI00710
C                   CALCULATED ITERATIVELY                               MAI00720
C   TTOLD(NN)       VALUE OF TT FROM LAST ITERATION                      MAI00730
C   Z(NN)           Z-COORDINATE OF NODE                                 MAI00740
C   DELTA(NL)       LENGTH OF ELEMENT                                    MAI00750
C   XOLD(NL)        AVERAGE MOISTURE CONTENT IN ELEMENT AT START OF LAST MAI00760
C                   TIME STEP                                            MAI00770
C   POR(NS)         MAXIMUM MOISTURE CONTENT REACHED UPON RE-WETTING     MAI00780
C                   A PREVIOUSLY DRY SOIL. DENOTED BY THETA-SUB-U        MAI00790
C                   IN MILLY (1980), IT IS APPROXIMATELY                 MAI00800
C                   NINE-TENTHS OF TOTAL POROSITY                        MAI00810
C   BCP             VALUE OF SPECIFIED MATRIC HEAD OR OF MASS FLUX AT    MAI00820
C                   SURFACE BOUNDARY                                     MAI00830
C   BCT             VALUE OF TEMPERATURE OR OF HEAT FLUX AT SURFACE      MAI00840
C                   BOUNDARY                                             MAI00850
C   CPLGE           PRODUCT OF CPSI AND LOG BASE 10 OF E                 MAI00860
C   CPSI            TEMPERATURE COEFFICIENT OF MATRIC HEAD               MAI00870
C   DELT            DURATION OF CURRENT TIME STEP                        MAI00880
C   DELTO           DURATION OF LAST TIME STEP                           MAI00890
C   DLOGE           LOG BASE 10 OF E                                     MAI00900
C   DLN10           NATURAL LOG OF TEN                                   MAI00910
C   DLT             POSSIBLE INPUT VALUE OF DELT FOR FIRST TIME STEP OF  MAI00920
C                   A SIMULATION PERIOD                                  MAI00930
C   DURTN           DURATION OF CURRENT SIMULATION PERIOD                MAI00940
C   PERR            CONVERGENCE CRITERION FOR MATRIC HEAD                MAI00950
C   PN              SPECIFIED VALUE OF MATRIC HEAD WHEN FIRST-TYPE B.C.  MAI00960
C                   IS USED                                              MAI00970
C   RHOL            DENSITY OF LIQUID WATER                              MAI00980
C   ST              CURRENT TOTAL ENTHALPY STORAGE                       MAI00990
C   STI             INITIAL VALUE OF ST AT START OF ENTIRE SIMULATION    MAI01000
```

```
C   SX              CURRENT TOTAL WATER STORAGE                                 MAI01010
C   SXI             INITIAL VALUE OF SX                                         MAI01020
C   TEND            TIME TO BE REACHED AT THE END OF THE CURRENT                MAI01030
C                   SIMULATION PERIOD                                           MAI01040
C   TERR            CONVERGENCE CRITERION FOR TEMPERATURE                       MAI01050
C   TFLUX           CUMULATIVE NET INFLUX OF HEAT THROUGH BOUNDARIES            MAI01060
C   THETA           NOT IN USE                                                  MAI01070
C   TIME            TIME ELAPSED SINCE START OF SIMULATION                      MAI01080
C   TSAVE1          TEMPERATURE AT END OF TIME STEP AT BOTTOM OF COLUMN         MAI01090
C                   SAVED FOR LATER USE IN TBC2                                 MAI01100
C   TSAVE2          NOT IN USE                                                  MAI01110
C   TZERO           REFERENCE TEMPERATURE                                       MAI01120
C   XFLUX           CUMULATIVE NET INFLUX OF WATER THROUGH BOUNDARIES           MAI01130
C   ZETA            TEMPERATURE GRADIENT ENHANCEMENT FACTOR FOR VAPOR           MAI01140
C                   FLOW                                                        MAI01150
C   IH(NL)          WETTING DIRECTION VARIABLE - 1 IF WETTING,                  MAI01160
C                   2 IF DRYING                                                 MAI01170
C   IS(NL)          SOIL TYPE OF ELEMENT                                        MAI01180
C   I1 - I25        OUTPUT OPTIONS (1=ACTION; 0=NO ACTION)                      MAI01190
C   I1              OUTPUT OPTIONS                                              MAI01200
C   I2              SIMULATIONS OPTIONS (J1-J9)                                 MAI01210
C   I3              OTHER SIMULATION SCALARS                                    MAI01220
C   I4              NODE AND ELEMENT INFORMATION                                MAI01230
C   I5              INPUT SOIL INFORMATION                                      MAI01240
C   I6              COMPUTED SOIL PARAMETERS                                    MAI01250
C   I7              SIMULATION PERIOD PARAMETERS                                MAI01260
C   I8              MATRIC POTENTIAL                                            MAI01270
C                        -1 - NEVER                                             MAI01280
C                         0 - EVERY SIMULATION PERIOD                           MAI01290
C                         1 - EVERY TIME STEP                                   MAI01300
C                         2 - EVERY ITERATION                                   MAI01310
C   I9              TEMPERATURE (SAME CODE AS FOR I8)                           MAI01320
C   I10             MOISTURE CONTENT                                            MAI01330
C                        -1 - NEVER                                             MAI01340
C                         0 - EVERY SIMULATION PERIOD                           MAI01350
C                         1 - EVERY TIME STEP                                   MAI01360
C   I11             MASS BALANCE INFORMATION (SAME CODE AS FOR I10)             MAI01370
C   I12             ENERGY BALANCE INFORMATION (SAME CODE AS FOR I10)           MAI01380
C   I13             NOT IN USE                                                  MAI01390
C   I14             NOT IN USE                                                  MAI01400
C   I15             NOT IN USE                                                  MAI01410
C   I16             NOT IN USE                                                  MAI01420
C   I17             MOISTURE EQUATION COEFFICIENTS                              MAI01430
C   I18             MOISTURE EQUATION MATRICES                                  MAI01440
C   I19             MOISTURE EQUATION                                           MAI01450
C   I20             TEMPERATURE EQUATION COEFFICIENTS                           MAI01460
C   I21             TEMPERATURE EQUATION MATRICES                               MAI01470
C   I22             TEMPERATURE EQUATION                                        MAI01480
C   I23             IF NON-ZERO, WRITE SOLUTION AT END OF PERIOD ON             MAI01490
C                   FILE I23                                                    MAI01500
```

```
C   I24             IF EQUAL TO 3, SOLVE ISOTHERMAL MOISTURE EQUATION     MAI01510
C   I25             NOT IN USE                                            MAI01520
C   J1 - J9         SIMULATION OPTIONS (1=ACTION; 0=NO ACTIONS)           MAI01530
C   J1              NOT IN USE                                            MAI01540
C   J2              NOT IN USE                                            MAI01550
C   J3              EQUALS 1 OR 2 FOR A MASS LUMPING OPTION               MAI01560
C   J4              EQUALS 1 OR 2 FOR AN ENERGY LUMPING OPTION            MAI01570
C   J5              ALTERNATE WETTING CURVE EXPRESSION TO BE USED         MAI01580
C   J6              NOT IN USE                                            MAI01590
C   J7              NOT IN USE                                            MAI01600
C   J8              IF EQUAL TO 2, USE GEOMETRIC MEAN FOR ELEMENT         MAI01610
C                   INTEGRATION OF CONDUCTIVITY TERMS                     MAI01620
C   J9              NOT IN USE                                            MAI01630
C   ITDES           TARGET NUMBER OF ITERATIONS PER TIME STEP             MAI01640
C   KIT             CURRENT ITERATION NUMBER                              MAI01650
C   KK              NOT IN USE                                            MAI01660
C   KT              TIME STEP NUMBER                                      MAI01670
C   NBCP            TYPE OF SURFACE B.C. ON MATRIC POTENTIAL              MAI01680
C   NBCT            TYPE OF SURFACE B.C. ON TEMPERATURE                   MAI01690
C   NL              NUMBER OF ELEMENTS                                    MAI01700
C   NN              NUMBER OF NODES                                       MAI01710
C   NPER            NUMBER OF SIMULATION PERIOD                           MAI01720
C   NS              NUMBER OF SOIL TYPES                                  MAI01730
C   NSTEPS          NOT IN USE                                            MAI01740
C*************************************************************************MAI01750
C*************************************************************************MAI01760
C                                                                         MAI01770
C   COMMON AND DIMENSION STATEMENTS                                       MAI01780
C                                                                         MAI01790
C*************************************************************************MAI01800
      IMPLICIT REAL*8(A-H,O-Z)                                            MAI01810
      COMMON C(46,2),D(46,2)                                              MAI01820
      COMMON B1(45,2),B7(45,2),CK(45,2),C1(45,2),C2(45,2),C3(45,2),       MAI01830
     *C4(45,2),C5(45,2),DXDP(45,2),EF(45,2),X(45,2),XX(45,2),XWRE(45,2)   MAI01840
      COMMON B3(46),B4(46),COR(46),                                       MAI01850
     *DELTA(46),P(46),                                                    MAI01860
     *PFU(46),PP(46),PPOLD(46),RHOV(46),T(46),                            MAI01870
     *TT(46),TTOLD(46),Z(46)                                              MAI01880
      COMMON XOLD(45)                                                     MAI01890
      COMMON POR(44)                                                      MAI01900
      COMMON BCP,BCT,CPLGE,CPSI,DELT,DELTO,DLOGE,DLN10,DLT,DURTN,PERR,    MAI01910
     *PN,RHOL,ST,STI,SX,SXI,TEND,TERR,TFLUX,THETA,TIME,TSAVE1,TSAVE2,     MAI01920
     *TZERO,XFLUX,ZETA                                                    MAI01930
      COMMON IH(46),IS(46)                                                MAI01940
      COMMON I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15,I16,I17,  MAI01950
     *I18,I19,I20,I21,I22,I23,I24,I25,J1,J2,J3,J4,J5,J6,J7,J8,J9,ITDES,   MAI01960
     *KIT,KK,KT,NBCP,NBCT,NL,NN,NPER,NS,NSTEPS                            MAI01970
C*************************************************************************MAI01980
      DIMENSION A(46,2),B(46,2)                                           MAI01990
C                                                                         MAI02000
```

```
C   A(NN,2)        STORAGE MATRIX ASSOCIATED WITH MATRIC HEAD          MAI02010
C   B(NN,2)        STORAGE MATRIX ASSOCIATED WITH TEMPERATURE          MAI02020
C*******************************************************************MAI02030
C                                                                      MAI02040
C   PERFORM INITIALIZATIONS.                                           MAI02050
C                                                                      MAI02060
      CALL INIT2                                                       MAI02070
C                                                                      MAI02080
C*******************************************************************MAI02090
C                                                                      MAI02100
C   BEGIN A NEW SIMULATION PERIOD.                                     MAI02110
C                                                                      MAI02120
  10  CALL NEWPR2                                                      MAI02130
C                                                                      MAI02140
C*******************************************************************MAI02150
C                                                                      MAI02160
C   BEGIN A NEW TIME STEP.                                             MAI02170
C                                                                      MAI02180
  20  CALL NWSTP2                                                      MAI02190
C                                                                      MAI02200
C*******************************************************************MAI02210
C*******************************************************************MAI02220
C                                                                      MAI02230
C                                                                      MAI02240
C                                                                      MAI02250
C        THE MAIN ITERATION LOOP FOLLOWS. THE MASS AND HEAT            MAI02260
C   CONSERVATION EQUATIONS ARE SOLVED ALTERNATELY UNTIL                MAI02270
C   SOME PRE-SPECIFIED DEGREE OF CONVERGENCE IS OBTAINED.              MAI02280
C   SEE SECTION 4.4.                                                   MAI02290
C                                                                      MAI02300
C                                                                      MAI02310
C                                                                      MAI02320
C*******************************************************************MAI02330
C                                                                      MAI02340
C   DETERMINE THE COEFFICIENTS IN THE MASS CONSERVATION                MAI02350
C   EQUATION (SECTION 4.2.1).                                          MAI02360
C                                                                      MAI02370
  30  CALL PPRAM2                                                      MAI02380
C                                                                      MAI02390
C*******************************************************************MAI02400
C                                                                      MAI02410
C   SET UP THE GALERKIN FINITE ELEMENT MATRICES FOR THE MASS           MAI02420
C   CONSERVATION EQUATION (SECTION 4.2.2).                             MAI02430
C                                                                      MAI02440
      CALL MAT2(A,B,C,D,PP,C1,C2,C3,C4,CK,DELTA,1,NN,NL,I18,I24,J2,J3,J4MAI02450
     1,J8)                                                             MAI02460
C                                                                      MAI02470
C*******************************************************************MAI02480
C                                                                      MAI02490
C   FINITE DIFFERENCE THE TIME DERIVATIVE AND SET UP THE EQUATION      MAI02500
```

```
C   FOR THE MATRIX SOLVER (SECTION 4.3).                                MAI02510
C                                                                       MAI02520
      CALL EQN2(A,B,C,D,P,PP,TT,T,DELT,1.NN.NL.I24,J2)                  MAI02530
C                                                                       MAI02540
C***********************************************************************MAI02550
C                                                                       MAI02560
C   INCORPORATE THE BOUNDARY CONDITIONS INTO THE                        MAI02570
C   MATRIX EQUATION (SECTION 4.2.4).                                    MAI02580
C                                                                       MAI02590
      CALL PBC2                                                         MAI02600
C                                                                       MAI02610
C***********************************************************************MAI02620
C                                                                       MAI02630
C   SOLVE THE TRIDIAGONAL MASS EQUATION (SECTION 4.4).                  MAI02640
C                                                                       MAI02650
      CALL SOLVE2(C,PP,NN)                                              MAI02660
C                                                                       MAI02670
C***********************************************************************MAI02680
C                                                                       MAI02690
      IF(I8.EQ.2.OR.KIT.GT.4*ITDES) WRITE(6,1010) (I.PP(I).I=1.NN)      MAI02700
C                                                                       MAI02710
C***********************************************************************MAI02720
C                                                                       MAI02730
C   REPEAT THE SAME SERIES OF OPERATIONS FOR THE HEAT                   MAI02740
C   CONSERVATION EQUATION. THE ISOTHERMAL MOISTURE FLOW OPTION IS       MAI02750
C   INVOKED WHEN I24 IS EQUAL TO 3, IN WHICH CASE THE HEAT              MAI02760
C   CONSERVATION EQUATION IS SKIPPED.                                   MAI02770
C                                                                       MAI02780
C***********************************************************************MAI02790
C                                                                       MAI02800
      IF(I24.EQ.3) GO TO 40                                             MAI02810
C                                                                       MAI02820
C***********************************************************************MAI02830
C                                                                       MAI02840
      CALL TPRAM2                                                       MAI02850
C                                                                       MAI02860
C***********************************************************************MAI02870
C                                                                       MAI02880
      CALL MAT2(A,B,C,D,TT,C1,C2,C3,C4,C5,DELTA,2,NN,NL,I21,0,J2,J3,J4,JMAI02890
     18)                                                                MAI02900
C                                                                       MAI02910
C***********************************************************************MAI02920
C                                                                       MAI02930
      CALL EQN2(B,A,D,C,T,TT,PP,P,DELT,2,NN,NL,0,J2)                    MAI02940
C                                                                       MAI02950
C***********************************************************************MAI02960
C                                                                       MAI02970
      CALL TBC2                                                         MAI02980
C                                                                       MAI02990
C***********************************************************************MAI03000
```

```
C                                                                       MAI03010
      CALL SOLVE2(D,TT,NN)                                              MAI03020
C                                                                       MAI03030
C***********************************************************************MAI03040
C                                                                       MAI03050
      IF(I9.EQ.2) WRITE(6,1020) (I,TT(I),I=1,NN)                        MAI03060
C                                                                       MAI03070
C***********************************************************************MAI03080
C                                                                       MAI03090
C        CHECK FOR CONVERGENCE (SECTION 4.4). IF CONVERGENCE IS         MAI03100
C   OBTAINED BRANCH OUT OF THE ITERATION LOOP (GO TO 60). IF NOT.       MAI03110
C   PREPARE FOR A NEW ITERATION (GO TO 50).                             MAI03120
C                                                                       MAI03130
C                                                                       MAI03140
  40  CALL CHK2(PP,PPOLD,NN,PERR,ICHK,1)                                MAI03150
      IF(ICHK.EQ.0) GO TO 50                                            MAI03160
      IF(I24.EQ.3) GO TO 60                                             MAI03170
      CALL CHK2(TT,TTOLD,NN,TERR,ICHK,2)                                MAI03180
      IF(ICHK.EQ.0) GO TO 50                                            MAI03190
      GO TO 60                                                          MAI03200
C                                                                       MAI03210
C***********************************************************************MAI03220
C                                                                       MAI03230
  50  IF(I25.GT.0.AND.KIT.EQ.I25) GO TO 60                              MAI03240
      CALL NEWIT2                                                       MAI03250
      GO TO 30                                                          MAI03260
C                                                                       MAI03270
C***********************************************************************MAI03280
C                                                                       MAI03290
C   END OF MAIN ITERATION LOOP.                                         MAI03300
C                                                                       MAI03310
C***********************************************************************MAI03320
C***********************************************************************MAI03330
C                                                                       MAI03340
C   THE ITERATION CYCLE FOR THIS TIME STEP HAS TERMINATED.              MAI03350
C   COMPUTE NEW VALUES OF MOISTURE CONTENT AND PERFORM                  MAI03360
C   THE DESIRED OUTPUT OPERATIONS.                                      MAI03370
C                                                                       MAI03380
  60  CALL SOIL22(1)                                                    MAI03390
      DO 70 L=1,NL                                                      MAI03400
      DO 70 N=1,2                                                       MAI03410
  70  X(L,N)=XX(L,N)                                                    MAI03420
      IF(I8.EQ.1) WRITE(6,1010) (I,PP(I),I=1,NN)                        MAI03430
      IF(I9.EQ.1) WRITE(6,1020) (I,TT(I),I=1,NN)                        MAI03440
      IF(I10.EQ.1) WRITE(6,1030) ((L,N,X(L,N),N=1,2),L=1,NL)            MAI03450
C                                                                       MAI03460
C***********************************************************************MAI03470
C                                                                       MAI03480
C   CALCULATE MASS AND HEAT BALANCES, IF REQUIRED.                      MAI03490
C                                                                       MAI03500
```

```
      IF(I11.GE.0.OR.I12.GE.0) CALL BAL2(0)                              MAI03510
C                                                                        MAI03520
C*************************************************************************MAI03530
C                                                                        MAI03540
C   CHECK FOR THE END OF THE CURRENT SIMULATION PERIOD.                  MAI03550
C   IF IT IS NOT THE END, BEGIN A NEW TIME STEP.                         MAI03560
C                                                                        MAI03570
      IF(DABS(TIME-TEND).GT.1.D-10) GO TO 20                             MAI03580
C                                                                        MAI03590
C*************************************************************************MAI03600
C                                                                        MAI03610
C   IF THE SIMULATION PERIOD HAS ENDED, PERFORM THE DESIRED              MAI03620
C   OUTPUT OPERATIONS AND BEGIN A NEW SIMULATION PERIOD.                 MAI03630
C                                                                        MAI03640
      IF(I8.EQ.0) WRITE(6,1010) (I,PP(I),I=1,NN)                         MAI03650
      IF(I9.EQ.0) WRITE(6,1020) (I,TT(I),I=1,NN)                         MAI03660
      IF(I10.EQ.0) WRITE(6,1030) ((L,N,X(L,N),N=1,2),L=1,NL)             MAI03670
      IF(I23.NE.0) WRITE(I23) PP,TT,X,Z,TIME                             MAI03680
      GO TO 10                                                           MAI03690
C                                                                        MAI03700
C                                                                        MAI03710
C*************************************************************************MAI03720
C                                                                        MAI03730
C   FORMAT STATEMENTS                                                    MAI03740
C                                                                        MAI03750
C*************************************************************************MAI03760
C                                                                        MAI03770
1010  FORMAT(///10X,'MATRIC HEAD'/10X,11(1H-)//,20(2X,5(                 MAI03780
     1I3,2X,F14.2,4X)/))                                                 MAI03790
1020  FORMAT(///10X,'TEMPERATURE'/10X,11(1H-)//,20(2X,5(                 MAI03800
     1I3,2X,F14.11,4X)/))                                                MAI03810
1030  FORMAT(///10X,'MOISTURE CONTENT'/10X,16(1H-)//,20(2X,5(            MAI03820
     12I3,2X,F11.9,4X)/))                                                MAI03830
      END                                                                MAI03840
C                                                                        MAI03850
C                                                                        MAI03860
C*************************************************************************MAI03870
```

```
C*******************************************************************INI00010
C                                                                   INI00020
      SUBROUTINE INIT2                                              INI00030
C                                                                   INI00040
C*******************************************************************INI00050
C                                                                   INI00060
C        THIS SUBROUTINE IS CALLED ONCE AT THE BEGINNING OF A SIMULATIONINI00070
C   TO PERFORM INITIAL INPUT AND OUTPUT OPERATIONS AND TO INITIALIZE  INI00080
C   CERTAIN VARIABLES.                                              INI00090
C                                                                   INI00100
C*******************************************************************INI00110
C                                                                   INI00120
C   COMMON AND DIMENSION STATEMENTS                                 INI00130
C                                                                   INI00140
C*******************************************************************INI00150
      IMPLICIT REAL*8(A-H,O-Z)                                      INI00160
      COMMON C(46,2),D(46,2)                                        INI00170
      COMMON B1(45,2),B7(45,2),CK(45,2),C1(45,2),C2(45,2),C3(45,2),  INI00180
     *C4(45,2),C5(45,2),DXDP(45,2),EF(45,2),X(45,2),XX(45,2),XWRE(45,2) INI00190
      COMMON B3(46),B4(46),COR(46),                                 INI00200
     *DELTA(46),P(46),                                              INI00210
     *PFU(46),PP(46),PPOLD(46),RHOV(46),T(46),                      INI00220
     *TT(46),TTOLD(46),Z(46)                                        INI00230
      COMMON XOLD(45)                                               INI00240
      COMMON POR(44)                                                INI00250
      COMMON BCP,BCT,CPLGE,CPSI,DELT,DELTO,DLOGE,DLN10,DLT,DURTN,PERR,  INI00260
     *PN,RHOL,ST,STI,SX,SXI,TEND,TERR,TFLUX,THETA,TIME,TSAVE1,TSAVE2,  INI00270
     *TZERO,XFLUX,ZETA                                              INI00280
      COMMON IH(46),IS(46)                                          INI00290
      COMMON I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15,I16,I17,INI00300
     *I18,I19,I20,I21,I22,I23,I24,I25,J1,J2,J3,J4,J5,J6,J7,J8,J9,ITDES,  INI00310
     *KIT,KK,KT,NBCP,NBCT,NL,NN,NPER,NS,NSTEPS                      INI00320
      COMMON /CKTEE/ CKTN                                           INI00330
C*******************************************************************INI00340
C                                                                   INI00350
C   INITIALIZE CONSTANTS.                                           INI00360
C                                                                   INI00370
C*******************************************************************INI00380
      DLOGE=0.4342944819D0                                          INI00390
      DLN10=2.3025850930D0                                          INI00400
      CPSI=-0.0017                                                  INI00410
      ZETA=2.0                                                      INI00420
      RHOL=0.998                                                    INI00430
      CPLGE=CPSI*DLOGE                                              INI00440
C*******************************************************************INI00450
C                                                                   INI00460
C   PERFORM INITIAL INPUT AND OUTPUT OPERATIONS.                    INI00470
C                                                                   INI00480
C*******************************************************************INI00490
      READ(5,1020) I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15,  INI00500
```

```
     *I16,I17,I18,I19,I20,I21,I22,I23,I24,I25,I26,I27,I28,I29,I30       INI00510
      IF (I1.NE.1) GO TO 10                                             INI00520
      WRITE(6,1030) I1,I2,I3,I4,I5,I6,I7                                INI00530
      WRITE(6,1040) I8,I9,I10                                           INI00540
      WRITE(6,1050) I11,I12,I13,I14,I15,I16,I17,I18                     INI00550
      WRITE(6,1060) I19,I20,I21,I22,I23,I24,I25                         INI00560
  10  READ(5,1070) J1,J2,J3,J4,J5,J6,J7,J8,J9                           INI00570
      IF(I2.EQ.1) WRITE(6,1080) J1,J2,J3,J4,J5,J6,J7,J8,J9              INI00580
      READ (5,1090) NN,NS,PERR,TERR,TZERO,ITDES                         INI00590
      NL=NN-1                                                           INI00600
      IF(I3.EQ.1) WRITE(6,1100) NN,NL,NS,PERR,TERR,TZERO,ITDES          INI00610
      READ(5,1110) (Z(I),P(I),T(I),I=1,NN)                              INI00620
      IF(I4.EQ.1) WRITE(6,1130) (I,Z(I),P(I),T(I),I=1,NN)               INI00630
      READ(5,1120) (IS(L),IH(L),(XWRE(L,N),N=1,2),L=1,NL)               INI00640
C*****************************************************************    INI00650
C                                                                       INI00660
C   COMPUTE ELEMENT LENGTHS AND INITIALIZE PP AND TT.                   INI00670
C                                                                       INI00680
C*****************************************************************    INI00690
      DO 20 L=1,NL                                                      INI00700
  20  DELTA(L)=Z(L+1)-Z(L)                                              INI00710
      DO 30 I=1,NN                                                      INI00720
      PP(I)=P(I)                                                        INI00730
  30  TT(I)=T(I)                                                        INI00740
C*****************************************************************    INI00750
C                                                                       INI00760
C   THE CALL TO SOILI2 WILL READ SOIL PROPERTY DATA AND COMPUTE         INI00770
C   THE RELEVANT DERIVED QUANTITIES.                                    INI00780
C                                                                       INI00790
      CALL SOILI2                                                       INI00800
C                                                                       INI00810
C*****************************************************************    INI00820
C                                                                       INI00830
C   COMPUTE INITIAL MOISTURE CONTENTS AND INITIALIZE X AND XOLD.        INI00840
C                                                                       INI00850
C*****************************************************************    INI00860
      CALL SOIL22(1)                                                    INI00870
      DO 40 L=1,NL                                                      INI00880
      X(L,1)=XX(L,1)                                                    INI00890
      X(L,2)=XX(L,2)                                                    INI00900
  40  XOLD(L)=(X(L,1)+X(L,2))/2.                                        INI00910
      IF(I4.EQ.1) WRITE(6,1140) (L,IS(L),IH(L),X(L,1),X(L,2),(XWRE(L,N),INI00920
     1N=1,2),XOLD(L),L=1,NL)                                            INI00930
C*****************************************************************    INI00940
C                                                                       INI00950
C   INITIALIZE VARIABLES.                                               INI00960
C                                                                       INI00970
C*****************************************************************    INI00980
      CALL BAL2(1)                                                      INI00990
      SXI=SX                                                            INI01000
```

```
      STI=ST                                                            INI01010
      XFLUX=0.0                                                         INI01020
      TFLUX=0.0                                                         INI01030
      IF(I24.EQ.3) GO TO 50                                             INI01040
      CKTN=1.0                                                          INI01050
      CKTN=CKTT2(TZERO)                                                 INI01060
   50 DELT=1.0                                                          INI01070
      TIME=0.0                                                          INI01080
      NPER=0                                                            INI01090
C*************************************************************************INI01100
C                                                                       INI01110
C   FORMAT STATEMENTS                                                   INI01120
C                                                                       INI01130
C*************************************************************************INI01140
1010  FORMAT('1'///,1X,115(1H*))                                        INI01150
1020  FORMAT(3012)                                                      INI01160
1030  FORMAT(///5X,'OUTPUT OPTIONS -- 1=ACTION  0=NO ACTION'./5X.       INI01170
     139(1H=)//10X,'OUTPUT OPTIONS',26X,'I1  ',I1.                      INI01180
     2/10X,'SIMULATION OPTIONS',22X,'I2  ',I1.                          INI01190
     3/10X,'OTHER SIMULATION SCALARS',16X,'I3  ',I1.                    INI01200
     4/10X,'NODE AND ELEMENT INFORMATION',12X,'I4  ',I1.                INI01210
     5/10X,'INPUT SOIL INFORMATION',18X,'I5  ',I1.                      INI01220
     6/10X,'COMPUTED SOIL PARAMETERS',16X,'I6  ',I1.                    INI01230
     7/10X,'SIMULATION PERIOD PARAMETERS',12X,'I7  ',I1)                INI01240
1040  FORMAT(10X,'MATRIC POTENTIAL',24X,'I8 ',I2.                       INI01250
     1/15X,'-1 - NEVER'                                                 INI01260
     2/16X,'0 - EVERY SIMULATION PERIOD'                                INI01270
     3/16X,'1 - EVERY TIME STEP'                                        INI01280
     4/16X,'2 - EVERY ITERATION'                                        INI01290
     6/10X,'TEMPERATURE SOLUTION',20X,'I9 ',I2.                         INI01300
     7/15X,'(SAME CODE AS FOR I8)'                                      INI01310
     8/10X,'MOISTURE CONTENT',23X,'I10 ',I2/15X,'-1 - NEVER'/16X,'0 - EVINI01320
     9ERY SIMULATION PERIOD'/16X,'1 - EVERY TIME STEP')                 INI01330
1050  FORMAT(10X,'MASS BALANCE INFORMATION',15X,'I11 ',I2.               INI01340
     9/15X,'(SAME CODE AS FOR I10)',                                    INI01350
     1/10X,'ENERGY BALANCE INFORMATION',13X,'I12 ',I2/15X,'(SAME CODE ASINI01360
     1 FOR I10)'/10X,'NOT IN USE',29X,'I13  ',I1                        INI01370
     2/10X,'NOT IN USE',29X,'I14  ',I1.                                 INI01380
     3/10X,'NOT IN USE',29X,'I15  ',I1.                                 INI01390
     4/10X,'NOT IN USE',29X,'I16  ',I1.                                 INI01400
     5/10X,'MOISTURE EQUATION COEFFICIENTS',9X,'I17  ',I1.              INI01410
     6/10X,'MOISTURE EQUATION MATRICES',13X,'I18  ',I1)                 INI01420
1060  FORMAT(10X,'MOISTURE EQUATION',22X,'I19  ',I1.                     INI01430
     8/10X,'TEMPERATURE EQUATION COEFFICIENTS',6X,'I20  ',I1.           INI01440
     9/10X,'TEMPERATURE EQUATION MATRICES',10X,'I21  ',I1.              INI01450
     1/10X,'TEMPERATURE EQUATION',19X,'I22  ',I1.                       INI01460
     2/10X,'WRITE SOLUTION ON FILE I23 EACH PERIOD',1X,'I23',I3.        INI01470
     3/10X,'IF EQUAL TO 3, USE ISOTHERMAL EQUATION',1X,'I24',I3.        INI01480
     4/10X,'NOT IN USE',29X,'I25',I3)                                   INI01490
1070  FORMAT(9I2)                                                       INI01500
```

```
1080  FORMAT(///5X,'SIMULATION OPTIONS'/5X,18(1H=)/                    INI01510
     1/10X,'NOT IN USE',30X,'J1  ',I1,                                 INI01520
     2/10X,'NOT IN USE',30X,'J2  ',I1,                                 INI01530
     3/10X,'MASS LUMPING OPTION',21X,'J3  ',I1,                        INI01540
     4/10X,'ENERGY LUMPING OPTION',19X,'J4  ',I1,                      INI01550
     5/10X,'USE ALTERNATE RETENTION CURVE',11X,'J5  ',I1,              INI01560
     6/10X,'NOT IN USE',30X,'J6  ',I1,                                 INI01570
     7/10X,'NOT IN USE',30X,'J7  ',I1,                                 INI01580
     8/10X,'NOT IN USE',30X,'J8  ',I1,                                 INI01590
     9/10X,'ITERATIVE MOISTURE CAPACITY SCHEME',6X,'J9  ',I1)          INI01600
1090  FORMAT(6G10.0)                                                   INI01610
1100  FORMAT(///5X,'OTHER SIMULATION SCALARS'/5X,24(1H=)/               INI01620
     1/10X,'NUMBER OF NODES',29X,I3,                                   INI01630
     2/10X,'NUMBER OF ELEMENTS',26X,I3                                 INI01640
     3/10X,'NUMBER OF SOIL TYPES',24X,I3,                              INI01650
     4/10X,'MOISTURE CONVERGENCE CRITERION',7X,D10.1,                  INI01660
     5/10X,'TEMP. CONVERGENCE CRITERION',10X,D10.1,' DEG CELSIUS'      INI01670
     6/10X,'REFERENCE TEMPERATURE',16X,F10.0,' DEG CELSIUS',           INI01680
     7/10X,'DESIRED NO. OF ITERATIONS',12X,I10)                        INI01690
1110  FORMAT(3G10.0,50X)                                               INI01700
1120  FORMAT(4G10.0,40X)                                               INI01710
1130  FORMAT(///5X,'NODAL DATA'/5X,10(1H=)//10X,'NODE',10X,            INI01720
     1'Z',14X,'P',14X,'T'/10X,52(1H-)/                                 INI01730
     2(10X,I3,6X,F8.2,7X,F10.0,7X,F5.1))                               INI01740
1140  FORMAT(///5X,'ELEMENT DATA'/5X,12(1H=)//10X,                     INI01750
     1'ELEMENT',5X,'IS',5X,'IH',5X,'X(L,1)',5X,'X(L,2)',5X,'XWRE(L,1)',5INI01760
     2X,'XWRE(L,2)',5X,'XOLD'/10X,81(1H-)/(11X,I4,2X,2I7,2F11.4,F14.4,F1INI01770
     34.4,F10.4))                                                      INI01780
      RETURN                                                         INI01790
      END                                                            INI01800
C*****************************************************************INI01810
```

```
C*****************************************************************NEW00010
C                                                                 NEW00020
      SUBROUTINE NEWPR2                                           NEW00030
C                                                                 NEW00040
C*****************************************************************NEW00050
C                                                                 NEW00060
C        THIS SUBROUTINE READS DATA DEFINING THE BOUNDARY CONDITIONS NEW00070
C   FOR A NEW SIMULATION PERIOD. (A NEW SIMULATION PERIOD BEGINS  NEW00080
C   WHENEVER THE BOUNDARY CONDITIONS CHANGE.) IN GENERAL, A SIMULATION NEW00090
C   PERIOD WILL CONSIST OF MANY TIME STEPS.                       NEW00100
C                                                                 NEW00110
C*****************************************************************NEW00120
C                                                                 NEW00130
C COMMON AND DIMENSION STATEMENTS                                 NEW00140
C                                                                 NEW00150
C*****************************************************************NEW00160
      IMPLICIT REAL*8(A-H,O-Z)                                    NEW00170
      COMMON C(46,2),D(46,2)                                      NEW00180
      COMMON B1(45,2),B7(45,2),CK(45,2),C1(45,2),C2(45,2),C3(45,2), NEW00190
     *C4(45,2),C5(45,2),DXDP(45,2),EF(45,2),X(45,2),XX(45,2),XWRE(45,2) NEW00200
      COMMON B3(46),B4(46),COR(46),                               NEW00210
     *DELTA(46),P(46),                                            NEW00220
     *PFU(46),PP(46),PPOLD(46),RHOV(46),T(46),                    NEW00230
     *TT(46),TTOLD(46),Z(46)                                      NEW00240
      COMMON XOLD(45)                                             NEW00250
      COMMON POR(44)                                              NEW00260
      COMMON BCP,BCT,CPLGE,CPSI,DELT,DELTO,DLOGE,DLN10,DLT,DURTN,PERR, NEW00270
     *PN,RHOL,ST,STI,SX,SXI,TEND,TERR,TFLUX,THETA,TIME,TSAVE1,TSAVE2, NEW00280
     *TZERO,XFLUX,ZETA                                            NEW00290
      COMMON IH(46),IS(46)                                        NEW00300
      COMMON I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15,I16,I17, NEW00310
     *I18,I19,I20,I21,I22,I23,I24,I25,J1,J2,J3,J4,J5,J6,J7,J8,J9,ITDES, NEW00320
     *KIT,KK,KT,NBCP,NBCT,NL,NN,NPER,NS,NSTEPS                    NEW00330
C*****************************************************************NEW00340
C                                                                 NEW00350
C   READ PERIOD DURATION, POSSIBLE SUGGESTED INITIAL TIME STEP.   NEW00360
C   AND BOUNDARY CONDITIONS. PROGRAM EXECUTION IS TERMINATED      NEW00370
C   WHEN A NEGATIVE DURATION IS SPECIFIED.                        NEW00380
C                                                                 NEW00390
C*****************************************************************NEW00400
      READ(5,1010) DURTN,DLT,NBCP,BCP,NBCT,BCT                    NEW00410
      IF (DURTN.LT.0.) STOP                                       NEW00420
      NPER=NPER+1                                                 NEW00430
      TEND=TIME+DURTN                                             NEW00440
      IF(I7.EQ.1) WRITE(6,1020) NPER,DURTN,TIME,TEND,NBCP,BCP,NBCT,BCT NEW00450
      KT=0                                                        NEW00460
      RETURN                                                      NEW00470
C*****************************************************************NEW00480
C                                                                 NEW00490
C   FORMAT STATEMENTS                                             NEW00500
```

```
C                                                                       NEW00510
C***********************************************************************NEW00520
1010  FORMAT(6G10.0)                                                    NEW00530
1020  FORMAT(////1X,115(1H*)///5X,'SIMULATION PERIOD NUMBER',I3./5X,27(1NEW00540
     1H=)//10X,  'DURATION',8X,F9.2,5X,' SECONDS',/10X,'START TIME',8X,DNEW00550
     *12.3,' SECONDS',/10X,'END TIME',10X,D12.3,' SECONDS',            NEW00560
     2/10X,'NBCP',17X,I1/10X,                                          NEW00570
     3'BCP',10X,D16.2/10X,'NBCT',17X,I1/10X,'BCT',10X,D16.2)           NEW00580
      END                                                               NEW00590
C***********************************************************************NEW00600
```

```
C*******************************************************************NWS00010
C                                                                     NWS00020
      SUBROUTINE NWSTP2                                               NWS00030
C                                                                     NWS00040
C*******************************************************************NWS00050
C                                                                     NWS00060
C        THIS SUBROUTINE IS CALLED BY THE MAIN PROGRAM TO PERFORM     NWS00070
C   THE COMPUTATIONS NECESSARY AT THE BEGINNING OF EACH TIME STEP.    NWS00080
C   ENTRY NEWIT2 IS CALLED AT THE START OF EACH SUBSEQUENT ITERATION. NWS00090
C                                                                     NWS00100
C*******************************************************************NWS00110
C                                                                     NWS00120
C   COMMON AND DIMENSION STATEMENTS                                   NWS00130
C                                                                     NWS00140
C*******************************************************************NWS00150
      IMPLICIT REAL*8(A-H,O-Z)                                        NWS00160
      COMMON C(46,2),D(46,2)                                          NWS00170
      COMMON B1(45,2),B7(45,2),CK(45,2),C1(45,2),C2(45,2),C3(45,2),   NWS00180
     *C4(45,2),C5(45,2),DXDP(45,2),EF(45,2),X(45,2),XX(45,2),XWRE(45,2) NWS00190
      COMMON B3(46),B4(46),COR(46),                                   NWS00200
     *DELTA(46),P(46),                                                NWS00210
     *PFU(46),PP(46),PPOLD(46),RHOV(46),T(46),                        NWS00220
     *TT(46),TTOLD(46),Z(46)                                          NWS00230
      COMMON XOLD(45)                                                 NWS00240
      COMMON POR(44)                                                  NWS00250
      COMMON BCP,BCT,CPLGE,CPSI,DELT,DELTO,DLOGE,DLN10,DLT,DURTN,PERR, NWS00260
     *PN,RHOL,ST,STI,SX,SXI,TEND,TERR,TFLUX,THETA,TIME,TSAVE1,TSAVE2,  NWS00270
     *TZERO,XFLUX,ZETA                                                NWS00280
      COMMON IH(46),IS(46)                                            NWS00290
      COMMON I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15,I16,I17,NWS00300
     *I18,I19,I20,I21,I22,I23,I24,I25,J1,J2,J3,J4,J5,J6,J7,J8,J9,ITDES, NWS00310
     *KIT,KK,KT,NBCP,NBCT,NL,NN,NPER,NS,NSTEPS                        NWS00320
      COMMON /ADHOC/ PB(44)                                           NWS00330
      DIMENSION POLD(46),TOLD(46)                                     NWS00340
C*******************************************************************NWS00350
C                                                                     NWS00360
C   INCREMENT THE TIME STEP NUMBER AND CHOOSE A NEW TIME STEP LENGTH. NWS00370
C   THE LENGTH OF THE FIRST TIME STEP OF EACH SIMULATION PERIOD MAY    NWS00380
C   BE SPECIFIED AS DLT. FOR LATER TIME STEPS. OR IF DLT IS LESS THAN NWS00390
C   ZERO, THE NEW VALUE IS CALCULATED BY COMPARING THE NUMBER OF      NWS00400
C   ITERATIONS REQUIRED FOR THE LAST TIME STEP TO THE DESIRED         NWS00410
C   NUMBER OF ITERATIONS, ITDES, AND DELT IS ADJUSTED ACCORDINGLY -   NWS00420
C   INCREASED FOR KIT LESS THAN ITDES, DECREASED FOR KIT GREATER THAN NWS00430
C   ITDES. THE TIME STEP IS REDUCED, IF NECESSARY, TO PREVENT THE     NWS00440
C   TIME STEP FROM OVERSHOOTING THE END OF THE SIMULATION PERIOD.     NWS00450
C                                                                     NWS00460
C*******************************************************************NWS00470
      KT=KT+1                                                         NWS00480
      DELTO=DELT                                                      NWS00490
      IF(KT.EQ.1.AND.DLT.GT.0.) GO TO 20                              NWS00500
```

```
      IF (KIT.GE.ITDES) GO TO 10                                        NWS00510
      DELT=1.2*DELTO                                                    NWS00520
      GO TO 30                                                          NWS00530
   10 DELT=DELTO*ITDES*ITDES/KIT/KIT                                    NWS00540
      GO TO 30                                                          NWS00550
   20 DELT=DLT                                                          NWS00560
   30 IF(DELT.GT.TEND-TIME) DELT=TEND-TIME                              NWS00570
      TIME=TIME+DELT                                                    NWS00580
      IF(I8.GT.0.OR.I9.GT.0.OR.I10.GT.0.OR.I11.GT.0.OR.I12.GT.0)        NWS00590
     1 WRITE(6,1010) KT,TIME                                            NWS00600
C*********************************************************************NWS00610
C                                                                       NWS00620
C   UPDATE THE VARIOUS STATE VARIABLES. (THE PURPOSE OF THE MINIMUM     NWS00630
C   OPERATOR IS DISCUSSED IN SECTION 4.7.3.)                            NWS00640
C                                                                       NWS00650
C*********************************************************************NWS00660
      DO 40 I=1,NN                                                      NWS00670
      POLD(I)=P(I)                                                      NWS00680
      P(I)=DMIN1(PP(I),-PB(1))                                          NWS00690
      IF(I24.EQ.3) GO TO 40                                             NWS00700
      TOLD(I)=T(I)                                                      NWS00710
      T(I)=TT(I)                                                        NWS00720
   40 CONTINUE                                                          NWS00730
      IF(NPER.EQ.1.AND.KT.EQ.1) GO TO 50                                NWS00740
C*********************************************************************NWS00750
C                                                                       NWS00760
C   UPDATE THE WETTING HISTORY ON THE BASIS OF THE LAST                 NWS00770
C   TIME STEP (EXPLICIT METHOD - SECTION 4.5).                          NWS00780
C                                                                       NWS00790
      IF (J5.NE.1) CALL SOIL12                                          NWS00800
C                                                                       NWS00810
C*********************************************************************NWS00820
   50 KIT=1                                                             NWS00830
      IF(I9.EQ.2.OR.I10.EQ.2.OR.I17.EQ.1.OR.I18.EQ.1.OR.I19.EQ.1.OR.I20.NWS00840
     1EQ.1.OR.I21.EQ.1.OR.I22.EQ.1) WRITE(6,1020) KIT                   NWS00850
C*********************************************************************NWS00860
C                                                                       NWS00870
C   ESTIMATE THE MOISTURE POTENTIAL AND TEMPERATURE AT THE END OF THE   NWS00880
C   NEW TIME STEP BY EXTRAPOLATION FOR THE FIRST ITERATION.             NWS00890
C                                                                       NWS00900
C*********************************************************************NWS00910
      DO 60 I=1,NN                                                      NWS00920
      PP(I)=P(I)+(P(I)-POLD(I))*DELT/DELTO                              NWS00930
      PPOLD(I)=PP(I)                                                    NWS00940
      IF(I24.EQ.3) GO TO 60                                             NWS00950
      TT(I)=T(I)+(T(I)-TOLD(I))*DELT/DELTO                              NWS00960
      TTOLD(I)=TT(I)                                                    NWS00970
   60 CONTINUE                                                          NWS00980
C*********************************************************************NWS00990
C                                                                       NWS01000
```

```
C   IF THERE IS A FIRST-TYPE BOUNDARY CONDITION AT THE SURFACE.          NWS01010
C   THE ESTIMATE CAN BE REPLACED BY THE KNOWN VALUE. PN IS SAVED         NWS01020
C   FOR LATER APPLICATION OF THE BOUNDARY CONDITION.                     NWS01030
C                                                                        NWS01040
C*************************************************************************NWS01050
      IF(NBCP.EQ.2) GO TO 70                                             NWS01060
      PN=BCP                                                             NWS01070
      PP(NN)=BCP                                                         NWS01080
      PPOLD(NN)=PP(NN)                                                   NWS01090
  70  RETURN                                                             NWS01100
C*************************************************************************NWS01110
C*************************************************************************NWS01120
C                                                                        NWS01130
      ENTRY NEWIT2                                                       NWS01140
C                                                                        NWS01150
C*************************************************************************NWS01160
C                                                                        NWS01170
C   NEWIT2 IS CALLED AT THE START OF EACH NEW ITERATION TO               NWS01180
C   INCREMENT KIT AND TO UPDATE PPOLD AND TTOLD, WHICH ARE SAVED         NWS01190
C   FOR THE CONVERGENCE CHECK                                            NWS01200
C                                                                        NWS01210
C*************************************************************************NWS01220
      KIT=KIT+1                                                          NWS01230
      IF(I9.EQ.2.OR.I10.EQ.2.OR.I17.EQ.1.OR.I18.EQ.1.OR.I19.EQ.1.OR.I20.NWS01240
     1EQ.1.OR.I21.EQ.1.OR.I22.EQ.1) WRITE(6,1020) KIT                    NWS01250
      DO 80 I=1,NN                                                       NWS01260
      PPOLD(I)=PP(I)                                                     NWS01270
      IF(I24.EQ.3) GO TO 80                                              NWS01280
      TTOLD(I)=TT(I)                                                     NWS01290
  80  CONTINUE                                                           NWS01300
      RETURN                                                             NWS01310
C*************************************************************************NWS01320
C                                                                        NWS01330
C   FORMAT STATEMENTS                                                    NWS01340
C                                                                        NWS01350
C*************************************************************************NWS01360
1010  FORMAT(//1X,115(1H=),//10X,'TIME STEP NUMBER'.I13/10X, 'TOTAL ELAPNWS01370
     1SED TIME',2X,D9.3,' SECONDS')                                      NWS01380
1020  FORMAT(//1X,115(1H_)//12X,'ITERATION NO.',7X,I3)                   NWS01390
      END                                                                NWS01400
C*************************************************************************NWS01410
```

```
C*************************************************************************PPR00010
C                                                                         PPR00020
      SUBROUTINE PPRAM2                                                   PPR00030
C                                                                         PPR00040
C*************************************************************************PPR00050
C                                                                         PPR00060
C        THE PURPOSE OF THIS SUBROUTINE IS TO CALCULATE THE COEFFICIENTSPPR00070
C   OF THE MASS CONSERVATION EQUATION. THESE WILL LATER BE USED IN THE  PPR00080
C   ASSEMBLY OF THE FINITE ELEMENT MATRICES. SOME OF THESE. STORED AS   PPR00090
C   VECTORS OR MATRICES, ARE SAVED FOR EVALUATION OF THE HEAT           PPR00100
C   EQUATION COEFFICIENTS IN SUBROUTINE TRPAM2.                         PPR00110
C                                                                         PPR00120
C*************************************************************************PPR00130
C                                                                         PPR00140
C   THE FOLLOWING VARIABLES AND FUNCTIONS, NOT DEFINED IN THE UNNAMED   PPR00150
C   COMMON BLOCK, APPEAR IN THIS SUBROUTINE:                            PPR00160
C                                                                         PPR00170
C*************************************************************************PPR00180
C   NAME           DESCRIPTION                                          PPR00190
C*************************************************************************PPR00200
C   CKT            TEMPERATURE CORRECTION FOR HYDRAULIC CONDUCTIVITY    PPR00210
C   CKTT2          A FUNCTION THAT RETURNS THE TEMPERATURE CORRECTION   PPR00220
C                  FOR HYDRAULIC CONDUCTIVITY, GIVEN TEMPERATURE        PPR00230
C   DA             DIFFUSION COEFFICIENT OF WATER VAPOR IN AIR          PPR00240
C   DRZDT2         A FUNCTION THAT RETURNS THE SLOPE OF THE SATURATION  PPR00250
C                  VAPOR DENSITY CURVE, GIVEN TEMPERATURE               PPR00260
C   GORT           GRAVITATIONAL ACCELERATION DIVIDED BY THE PRODUCT OF PPR00270
C                  THE GAS CONSTANT OF WATER AND THE ABSOLUTE           PPR00280
C                  TEMPERATURE ("G OVER R T")                           PPR00290
C   INT2           SEE DESCRIPTION IN TEXT BELOW                        PPR00300
C   RH             RELATIVE HUMIDITY                                    PPR00310
C   RHOZZ2         A FUNCTION THAT RETURNS THE VALUE OF THE SATURATION  PPR00320
C                  VAPOR DENSITY, GIVEN THE TEMPERATURE                 PPR00330
C   TK             ABSOLUTE (KELVIN) TEMPERATURE                        PPR00340
C   TORT           TORTUOSITY OF AIR-FILLED SOIL FRACTION.              PPR00350
C*************************************************************************PPR00360
C                                                                         PPR00370
C   COMMON AND DIMENSION STATEMENTS                                     PPR00380
C                                                                         PPR00390
C*************************************************************************PPR00400
      IMPLICIT REAL*8(A-H,O-Z)                                            PPR00410
      COMMON C(46,2),D(46,2)                                              PPR00420
      COMMON B1(45,2),B7(45,2),CK(45,2),C1(45,2),C2(45,2),C3(45,2),       PPR00430
     *C4(45,2),C5(45,2),DXDP(45,2),EF(45,2),X(45,2),XX(45,2),XWRE(45,2)   PPR00440
      COMMON B3(46),B4(46),COR(46),                                       PPR00450
     *DELTA(46),P(46),                                                    PPR00460
     *PFU(46),PP(46),PPOLD(46),RHOV(46),T(46),                            PPR00470
     *TT(46),TTOLD(46),Z(46)                                              PPR00480
      COMMON XOLD(45)                                                     PPR00490
      COMMON POR(44)                                                      PPR00500
```

```
      COMMON BCP,BCT,CPLGE,CPSI,DELT,DELTO,DLOGE,DLN10,DLT,DURTN,PERR,   PPR00510
     *PN,RHOL,ST,STI,SX,SXI,TEND,TERR,TFLUX,THETA,TIME,TSAVE1,TSAVE2,   PPR00520
     *TZERO,XFLUX,ZETA                                                  PPR00530
      COMMON IH(46),IS(46)                                              PPR00540
      COMMON I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15,I16,I17,PPR00550
     *I18,I19,I20,I21,I22,I23,I24,I25,J1,J2,J3,J4,J5,J6,J7,J8,J9,ITDES. PPR00560
     *KIT,KK,KT,NBCP,NBCT,NL,NN,NPER,NS,NSTEPS                          PPR00570
      DIMENSION CKT(46),DA(46)                                          PPR00580
      DIMENSION INT2(46)                                                PPR00590
C*******************************************************************PPR00600
      IF(I17.EQ.1.AND.I24.NE.3) WRITE(6,1040)                           PPR00610
C*******************************************************************PPR00620
C                                                                       PPR00630
C   CALCULATE THE MOISTURE CONTENT, XX, AND THE SPECIFIC MOISTURE       PPR00640
C   CAPACITY, DXDP.                                                     PPR00650
C                                                                       PPR00660
      CALL SOIL22(2)                                                    PPR00670
C                                                                       PPR00680
C*******************************************************************PPR00690
C                                                                       PPR00700
C   CALCULATE THE HYDRAULIC CONDUCTIVITY                                PPR00710
      CALL SOIL32                                                       PPR00720
C                                                                       PPR00730
C*******************************************************************PPR00740
C                                                                       PPR00750
C   IN THE FIRST LOOP, WE COMPUTE THOSE VARIABLES THAT CAN BE           PPR00760
C   DEFINED UNIQUELY AT EACH NODE, INDEPENDENT OF THE ELEMENT. THESE    PPR00770
C   WILL DEPEND ONLY ON TEMPERATURE AND MATRIC POTENTIAL.               PPR00780
C                                                                       PPR00790
C*******************************************************************PPR00800
      DO 10 I=1,NN                                                      PPR00810
      IF(KT.EQ.1) INT2(I)=0                                             PPR00820
      IF(I24.EQ.3) GO TO 10                                             PPR00830
      TK=TT(I)+273.16                                                   PPR00840
      CKT(I)=CKTT2(TT(I))                                               PPR00850
      GORT=2.13D-04/TK                                                  PPR00860
      RH=DEXP(PP(I)*GORT)                                               PPR00870
      RHOV(I)=RH*RHOZZ2(TT(I))                                          PPR00880
      DA(I)=5.8D-07*TK**2.3                                             PPR00890
      COR(I)=DEXP(-CPSI*(TT(I)-TZERO))                                  PPR00900
      B3(I)=RHOV(I)*GORT                                                PPR00910
      B4(I)=RH*DRZDT2(TT(I))-B3(I)*PP(I)/TK                             PPR00920
      IF(I17.EQ.1) WRITE(6,1030) I,CKT(I),COR(I),RHOV(I),RH,GORT,DA(I), PPR00930
     1B3(I),B4(I)                                                       PPR00940
  10  CONTINUE                                                          PPR00950
      IF(I17.EQ.1.AND.I24.NE.3) WRITE(6,1010)                           PPR00960
C*******************************************************************PPR00970
C                                                                       PPR00980
C        IN THE SECOND LOOP, WE COMPUTE VARIABLES THAT ARE DEFINED      PPR00990
C   AT EACH END (NODE) OF EACH ELEMENT. THIS INCLUDES THE ACTUAL        PPR01000
```

```
C   COEFFICIENTS OF THE MASS EQUATION, DENOTED BY C1. C2. C3. C4.        PPR01010
C   AND CK (INSTEAD OF C5). SEE SECTION 4.1.                             PPR01020
C        THE VECTOR INT2 HAS A VALUE OF ONE AT THE NODES AT WHICH        PPR01030
C   THE SPECIAL ITERATIVE METHOD FOR ESTIMATION OF THE STORAGE           PPR01040
C   COEFFICIENT IS TO BE INVOKED. IT IS ACTIVATED WHENEVER A SIGNIFICANTPPR01050
C   CHANGE IN MATRIC HEAD OCCURS AT A GIVEN NODE.                        PPR01060
C                                                                        PPR01070
C*************************************************************************PPR01080
      DO 60 L=1,NL                                                       PPR01090
      DO 60 N=1,2                                                        PPR01100
      I=N+L-1                                                            PPR01110
      J=IS(L)                                                            PPR01120
      IF(I24.EQ.3) GO TO 20                                              PPR01130
      CK(L,N)=CK(L,N)*CKT(I)                                             PPR01140
      B1(L,N)=1.11*POR(J)-XX(L,N)                                        PPR01150
      IF(B1(L,N).LE.0.) B1(L,N)=1.D-14                                   PPR01160
      TORT=B1(L,N)**0.67                                                 PPR01170
      B2=B1(L,N)/RHOL                                                    PPR01180
  20  IF(J9.EQ.0) GO TO 30                                               PPR01190
      IF (DABS((PP(I)-P(I))/P(I)).GT.1.D-04) INT2(I)=1                   PPR01200
      IF (PP(I)-P(I).EQ.0.) INT2(I)=0                                    PPR01210
      IF (INT2(I).EQ.1) DXDP(L,N)=(XX(L,N)-X(L,N))/(PP(I)-P(I))          PPR01220
  30  IF(I24.EQ.3) GO TO 40                                              PPR01230
      B5=(1.-RHOV(I)/RHOL)*DXDP(L,N)*COR(I)                              PPR01240
      B7(L,N)=DA(I)*B2*B3(I)*TORT                                        PPR01250
      C1(L,N)=B5+B2*B3(I)                                                PPR01260
      C2(L,N)=-B5*CPSI*PP(I)+B2*B4(I)                                    PPR01270
      C3(L,N)=CK(L,N)+B7(L,N)                                            PPR01280
      C4(L,N)=DA(I)*B4(I)*EF(L,N)*ZETA/RHOL                              PPR01290
      GO TO 50                                                           PPR01300
  40  C1(L,N)=DXDP(L,N)                                                  PPR01310
      C3(L,N)=CK(L,N)                                                    PPR01320
      GO TO 60                                                           PPR01330
  50  IF(I17.EQ.1) WRITE(6,1020) L,N,XX(L,N).DXDP(L,N).CK(L,N).EF(L,N).  PPR01340
     1TORT,C1(L,N),C2(L,N),C3(L,N),C4(L,N)                               PPR01350
  60  CONTINUE                                                           PPR01360
      IF(I17.EQ.1.AND.I24.EQ.3) WRITE(6.1050) ((L.N.XX(L.N).DXDP(L.N).   PPR01370
     1CK(L,N),N=1,2),L=1,NL)                                             PPR01380
      RETURN                                                             PPR01390
C*************************************************************************PPR01400
C                                                                        PPR01410
C   FORMAT STATEMENTS                                                    PPR01420
C                                                                        PPR01430
C*************************************************************************PPR01440
1010  FORMAT(//2X,'L',2X,'N',3X,'XX',9X.'DXDP'.12X.'CK'.8X.'EF'.         PPR01450
     13X,'TORT',10X,                                                     PPR01460
     2'C1',13X,'C2',13X,'C3',13X,'C4')                                   PPR01470
1020  FORMAT(1X,I2,I3,2X,F3.2,1X,2D15.4.2X.F5.2,1X.                      PPR01480
     1F5.2,1X,4D15.4)                                                    PPR01490
1030  FORMAT(1X,I3,2(2X,F6.3),6D15.4)                                    PPR01500
```

```
1040  FORMAT(//3X,'I',4X,'CKT',5X,'COR',9X,'RHOV',12X,'RH',12X,'GORT',   PPR01510
     112X,'DA',13X,'B3',13X,'B4')                                        PPR01520
1050  FORMAT(//2X,'L',2X,'N',3X,'XX',9X,'DXDP',12X,'CK',                 PPR01530
     1/(1X,I2,I3,2X,F3.2,1X,2D15.4))                                     PPR01540
      END                                                                PPR01550
C*******************************************************************PPR01560
```

```
C*****************************************************************TPR00010
C                                                                 TPR00020
      SUBROUTINE TPRAM2                                           TPR00030
C                                                                 TPR00040
C*****************************************************************TPR00050
C                                                                 TPR00060
C        THIS SUBROUTINE CALCULATES THE COEFFICIENTS OF THE HEAT  TPR00070
C   CONSERVATION EQUATION.                                        TPR00080
C                                                                 TPR00090
C*****************************************************************TPR00100
C                                                                 TPR00110
C   THE FOLLOWING VARIABLES, NOT DEFINED IN THE MAIN PROGRAM OR IN TPR00120
C   SUBROUTINE PPRAM2, ARE USED IN THIS ROUTINE.                  TPR00130
C                                                                 TPR00140
C*****************************************************************TPR00150
C   NAME           DESCRIPTION                                    TPR00160
C*****************************************************************TPR00170
C   B2(NN)         VOLUMETRIC SENSIBLE HEAT CONTENT OF LIQUID WATER TPR00180
C   B5(NN)         HEAT OF WETTING TIMES DENSITY OF WATER         TPR00190
C   B6             A PART OF THE HEAT STORAGE COEFFICIENTS        TPR00200
C   EL(NN)         THE LATENT HEAT OF EVAPORATION OF WATER VAPOR  TPR00210
C   ELCP(NN)       THE SUM OF EL AND THE SENSIBLE HEAT OF LIQUID WATER TPR00220
C   HC             THE VOLUMETRIC HEAT CAPACITY OF BULK POROUS MEDIUM TPR00230
C   TCON           THE EFFECTIVE THERMAL CONDUCTIVITY OF THE MEDIUM TPR00240
C   TK             ABSOLUTE (KELVIN) TEMPERATURE                  TPR00250
C*****************************************************************TPR00260
C                                                                 TPR00270
C   COMMON AND DIMENSION STATEMENTS                               TPR00280
C                                                                 TPR00290
C*****************************************************************TPR00300
      IMPLICIT REAL*8(A-H,O-Z)                                    TPR00310
      COMMON C(46,2),D(46,2)                                      TPR00320
      COMMON B1(45,2),B7(45,2),CK(45,2),C1(45,2),C2(45,2),C3(45,2), TPR00330
     *C4(45,2),C5(45,2),DXDP(45,2),EF(45,2),X(45,2),XX(45,2),XWRE(45,2) TPR00340
      COMMON B3(46),B4(46),COR(46),                               TPR00350
     *DELTA(46),P(46),                                            TPR00360
     *PFU(46),PP(46),PPOLD(46),RHOV(46),T(46),                    TPR00370
     *TT(46),TTOLD(46),Z(46)                                      TPR00380
      COMMON XOLD(45)                                             TPR00390
      COMMON POR(44)                                              TPR00400
      COMMON BCP,BCT,CPLGE,CPSI,DELT,DELTO,DLOGE,DLN10,DLT,DURTN,PERR, TPR00410
     *PN,RHOL,ST,STI,SX,SXI,TEND,TERR,TFLUX,THETA,TIME,TSAVE1,TSAVE2, TPR00420
     *TZERO,XFLUX,ZETA                                            TPR00430
      COMMON IH(46),IS(46)                                        TPR00440
      COMMON I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15,I16,I17,TPR00450
     *I18,I19,I20,I21,I22,I23,I24,I25,J1,J2,J3,J4,J5,J6,J7,J8,J9,ITDES, TPR00460
     *KIT,KK,KT,NBCP,NBCT,NL,NN,NPER,NS,NSTEPS                    TPR00470
      DIMENSION B2(46),B5(46),EL(46),ELCP(46)                     TPR00480
C*****************************************************************TPR00490
C                                                                 TPR00500
```

```
C   JUST AS IN PPRAM2, THERE ARE TWO LOOPS USED IN COMPUTATIONS. THE    TPR00510
C   FIRST COMPUTES NODAL QUANTITIES THAT ARE THE SAME APPROACHED FROM   TPR00520
C   EITHER SIDE OF THE NODE. THE SECOND LOOP COMPUTES THOSE THAT ARE    TPR00530
C   NOT, AND SETS UP THE ACTUAL COEFFICIENTS.                           TPR00540
C                                                                       TPR00550
C*******************************************************************TPR00560
C                                                                       TPR00570
C   FIRST LOOP                                                          TPR00580
C                                                                       TPR00590
C*******************************************************************TPR00600
      IF(I20.EQ.1) WRITE(6,1010)                                        TPR00610
      DO 10 I=1,NN                                                      TPR00620
      TK=TT(I)+273.16                                                   TPR00630
      EL(I)=597.3-0.552*(TT(I)-TZERO)                                   TPR00640
      ELCP(I)=EL(I)+TT(I)-TZERO                                         TPR00650
      W=-981.*PP(I)*(-CPSI*TK+1.)/4.18D+07                              TPR00660
      W=0.0                                                             TPR00670
      B2(I)=RHOL*(TT(I)-TZERO)                                          TPR00680
 10   B5(I)=RHOL*W                                                      TPR00690
C*******************************************************************TPR00700
C                                                                       TPR00710
C   SECOND LOOP                                                         TPR00720
C                                                                       TPR00730
C*******************************************************************TPR00740
      DO 20 L=1,NL                                                      TPR00750
      DO 20 N=1,2                                                       TPR00760
      I=L+N-1                                                           TPR00770
      HC=0.48+RHOL*XX(L,N)+0.448*RHOV(I)*B1(L,N)                        TPR00780
      TCON=(1.+XX(L,N))*2.4D-03                                         TPR00790
      B6=(B2(I)-B5(I)-ELCP(I)*RHOV(I))*COR(I)*DXDP(L,N)                 TPR00800
      C1(L,N)=ELCP(I)*B1(L,N)*B3(I)+B6                                  TPR00810
      C2(L,N)=HC+ELCP(I)*B1(L,N)*B4(I)-B6*CPSI*PP(I)                    TPR00820
      C3(L,N)=RHOL*EL(I)*B7(L,N)+B2(I)*C3(L,N)                          TPR00830
      C4(L,N)=TCON+B2(I)*C4(L,N)                                        TPR00840
      C5(L,N)=B2(I)*CK(L,N)                                             TPR00850
 20   IF(I20.EQ.1) WRITE(6,1020) L,N,HC,TCON,C1(L,N),C2(L,N),C3(L,N),   TPR00860
     1C4(L,N),C5(L,N)                                                   TPR00870
      TSAVE1=TT(1)                                                      TPR00880
      TSAVE2=TT(NN)                                                     TPR00890
      RETURN                                                            TPR00900
1010  FORMAT(//5X,'L',2X,'N',9X,'HC',12X,'TCON',12X,'C1',13X,'C2',13X,'CTPR00910
     13',13X,'C4',13X,'C5')                                             TPR00920
1020  FORMAT(3X,2I3,7D15.4)                                             TPR00930
      END                                                               TPR00940
C*******************************************************************TPR00950
```

```
C*******************************************************************MAT00010
C                                                                      MAT00020
      SUBROUTINE MAT2(A,B,C,D,X,C1,C2,C3,C4,C5,DELTA,M,NN,NL,IK,I24,J2,JMAT00030
     13,J4,J8)                                                        MAT00040
C                                                                      MAT00050
C*******************************************************************MAT00060
C                                                                      MAT00070
C        SUBROUTINE MAT2 ASSEMBLES THE GLOBAL COEFFICIENT MATRICES OF  MAT00080
C   THE GALERKIN EXPRESSIONS FOR THE CONSERVATION EQUATIONS. IT USES   MAT00090
C   THE FUNCTIONAL COEFFICIENT SCHEME TO EVALUATE THE MATRIX ELEMENTS. MAT00100
C   AND EMPLOYS ONE OF SEVERAL MASS WEIGHTING SCHEMES. (SECTION 4.6    MAT00110
C   DESCRIBES ONE MASS LUMPING PROCEDURE.)                             MAT00120
C                                                                      MAT00130
C*******************************************************************MAT00140
C                                                                      MAT00150
      IMPLICIT REAL*8(A-H,O-Z)                                         MAT00160
      DIMENSION A(46,2),B(46,2),C(46,2),D(46,2),C1(45,2),C2(45,2),     MAT00170
     *C3(45,2),C4(45,2),C5(45,2),DELTA(46),X(46)                       MAT00180
C                                                                      MAT00190
C*******************************************************************MAT00200
C                                                                      MAT00210
C   INITIALIZE THE MATRICES.                                           MAT00220
C                                                                      MAT00230
C*******************************************************************MAT00240
      DO 20 I=1,NN                                                     MAT00250
      DO 10 J=1,2                                                      MAT00260
      A(I,J)=0.0                                                       MAT00270
      B(I,J)=0.0                                                       MAT00280
      C(I,J)=0.0                                                       MAT00290
   10 D(I,J)=0.0                                                       MAT00300
   20 X(I)=0.0                                                         MAT00310
C*******************************************************************MAT00320
C                                                                      MAT00330
C   EVALUATE MATRICES ACCORDING TO THE SPECIFIED OPTIONS.              MAT00340
C                                                                      MAT00350
C*******************************************************************MAT00360
      IF(J3.EQ.1) GO TO 40                                             MAT00370
      IF(J3.EQ.2) GO TO 60                                             MAT00380
      DO 30 L=1,NL                                                     MAT00390
      A(L,1)=A(L,1)+(C1(L,1)+C1(L,2)/3.)*DELTA(L)/4.                   MAT00400
      A(L,2)=A(L,2)+(C1(L,1)+C1(L,2))*DELTA(L)/12.                     MAT00410
   30 A(L+1,1)=A(L+1,1)+(C1(L,1)/3.+C1(L,2))*DELTA(L)/4.               MAT00420
      GO TO 80                                                         MAT00430
   40 DO 50 L=1,NL                                                     MAT00440
      A(L,1)=A(L,1)+(C1(L,1)+C1(L,2)/2.)*DELTA(L)/3.                   MAT00450
   50 A(L+1,1)=A(L+1,1)+(C1(L,1)/2.+C1(L,2))*DELTA(L)/3.               MAT00460
      GO TO 80                                                         MAT00470
   60 DO 70 L=1,NL                                                     MAT00480
      A(L,1)=A(L,1)+C1(L,1)*DELTA(L)/2.                                MAT00490
   70 A(L+1,1)=A(L+1,1)+C1(L,2)*DELTA(L)/2.                            MAT00500
```

```
 80   IF(I24.EQ.3) GO TO 140                                          MAT00510
      IF(J4.EQ.1) GO TO 100                                           MAT00520
      IF(J4.EQ.2) GO TO 120                                           MAT00530
      DO 90 L=1,NL                                                    MAT00540
      B(L,1)=B(L,1)+(C2(L,1)+C2(L,2)/3.)*DELTA(L)/4.                  MAT00550
      B(L,2)=B(L,2)+(C2(L,1)+C2(L,2))*DELTA(L)/12.                    MAT00560
 90   B(L+1,1)=B(L+1,1)+(C2(L,1)/3.+C2(L.2))*DELTA(L)/4.              MAT00570
      GO TO 140                                                       MAT00580
100   DO 110 L=1,NL                                                   MAT00590
      B(L,1)=B(L,1)+(C2(L,1)+C2(L,2)/2.)*DELTA(L)/3.                  MAT00600
110   B(L+1,1)=B(L+1,1)+(C2(L,1)/2.+C2(L.2))*DELTA(L)/3.              MAT00610
      GO TO 140                                                       MAT00620
120   DO 130 L=1,NL                                                   MAT00630
      B(L,1)=B(L,1)+C2(L,1)*DELTA(L)/2.                               MAT00640
130   B(L+1,1)=C2(L,2)*DELTA(L)/2.                                    MAT00650
140   DO 170 L=1,NL                                                   MAT00660
      IF(J8.EQ.2) GO TO 150                                           MAT00670
      CE=(C3(L,1)+C3(L,2))/2./DELTA(L)                                MAT00680
      GO TO 160                                                       MAT00690
150   CE=DSQRT(C3(L,1)*C3(L,2))/DELTA(L)                              MAT00700
160   C(L,1)=C(L,1)+CE                                                MAT00710
      C(L,2)=C(L,2)-CE                                                MAT00720
170   C(L+1,1)=C(L+1,1)+CE                                            MAT00730
      IF(I24.EQ.3) GO TO 190                                          MAT00740
      DO 180 L=1,NL                                                   MAT00750
      DE=(C4(L,1)+C4(L,2))/2./DELTA(L)                                MAT00760
      D(L,1)=D(L,1)+DE                                                MAT00770
      D(L,2)=D(L,2)-DE                                                MAT00780
180   D(L+1,1)=D(L+1,1)+DE                                            MAT00790
190   DO 220 L=1,NL                                                   MAT00800
      IF(J8.EQ.2) GO TO 200                                           MAT00810
      FE=(C5(L,1)+C5(L,2))/2.                                         MAT00820
      GO TO 210                                                       MAT00830
200   FE=DSQRT(C5(L,1)*C5(L,2))                                       MAT00840
210   X(L)=X(L)-FE                                                    MAT00850
220   X(L+1)=X(L+1)+FE                                                MAT00860
      IF(M.EQ.2) GO TO 230                                            MAT00870
      IF(IK.EQ.1) WRITE(6,1010) (I,(A(I,J),J=1,2),(B(I,J),J=1,2).     MAT00880
     *(C(I,J),J=1,2),(D(I,J),J=1,2),X(I),I=1,NN)                      MAT00890
      RETURN                                                          MAT00900
230   IF(IK.EQ.1) WRITE(6,1020) (I,(A(I,J),J=1,2),(B(I,J),J=1,2).     MAT00910
     *(C(I,J),J=1,2),(D(I,J),J=1,2),X(I),I=1,NN)                      MAT00920
      RETURN                                                          MAT00930
C*****************************************************************   MAT00940
C                                                                     MAT00950
C   FORMAT STATEMENTS                                                 MAT00960
C                                                                     MAT00970
C*****************************************************************   MAT00980
1010  FORMAT(//10X,'MOISTURE MATRICES'/10X,17(1H-)//5X,'I',6X,'A(I,1)',4MAT00990
     *X,'A(I,2)',5X,'B(I,1)',4X,'B(I,2)',5X,'C(I,1)',4X,'C(I,2)',5X,'D(IMAT01000
```

```
     *,1)',4X,'D(I,2)',5X,'F(I)'/(/3X,I4,3X,4(2D10.2,1X),D10.2))          MAT01010
1020  FORMAT(//10X,'TEMPERATURE MATRICES'/10X,20(1H-)//5X,'I',6X,'A(I,1)MAT01020
     1',4X,'A(I,2)',5X,'B(I,1)',4X,'B(I,2)',5X,'C(I,1)',4X,'C(I,2)',5X,'MAT01030
     2D(I,1)',4X,'D(I,2)',6X,'F(I)'/(/3X,I4,3X,4(2D10.2,1X),D10.2))       MAT01040
      END                                                                 MAT01050
C*******************************************************************************MAT01060
```

```
C*************************************************************************EQN00010
C                                                                         EQN00020
      SUBROUTINE EQN2(A,B,C,D,P,PP,TT,T,DELT,M,NN,NL,I24,J2)              EQN00030
C                                                                         EQN00040
C*************************************************************************EQN00050
C                                                                         EQN00060
C        THIS SUBROUTINE PERFORMS THE FINITE DIFFERENCE OF THE            EQN00070
C   TIME DERIVATIVES AND SETS UP THE MATRIX EQUATION FOR THE              EQN00080
C   MATRIX SOLVER. THESE STEPS ARE DESCRIBED IN SECTION 4.3.              EQN00090
C   WHEN THIS ROUTINE IS CALLED FROM THE MASS EQUATION LOOP, THE          EQN00100
C   STATE VARIABLES HAVE THE SAME NAMES INSIDE THIS SUBROUTINE            EQN00110
C   AS THEY DO IN THE MAIN COMMON BLOCK. THE NAMES ARE REVERSED.          EQN00120
C   HOWEVER, WHEN THIS ROUTINE IS CALLED FROM THE ENERGY LOOP.            EQN00130
C                                                                         EQN00140
C*************************************************************************EQN00150
      IMPLICIT REAL*8(A-H,O-Z)                                            EQN00160
      DIMENSION A(46,2),B(46,2),C(46,2),D(46,2),P(46),PP(46),             EQN00170
     1TT(46),T(46)                                                        EQN00180
C*************************************************************************EQN00190
C                                                                         EQN00200
C   THE MATRIX "SAVE" WILL REMEMBER THE COEFFICIENTS OF THE END NODES     EQN00210
C   FOR LATER USE IN THE EVALUATION OF END FLUXES                         EQN00220
C                                                                         EQN00230
      COMMON /MASBAL/ SAVE(2,3,2)                                         EQN00240
C*************************************************************************EQN00250
C                                                                         EQN00260
C   FIRST COMPUTE THE RIGHT-HAND-SIDE VECTOR AND STORE IT IN PP.          EQN00270
C                                                                         EQN00280
C*************************************************************************EQN00290
      IF(I24.EQ.3) GO TO 20                                               EQN00300
      CF2=B(1,1)/DELT                                                     EQN00310
      CF3=B(1,2)/DELT                                                     EQN00320
      PP(1)=-PP(1)+(A(1,1)/DELT)*P(1)+(A(1,2)/DELT)*P(2)                  EQN00330
     1-(CF2+D(1,1))*TT(1)-(CF3+D(1,2))*TT(2)+CF2*T(1)+CF3*T(2)            EQN00340
      DO 10 I=2,NL                                                        EQN00350
      CF1=B(I-1,2)/DELT                                                   EQN00360
      CF2=B(I,1)/DELT                                                     EQN00370
      CF3=B(I,2)/DELT                                                     EQN00380
   10 PP(I)=-PP(I)+(A(I-1,2)/DELT)*P(I-1)                                 EQN00390
     *           +(A(I,1)/DELT)*P(I)                                      EQN00400
     *           +(A(I,2)/DELT)*P(I+1)                                    EQN00410
     *           -(CF1+D(I-1,2))*TT(I-1)                                  EQN00420
     *           -(CF2+D(I,1))*TT(I)                                      EQN00430
     *           -(CF3+D(I,2))*TT(I+1)                                    EQN00440
     *           +CF1*T(I-1)+CF2*T(I)+CF3*T(I+1)                          EQN00450
      CF1=B(NN-1,2)/DELT                                                  EQN00460
      CF2=B(NN,1)/DELT                                                    EQN00470
      PP(NN)=-PP(NN)+(A(NN-1,2)/DELT)*P(NN-1)                             EQN00480
     *            +(A(NN,1)/DELT)*P(NN)                                   EQN00490
     *            -(CF1+D(NN-1,2))*TT(NN-1)                               EQN00500
```

```
     *              -(CF2+D(NN,1))*TT(NN)                                    EQN00510
     *              +CF1*T(NN-1)+CF2*T(NN)                                   EQN00520
      GO TO 40                                                               EQN00530
  20  PP(1)=-PP(1)+(A(1,1)*P(1) + A(1,2)*P(2))/DELT                          EQN00540
      DO 30 I=2,NL                                                           EQN00550
  30  PP(I)=-PP(I) + (A(I-1,2)*P(I-1)+A(I,1)*P(I)+A(I,2)*P(I+1))/DELT        EQN00560
      PP(NN)=-PP(NN) + (A(NL,2)*P(NL)+A(NN,1)*P(NN))/DELT                    EQN00570
C*************************************************************************EQN00580
C                                                                            EQN00590
C   NOW COMPUTE THE LEFT-HAND-SIDE MATRIX.                                   EQN00600
C                                                                            EQN00610
C*************************************************************************EQN00620
  40  DO 50 I=1,NN                                                           EQN00630
      DO 50 J=1,2                                                            EQN00640
  50  C(I,J)=A(I,J)/DELT+C(I,J)                                              EQN00650
C*************************************************************************EQN00660
C                                                                            EQN00670
C   SAVE THE EQUATIONS FOR THE TWO END NODES FOR FUTURE EVALUATION           EQN00680
C   OF THE MASS AND HEAT FLUXES INTO AND OUT OF THE SOIL COLUMN.             EQN00690
C                                                                            EQN00700
C*************************************************************************EQN00710
      SAVE(1,1,M)=PP(1)                                                      EQN00720
      SAVE(1,2,M)=C(1,1)                                                     EQN00730
      SAVE(1,3,M)=C(1,2)                                                     EQN00740
      SAVE(2,1,M)=PP(NN)                                                     EQN00750
      SAVE(2,2,M)=C(NN-1,2)                                                  EQN00760
      SAVE(2,3,M)=C(NN,1)                                                    EQN00770
      RETURN                                                                 EQN00780
      END                                                                    EQN00790
C*************************************************************************EQN00800
```

```
C*******************************************************************PBC00010
C                                                                     PBC00020
      SUBROUTINE PBC2                                                 PBC00030
C                                                                     PBC00040
C*******************************************************************PBC00050
C                                                                     PBC00060
C        THIS SUBROUTINE INCORPORATES THE BOUNDARY CONDITIONS THAT    PBC00070
C   WERE SPECIFIED IN NEWPR2 ACCORDING TO THE PROCEDURES DESCRIBED    PBC00080
C   IN SECTION 4.2.4. PBC2 TAKES CARE OF MASS BOUNDARY CONDITIONS,    PBC00090
C   WHILE ENTRY TBC2 IS FOR THE HEAT BOUNDARY CONDITIONS.             PBC00100
C                                                                     PBC00110
C*******************************************************************PBC00120
C                                                                     PBC00130
C   COMMON AND DIMENSION STATEMENTS                                   PBC00140
C                                                                     PBC00150
C*******************************************************************PBC00160
      IMPLICIT REAL*8(A-H,O-Z)                                        PBC00170
      COMMON C(46,2),D(46,2)                                          PBC00180
      COMMON B1(45,2),B7(45,2),CK(45,2),C1(45,2),C2(45,2),C3(45,2),   PBC00190
     *C4(45,2),C5(45,2),DXDP(45,2),EF(45,2),X(45,2),XX(45,2),XWRE(45,2) PBC00200
      COMMON B3(46),B4(46),COR(46),                                   PBC00210
     *DELTA(46),P(46),                                                PBC00220
     *PFU(46),PP(46),PPOLD(46),RHOV(46),T(46),                        PBC00230
     *TT(46),TTOLD(46),Z(46)                                          PBC00240
      COMMON XOLD(45)                                                 PBC00250
      COMMON POR(44)                                                  PBC00260
      COMMON BCP,BCT,CPLGE,CPSI,DELT,DELTO,DLOGE,DLN10,DLT,DURTN,PERR, PBC00270
     *PN,RHOL,ST,STI,SX,SXI,TEND,TERR,TFLUX,THETA,TIME,TSAVE1,TSAVE2,  PBC00280
     *TZERO,XFLUX,ZETA                                                PBC00290
      COMMON IH(46),IS(46)                                            PBC00300
      COMMON I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15,I16,I17,PBC00310
     *I18,I19,I20,I21,I22,I23,I24,I25,J1,J2,J3,J4,J5,J6,J7,J8,J9,ITDES, PBC00320
     *KIT,KK,KT,NBCP,NBCT,NL,NN,NPER,NS,NSTEPS                        PBC00330
      COMMON /MASBAL/ SAVE(2,3,2)                                     PBC00340
C*******************************************************************PBC00350
C                                                                     PBC00360
C   APPLY A TYPE I BOUNDARY CONDITION ON MASS (MATRIC HEAD) AT THE    PBC00370
C   BOTTOM OF THE COLUMN.                                             PBC00380
C                                                                     PBC00390
C*******************************************************************PBC00400
      PP(1)=P(1)                                                      PBC00410
      C(1,1)=1.0                                                      PBC00420
      PP(2)=PP(2)-C(1,2)*P(1)                                         PBC00430
      C(1,2)=0.0                                                      PBC00440
C*******************************************************************PBC00450
C                                                                     PBC00460
C   APPLY THE BOUNDARY CONDITION SPECIFIED BY NBCP AND BCP.           PBC00470
C                                                                     PBC00480
C*******************************************************************PBC00490
      IF(NBCP.EQ.2) GO TO 10                                          PBC00500
```

```
C*****************************************************************PBC00510
C                                                                 PBC00520
C   TYPE I BOUNDARY CONDITION AT LAND SURFACE.                    PBC00530
C                                                                 PBC00540
      PP(NN)=PN                                                   PBC00550
      C(NN,1)=1.0                                                 PBC00560
      PP(NN-1)=PP(NN-1)-C(NN-1,2)*PN                              PBC00570
      C(NN-1,2)=0.0                                               PBC00580
      GO TO 20                                                    PBC00590
C                                                                 PBC00600
C*****************************************************************PBC00610
C                                                                 PBC00620
C   MASS FLUX BOUNDARY CONDITION AT LAND SURFACE.                 PBC00630
C                                                                 PBC00640
  10  PP(NN)=PP(NN)-BCP                                           PBC00650
C                                                                 PBC00660
C*****************************************************************PBC00670
  20  IF(I19.EQ.1) WRITE(6,1010) (I,(C(I,J),J=1,2),PP(I),I=1,NN)  PBC00680
      RETURN                                                      PBC00690
C*****************************************************************PBC00700
C*****************************************************************PBC00710
C                                                                 PBC00720
      ENTRY TBC2                                                  PBC00730
C                                                                 PBC00740
C*****************************************************************PBC00750
C                                                                 PBC00760
C        APPLY BOUNDARY CONDITIONS ON HEAT. THE BOTTOM CONDITION IS THATPBC00770
C   HEAT LEAVES OR ENTERS BY ADVECTION ONLY. QPB IS THE MOISTURE FLUX ATPBC00780
C   THE BOTTOM OF THE COLUMN. IF A FLUX BOUNDARY CONDITION ON HEAT    PBC00790
C   IS USED AT THE SURFACE, BCT IS SPECIFIED AS TOTAL HEAT FLUX.      PBC00800
C   INCLUDING A POSSIBLE ADVECTION TERM.                          PBC00810
C                                                                 PBC00820
C*****************************************************************PBC00830
C                                                                 PBC00840
C   HEAT FLUX B.C. AT BOTTOM OF COLUMN.                           PBC00850
C                                                                 PBC00860
      QPB=-SAVE(1,1,1)+SAVE(1,2,1)*PP(1)+SAVE(1,3,1)*PP(2)        PBC00870
      TT(1)=TT(1)+QPB*RHOL*(TSAVE1-TZERO)                         PBC00880
C                                                                 PBC00890
C*****************************************************************PBC00900
      IF(NBCT.EQ.2) GO TO 30                                      PBC00910
C*****************************************************************PBC00920
C                                                                 PBC00930
C   TYPE ONE B.C. AT SURFACE.                                     PBC00940
C                                                                 PBC00950
C*****************************************************************PBC00960
      TT(NN)=BCT                                                  PBC00970
      D(NN,1)=1.0                                                 PBC00980
      TT(NN-1)=TT(NN-1)-D(NN-1,2)*BCT                             PBC00990
      D(NN-1,2)=0.0                                               PBC01000
```

```
      GO TO 40                                                          PBC01010
C                                                                       PBC01020
C***********************************************************************PBC01030
C                                                                       PBC01040
C   FLUX B.C. AT SURFACE.                                               PBC01050
C                                                                       PBC01060
  30  TT(NN)=TT(NN)-BCT                                                 PBC01070
C                                                                       PBC01080
C***********************************************************************PBC01090
  40  IF(I22.EQ.1) WRITE(6,1020) (I,(D(I,J),J=1,2),TT(I),I=1,NN)        PBC01100
      RETURN                                                            PBC01110
C***********************************************************************PBC01120
C                                                                       PBC01130
C   FORMAT STATEMENTS                                                   PBC01140
C                                                                       PBC01150
C***********************************************************************PBC01160
1010  FORMAT(//10X,'MOISTURE EQUATION'/10X,17(1H-)//20X,'I',10X,'LHS(I,IPBC01170
     1)',5X,'LHS(I,I+1)=LHS(I+1,I)',10X,'RHS(I)'/(/19X,I3,8X,D10.2,8X,  PBC01180
     2D10.2,14X,D10.2))                                                 PBC01190
1020  FORMAT(//10X,'TEMPERATURE EQUATION'/10X,20(1H-)//20X,'I',10X,'LHS(PBC01200
     1I,I)',5X,'LHS(I,I+1)=LHS(I+1,I)',10X,'RHS(I)'/(/19X,I3,8X,D10.2,8XPBC01210
     2,D10.2,14X,D10.2))                                                PBC01220
      END                                                               PBC01230
C***********************************************************************PBC01240
```

```
C*************************************************************************SOL00010
C                                                                         SOL00020
      SUBROUTINE SOLVE2(B,C,NN)                                           SOL00030
C                                                                         SOL00040
C*************************************************************************SOL00050
C                                                                         SOL00060
C        THIS ALGORITHM FOR THE SOLUTION OF A TRIDIAGONAL MATRIX          SOL00070
C   EQUATION IS REFERENCED IN SECTION 4.4.                                SOL00080
C                                                                         SOL00090
C*************************************************************************SOL00100
      IMPLICIT REAL*8(A-H,O-Z)                                            SOL00110
      DIMENSION B(46,2),C(46)                                             SOL00120
C*************************************************************************SOL00130
C                                                                         SOL00140
C   SOLUTION OF TRIDIAGONAL MATRIX EQUATION USING THE THOMAS ALGORITHM.   SOL00150
C                                                                         SOL00160
C*************************************************************************SOL00170
      C(1)=C(1)/B(1,1)                                                    SOL00180
      DO 10 I=2,NN                                                        SOL00190
      B(I,1)=B(I,1)-B(I-1,2)*B(I-1,2)/B(I-1,1)                            SOL00200
  10  C(I)=(C(I)-B(I-1,2)*C(I-1))/B(I,1)                                  SOL00210
      NL=NN-1                                                             SOL00220
      DO 20 II=1,NL                                                       SOL00230
      I=NN-II                                                             SOL00240
  20  C(I)=C(I)-B(I,2)*C(I+1)/B(I,1)                                      SOL00250
      RETURN                                                              SOL00260
      END                                                                 SOL00270
C*************************************************************************SOL00280
```

```
C*************************************************************************CHK00010
C                                                                         CHK00020
      SUBROUTINE CHK2(X,Y,NN,ERR,J,K)                                     CHK00030
C                                                                         CHK00040
C*************************************************************************CHK00050
C                                                                         CHK00060
C        THIS SUBROUTINE CHECKS FOR CONVERGENCE OF THE MASS (K=1)         CHK00070
C   AND HEAT (K=2) SOLUTIONS.                                             CHK00080
C                                                                         CHK00090
C*************************************************************************CHK00100
      IMPLICIT REAL*8(A-H,O-Z)                                            CHK00110
      DIMENSION X(NN),Y(NN)                                               CHK00120
C*************************************************************************CHK00130
      J=1                                                                 CHK00140
      IF(K.EQ.1) GO TO 20                                                 CHK00150
C*************************************************************************CHK00160
C                                                                         CHK00170
C   THE HEAT SOLUTION HAS CONVERGED WHEN COMPUTED TEMPERATURES FOR        CHK00180
C   TWO SUCCESSIVE ITERATIONS DIFFER BY LESS THAN ERR DEGREES.            CHK00190
C                                                                         CHK00200
C*************************************************************************CHK00210
      DO 10 I=1,NN                                                        CHK00220
      IF(DABS(X(I)-Y(I)).LT.ERR) GO TO 10                                 CHK00230
      J=0                                                                 CHK00240
      RETURN                                                              CHK00250
  10  CONTINUE                                                            CHK00260
      RETURN                                                              CHK00270
C*************************************************************************CHK00280
C                                                                         CHK00290
C   THE MASS EQUATION HAS CONVERGED WHEN SUCCESSIVE MATRIC POTENTIALS     CHK00300
C   HAVE A RELATIVE DIFFERENCE LESS THAN ERR.                             CHK00310
C                                                                         CHK00320
C*************************************************************************CHK00330
  20  DO 30 I=1,NN                                                        CHK00340
      IF(X(I).EQ.0) GO TO 30                                              CHK00350
      IF(DABS((X(I)-Y(I))/X(I)).LT.ERR) GO TO 30                          CHK00360
      J=0                                                                 CHK00370
      RETURN                                                              CHK00380
  30  CONTINUE                                                            CHK00390
      RETURN                                                              CHK00400
      END                                                                 CHK00410
C*************************************************************************CHK00420
```

```
C*************************************************************************BAL00010
C                                                                         BAL00020
      SUBROUTINE BAL2(M)                                                  BAL00030
C                                                                         BAL00040
C*************************************************************************BAL00050
C                                                                         BAL00060
C        THIS SUBROUTINE, IF CALLED FROM THE MAIN PROGRAM. CALCULATES     BAL00070
C   THE MASS AND ENERGY BALANCE ERRORS - CUMULATIVE AND FOR THE LAST      BAL00080
C   TIME STEP.                                                            BAL00090
C                                                                         BAL00100
C*************************************************************************BAL00110
C                                                                         BAL00120
C   COMMON AND DIMENSION STATEMENTS                                       BAL00130
C                                                                         BAL00140
C*************************************************************************BAL00150
      IMPLICIT REAL*8(A-H,O-Z)                                            BAL00160
      COMMON C(46,2),D(46,2)                                              BAL00170
      COMMON B1(45,2),B7(45,2),CK(45,2),C1(45,2),C2(45,2),C3(45,2),       BAL00180
     *C4(45,2),C5(45,2),DXDP(45,2),EF(45,2),X(45,2),XX(45,2),XWRE(45,2)   BAL00190
      COMMON B3(46),B4(46),COR(46),                                       BAL00200
     *DELTA(46),P(46),                                                    BAL00210
     *PFU(46),PP(46),PPOLD(46),RHOV(46),T(46),                            BAL00220
     *TT(46),TTOLD(46),Z(46)                                              BAL00230
      COMMON XOLD(45)                                                     BAL00240
      COMMON POR(44)                                                      BAL00250
      COMMON BCP,BCT,CPLGE,CPSI,DELT,DELTO,DLOGE,DLN10,DLT,DURTN,PERR,    BAL00260
     *PN,RHOL,ST,STI,SX,SXI,TEND,TERR,TFLUX,THETA,TIME,TSAVE1,TSAVE2,     BAL00270
     *TZERO,XFLUX,ZETA                                                    BAL00280
      COMMON IH(46),IS(46)                                                BAL00290
      COMMON I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15,I16,I17,  BAL00300
     *I18,I19,I20,I21,I22,I23,I24,I25,J1,J2,J3,J4,J5,J6,J7,J8,J9,ITDES,   BAL00310
     *KIT,KK,KT,NBCP,NBCT,NL,NN,NPER,NS,NSTEPS                            BAL00320
      COMMON /MASBAL/ SAVE(2,3,2)                                         BAL00330
C*************************************************************************BAL00340
C                                                                         BAL00350
C   CALCULATE TOTAL STORAGE OF WATER AND ENERGY CURRENTLY IN THE          BAL00360
C   COLUMN. FOR EACH ELEMENT, FIND THE STORAGE (VOLUMETRIC) AT EACH       BAL00370
C   END NODE. THEN INTEGRATE OVER THE ELEMENT. ASSUMING STORAGE TO VARY   BAL00380
C   LINEARLY INSIDE AN ELEMENT.                                           BAL00390
C                                                                         BAL00400
C*************************************************************************BAL00410
      SOLD=SX                                                             BAL00420
      STOLD=ST                                                            BAL00430
      SX=0.                                                               BAL00440
      ST=0.                                                               BAL00450
      IF(I24.EQ.3) GO TO 20                                               BAL00460
      DO 10 L=1,NL                                                        BAL00470
      J=IS(L)                                                             BAL00480
      X1=X(L,1)                                                           BAL00490
      X2=X(L,2)                                                           BAL00500
```

```
      T1=TT(L)                                                          BAL00510
      T2=TT(L+1)                                                        BAL00520
      P1=PP(L)                                                          BAL00530
      P2=PP(L+1)                                                        BAL00540
      RHOV1=RHOZZ2(T1)*DEXP(P1*2.13D-04/(T1+273.16))                    BAL00550
      RHOV2=RHOZZ2(T2)*DEXP(P2*2.13D-04/(T2+273.16))                    BAL00560
      XV1=(1.11*POR(J)-X1)*RHOV1/RHOL                                   BAL00570
      XV2=(1.11*POR(J)-X2)*RHOV2/RHOL                                   BAL00580
      ST1=(597.3+0.448*(T1-TZERO))*RHOL*XV1+(0.48+RHOL*X1)*(T1-TZERO)   BAL00590
      ST2=(597.3+0.448*(T2-TZERO))*RHOL*XV2+(0.48+RHOL*X2)*(T2-TZERO)   BAL00600
      SX=SX+(X1+X2+XV1+XV2)*DELTA(L)/2.                                 BAL00610
   10 ST=ST+(ST1+ST2)*DELTA(L)/2.                                       BAL00620
      GO TO 40                                                          BAL00630
   20 DO 30 L=1,NL                                                      BAL00640
   30 SX=SX + (X(L,1)+X(L,2))*DELTA(L)/2.                               BAL00650
   40 IF(M.EQ.1) RETURN                                                 BAL00660
C*********************************************************************BAL00670
C                                                                       BAL00680
C   USING THE MATRIX "SAVE", CALCULATE END FLUXES OF MASS.              BAL00690
C   FIND NET FLUX, TOTAL INFLOW FOR TIME STEP. TOTAL CUMULATIVE INFLOW. BAL00700
C   COMPUTE INCREASE IN STORAGE OF MASS DURING TIME STEP AND SINCE      BAL00710
C   START OF SIMULATION. FIND RATE OF STORAGE CHANGE FOR LAST TIME STEP.BAL00720
C   CALCULATE ERRORS FOR TOTAL SIMULATION AND FOR TIME STEP.            BAL00730
C                                                                       BAL00740
C*********************************************************************BAL00750
      QXT=SAVE(2,1,1)-SAVE(2,2,1)*PP(NN-1)-SAVE(2,3,1)*PP(NN)           BAL00760
      QXB=-SAVE(1,1,1)+SAVE(1,2,1)*PP(1)+SAVE(1,3,1)*PP(2)              BAL00770
      FLXNT=QXB-QXT                                                     BAL00780
      FLUX=FLXNT*DELT                                                   BAL00790
      XFLUX=XFLUX+FLUX                                                  BAL00800
      GAIN=SX-SOLD                                                      BAL00810
      CGAIN=SX-SXI                                                      BAL00820
      DSDT=GAIN/DELT                                                    BAL00830
      ER=(DSDT-FLXNT)*100./DSDT                                         BAL00840
      ERRC=(CGAIN-XFLUX)*100./CGAIN                                     BAL00850
      IF(I11.EQ.1.OR.(I11.EQ.0.AND.DABS(TIME-TEND).LT.1.D-10))          BAL00860
     1 WRITE(6,1010) QXT,QXB,FLXNT,DSDT,ER,XFLUX,CGAIN,ERRC             BAL00870
      IF(I24.EQ.3)  RETURN                                              BAL00880
C*********************************************************************BAL00890
C                                                                       BAL00900
C   REPEAT THE ABOVE COMPUTATIONS FOR THE ENERGY BALANCE.               BAL00910
C                                                                       BAL00920
C*********************************************************************BAL00930
      QTT=SAVE(2,1,2)-SAVE(2,2,2)*TT(NN-1)-SAVE(2,3,2)*TT(NN)           BAL00940
      QTB=-SAVE(1,1,2)+SAVE(1,2,2)*TT(1)+SAVE(1,3,2)*TT(2)              BAL00950
      FLXNT=QTB-QTT                                                     BAL00960
      FLUX=FLXNT*DELT                                                   BAL00970
      TFLUX=TFLUX+FLUX                                                  BAL00980
      GAIN=ST-STOLD                                                     BAL00990
      CGAIN=ST-STI                                                      BAL01000
```

```
      DSDT=GAIN/DELT                                                    BAL01010
      ER=(DSDT-FLXNT)*100./DSDT                                         BAL01020
      ERRC=(CGAIN-TFLUX)*100./CGAIN                                     BAL01030
      IF (I12.EQ.1.OR.(I12.EQ.0.AND.DABS(TIME-TEND).LT.1.D-10))         BAL01040
     1 WRITE(6,1020) QTT,QTB,FLXNT,DSDT,ER,TFLUX,CGAIN,ERRC             BAL01050
      RETURN                                                            BAL01060
C*******************************************************************BAL01070
C                                                                       BAL01080
C   FORMAT STATEMENTS                                                   BAL01090
C                                                                       BAL01100
C*******************************************************************BAL01110
1010  FORMAT(///10X,'MASS BALANCE INFORMATION'/10X,24(1H=)//15X,'DATA FOBAL01120
     1R CURRENT TIME STEP'//20X,'SURFACE FLUX RATE',10X,D12.5,' CM/S',  BAL01130
     2/20X,'BOTTOM FLUX RATE',11X,D12.5,' CM/S',/20X,'NET FLUX RATE',14XBAL01140
     3,D12.5,' CM/S'/20X,'RATE OF STORAGE CHANGE',5X,D12.5,' CM/S',     BAL01150
     4 //20X,'PERCENT ERROR',14X,F12.5,                                BAL01160
     5//15X,'CUMULATIVE DATA'//20X,'NET TOTAL FLUX',13X,D12.5,' CM',    BAL01170
     6 /20X,'TOTAL STORAGE CHANGE',7X,D12.5,' CM'//20X,'PERCENT ERROR', BAL01180
     714X,F12.5)                                                        BAL01190
1020  FORMAT(///10X,'ENERGY BALANCE INFORMATION'/10X,26(1H=)//15X,'DATA BAL01200
     1FOR CURRENT TIME STEP'//20X,'SURFACE FLUX RATE',10X,D12.5,' CAL/CMBAL01210
     22-S'/20X,'BOTTOM FLUX RATE',11X,D12.5,' CAL/CM2-S',/20X,'NET FLUX BAL01220
     3RATE',14X,D12.5,' CAL/CM2-S'/20X,'RATE OF STORAGE CHANGE',5X,D12.5BAL01230
     4,' CAL/CM2-S', //20X,'PERCENT ERROR',14X,F12.5,                   BAL01240
     5//15X,'CUMULATIVE DATA'//20X,'NET TOTAL FLUX',13X,D12.5,' CAL/CM2'BAL01250
     6,/20X,'TOTAL STORAGE CHANGE',7X,D12.5,' CAL/CM2'//20X,'PERCENT ERRBAL01260
     7OR',14X,F12.5)                                                    BAL01270
      END                                                               BAL01280
C*******************************************************************BAL01290
```

```
C*****************************************************************SOI00010
C                                                                 SOI00020
      SUBROUTINE SOILI2                                           SOI00030
C                                                                 SOI00040
C*****************************************************************SOI00050
C                                                                 SOI00060
C        THE FIRST PART OF THIS SUBROUTINE (SOILI2) IS CALLED ONCE AT   SOI00070
C   THE BEGINNING OF A SIMULATION IN ORDER TO READ THE SOIL PARAMETERS. SOI00080
C   COMPUTE OTHER DERIVED PARAMETERS, AND PERFORM PART OF THE          SOI00090
C   INTEGRATION OF THE RELATIVE HYDRAULIC CONDUCTIVITY FUNCTION.       SOI00100
C   THE OTHER ENTRIES ARE SOIL12, SOIL22, AND SOIL32, WHICH ARE        SOI00110
C   DESCRIBED BRIEFLY BELOW.                                           SOI00120
C                                                                 SOI00130
C*****************************************************************SOI00140
C*****************************************************************SOI00150
C                                                                 SOI00160
C   THE FOLLOWING VARIABLES, NOT DEFINED IN THE MAIN PROGRAM, ARE      SOI00170
C   USED IN THIS SUBPROGRAM:                                           SOI00180
C                                                                 SOI00190
C*****************************************************************SOI00200
C   NAME            DESCRIPTION                                    SOI00210
C*****************************************************************SOI00220
C   A1(NS)          MOISTURE CONTENT INTERCEPT OF THE CAPILLARY     SOI00230
C                   SEGMENT OF THE MAIN WETTING CURVE               SOI00240
C   A2(NS)          MOISTURE CONTENT INTERCEPT OF THE ADSORPTION    SOI00250
C                   SEGMENT OF THE MAIN WETTING CURVE               SOI00260
C   CKSAT(NS)       SATURATED (THETA-SUB-U), REFERENCE TEMPERATURE  SOI00270
C                   VALUE OF HYDRAULIC CONDUCTIVTY                  SOI00280
C   EM              CURVATURE PARAMETER OF MAIN WETTING CURVE       SOI00290
C   ENT(NS,8)      PARTIAL INTEGRALS OF RELATIVE HYDRAULIC CONDUCTIVTY SOI00300
C                   FUNCTIONS                                       SOI00310
C   NR              NUMBER OF BREAKPOINTS IN THE PIECE-WISE LINEAR  SOI00320
C                   APPROXIMATION OF THE MAIN WETTING CURVE (=7)    SOI00330
C   NSS             NUMBER OF SEGMENTS IN THE APPROXIMATION OF THE  SOI00340
C                   MAIN WETTING CURVE (=NR+1)                      SOI00350
C   NY              NUMBER OF VALUES OF XR,PR,PFR (=NR+2)           SOI00360
C   PB              NEGATIVE MATRIC HEAD AT REFERENCE TEMPERATURE   SOI00370
C                   TRANSITION BETWEEN SATURATED AND UNSATURATED REGIMES SOI00380
C   PFINT1(NS)      PF-INTERCEPT OF THE CAPILLARY SEGMENT OF THE MAIN SOI00390
C                   WETTING CURVE                                   SOI00400
C   PFINT2          PF-INTERCEPT OF THE ADSORPTION SEGMENT OF THE MAIN SOI00410
C                   WETTING CURVE (=7)                              SOI00420
C   PFK             VALUE OF PF AT WHICH LIQUID PHASE BECOMES       SOI00430
C                   DISCONTINUOUS                                   SOI00440
C   PFR(NS,9)      VALUES OF PF AT LIMITING POINTS IN THE APPROXIMATION SOI00450
C                   TO THE MAIN WETTING CURVE                       SOI00460
C   PFWLT           VALUE OF PF AT THE WILTING POINT (=4.2)         SOI00470
C   PR(NS,9)       VALUES OF MATRIC HEAD CORRESPONDINT TO PFR       SOI00480
C   R(7)           PARAMETER USED TO DETERMINE BREAKPOINTS IN PIECEWISE SOI00490
C                   LINEAR APPROXIMATION                            SOI00500
```

```
C   S1(NS)          SLOPE PARAMETER OF THE CAPILLARY SEGMENT                SOI00510
C   S2(NS)          SLOPE PARAMETER OF THE ADSORPTION SEGMENT               SOI00520
C   SS(NS,8)       SLOPE PARAMETERS OF THE PIECEWISE LINEAR                 SOI00530
C                   APPROXIMATION                                           SOI00540
C   XK(NS)          MOISTURE CONTENT CORRESPONDING TO PFK                   SOI00550
C   XR(NS,9)       MOISTURE CONTENTS CORRESPONDING TO PFR AND PR            SOI00560
C   XWILT(NS)       MOISTURE CONTENT AT WILTING POINT                       SOI00570
C***********************************************************************SOI00580
C***********************************************************************SOI00590
C                                                                          SOI00600
C   COMMON AND DIMENSION STATEMENTS                                        SOI00610
C                                                                          SOI00620
C***********************************************************************SOI00630
      IMPLICIT REAL*8(A-H,O-Z)                                             SOI00640
      COMMON C(46,2),D(46,2)                                               SOI00650
      COMMON B1(45,2),B7(45,2),CK(45,2),C1(45,2),C2(45,2),C3(45,2),        SOI00660
     *C4(45,2),C5(45,2),DXDP(45,2),EF(45,2),X(45,2),XX(45,2),XWRE(45,2)    SOI00670
      COMMON B3(46),B4(46),COR(46),                                        SOI00680
     *DELTA(46),P(46),                                                     SOI00690
     *PFU(46),PP(46),PPOLD(46),RHOV(46),T(46),                             SOI00700
     *TT(46),TTOLD(46),Z(46)                                               SOI00710
      COMMON XOLD(45)                                                      SOI00720
      COMMON POR(44)                                                       SOI00730
      COMMON BCP,BCT,CPLGE,CPSI,DELT,DELTO,DLOGE,DLN10,DLT,DURTN,PERR,     SOI00740
     *PN,RHOL,ST,STI,SX,SXI,TEND,TERR,TFLUX,THETA,TIME,TSAVE1,TSAVE2,      SOI00750
     *TZERO,XFLUX,ZETA                                                     SOI00760
      COMMON IH(46),IS(46)                                                 SOI00770
      COMMON I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15,I16,I17,   SOI00780
     *I18,I19,I20,I21,I22,I23,I24,I25,J1,J2,J3,J4,J5,J6,J7,J8,J9,ITDES,    SOI00790
     *KIT,KK,KT,NBCP,NBCT,NL,NN,NPER,NS,NSTEPS                             SOI00800
      COMMON /SOIL/ A1(44),A2(44),EM(44),S1(44),S2(44)                     SOI00810
      COMMON /ADHOC/ PB(44)                                                SOI00820
      DIMENSION ENT(44,8),PFR(44,9),PR(44,9),R(7),SS(44,8),XR(44,9)        SOI00830
      DIMENSION CKSAT(44),XK(44),XWILT(44),PFINT1(44)                      SOI00840
      DATA R /1.D-02,1.D-01,3.D-01,5.D-01,7.D-01,9.D-01,9.9D-01/           SOI00850
C***********************************************************************SOI00860
C                                                                          SOI00870
C   PERFORM INITIAL SOIL CALCULATIONS.                                     SOI00880
C                                                                          SOI00890
C***********************************************************************SOI00900
      READ(5,1010)(POR(I),PB(I),XWILT(I),EM(I),CKSAT(I),PFINT1(I),         SOI00910
     1I=1,NS)                                                              SOI00920
      IF(I5.EQ.1) WRITE(6,1020) (I,POR(I),PB(I),XWILT(I),EM(I),CKSAT(I),   SOI00930
     *PFINT1(I),I=1,NS)                                                    SOI00940
      PFINT2=7.                                                            SOI00950
      PFWLT=4.2                                                            SOI00960
      PFK=5.85                                                             SOI00970
      NR=7                                                                 SOI00980
      NSS=NR+1                                                             SOI00990
      NY=NR+2                                                              SOI01000
```

```
C*****************************************************************SOI01010
C                                                                 SOI01020
C   FIT MAIN WETTING CURVE TO THE SPECIFIED WILTING POINT         SOI01030
C   AND SATURATION POINT.                                         SOI01040
C                                                                 SOI01050
C*****************************************************************SOI01060
      DO 30 I=1,NS                                                SOI01070
      S1(I)=POR(I)/(PFINT1(I)-DLOG10(PB(I)))                      SOI01080
      S2(I)=XWILT(I)/(PFINT2-PFWLT)                               SOI01090
      A1(I)=S1(I)*PFINT1(I)                                       SOI01100
      A2(I)=S2(I)*PFINT2                                          SOI01110
      PFB=DLOG10(PB(I))                                           SOI01120
 10   XW=XWET(PFWLT,I)                                            SOI01130
      XP=XWET(PFB,I)                                              SOI01140
      IF(DABS(XW-XWILT(I)).LT.1.D-10.AND.DABS(XP-POR(I)).LT.1.D-10)SOI01150
     1 GO TO 20                                                   SOI01160
      A1(I)=A1(I)-(XP-POR(I))                                     SOI01170
      A2(I)=A2(I)-(XW-XWILT(I))                                   SOI01180
      GO TO 10                                                    SOI01190
 20   XK(I)=XWET(PFK,I)                                           SOI01200
 30   CONTINUE                                                    SOI01210
      IF(I6.EQ.1) WRITE(6,1030) (I,A1(I),A2(I),S1(I),S2(I),XK(I),I=1,NS)SOI01220
C*****************************************************************SOI01230
C                                                                 SOI01240
C   FIT THE MAIN WETTING CURVE TO A PIECEWISE LINEAR RELATION     SOI01250
C   BETWEEEN MOISTURE CONTENT AND PF. THIS APPROXIMATION IS       SOI01260
C   USED ONLY IN THE NUMERICAL INTEGRATION OF THE RELATIVE HYDRAULICSOI01270
C   CONDUCTIVITY FUNCTION.                                        SOI01280
C                                                                 SOI01290
C*****************************************************************SOI01300
      DO 90 I=1,NS                                                SOI01310
      XR(I,1)=XK(I)                                               SOI01320
      PFR(I,1)=PFK                                                SOI01330
      PR(I,1)=-10.**PFK                                           SOI01340
      DO 70 J=1,NR                                                SOI01350
      X1=XR(I,J)                                                  SOI01360
 40   PF1=(A1(I)-X1)/S1(I)                                        SOI01370
      X1OLD=X1                                                    SOI01380
      IF(XWET(PF1,I).LE.0.) GO TO 50                              SOI01390
      X1=DLOG(EM(I)*R(J)*XWET(PF1,I))/EM(I)                       SOI01400
      IF(DABS(X1-X1OLD).GT.1.D-05) GO TO 40                       SOI01410
      GO TO 60                                                    SOI01420
 50   PF1=PFK-1.D-10*J                                            SOI01430
 60   XR(I,J+1)=XWET(PF1,I)                                       SOI01440
      PFR(I,J+1)=PF1                                              SOI01450
      PR(I,J+1)=-10.**PF1                                         SOI01460
 70   SS(I,J)=(PFR(I,J)-PFR(I,J+1))/(XR(I,J+1)-XR(I,J))           SOI01470
      XR(I,NY)=POR(I)                                             SOI01480
      PFR(I,NY)=DLOG10(PB(I))                                     SOI01490
      PR(I,NY)=-PB(I)                                             SOI01500
```

```
      SS(I,NSS)=(PFR(I,NSS)-PFR(I,NY))/(XR(I,NY)-XR(I,NSS))          SOI01510
C*****************************************************************SOI01520
C                                                                    SOI01530
C   DO INITIAL INTEGRATION OF THE RELATIVE HYDRAULIC CONDUCTIVITY    SOI01540
C   FUNCTION.                                                        SOI01550
C                                                                    SOI01560
C*****************************************************************SOI01570
      DO 80 J=1,NSS                                                  SOI01580
      ENT(I,J)=(1./PR(I,J+1)-1./PR(I,J))/SS(I,J)                     SOI01590
      IF(J.EQ.1) GO TO 80                                            SOI01600
      ENT(I,J)=ENT(I,J)+ENT(I,J-1)                                   SOI01610
   80 CONTINUE                                                       SOI01620
   90 CONTINUE                                                       SOI01630
      IF(I6.EQ.1) WRITE(6,1040) (I,(XR(I,J),J=1,NY),(PFR(I,J),J=1,NY),  SOI01640
     1(PR(I,J),J=1,NY),(SS(I,J),J=1,NSS),(ENT(I,J),J=1,NSS),I=1,NS)  SOI01650
      RETURN                                                         SOI01660
C*****************************************************************SOI01670
C*****************************************************************SOI01680
C                                                                    SOI01690
      ENTRY SOIL12                                                   SOI01700
C                                                                    SOI01710
C*****************************************************************SOI01720
C                                                                    SOI01730
C   THIS ENTRY IS CALLED FROM NWSTP2 TO UPDATE THE WETTING HISTORY   SOI01740
C   INFORMATION, IF NECESSARY. IF THE CHANGE IN AVERAGE MOISTURE     SOI01750
C   CONTENT OF THE ELEMENT DURING THE LAST TIME STEP WAS             SOI01760
C   IN THE OPPOSITE DIRECTION OF THAT DURING THE PREVIOUS TIME STEP. SOI01770
C   THE HISTORY IS UPDATED. SEE SECTION 3.3.                         SOI01780
C                                                                    SOI01790
C*****************************************************************SOI01800
      DO 140 L=1,NL                                                  SOI01810
      J=IS(L)                                                        SOI01820
      EX=0.5*(X(L,1)+X(L,2))                                         SOI01830
      IF(IH(L).EQ.1.AND.XOLD(L).LE.EX) GO TO 140                     SOI01840
      IF(IH(L).EQ.2) GO TO 110                                       SOI01850
      IH(L)=2                                                        SOI01860
      DO 100 N=1,2                                                   SOI01870
      XWR=XWET(PFU(L+N-1),J)                                         SOI01880
      XWR=DMIN1(XWR,POR(J))                                          SOI01890
  100 XWRE(L,N)=XWR+POR(J)*(X(L,N)/XWR-1.)                           SOI01900
      GO TO 140                                                      SOI01910
  110 IF(XOLD(L).GE.EX) GO TO 140                                    SOI01920
      IH(L)=1                                                        SOI01930
      DO 130 N=1,2                                                   SOI01940
      XWR=XWET(PFU(L+N-1),J)                                         SOI01950
      XWR=DMIN1(XWR,POR(J))                                          SOI01960
      IF(XWR.EQ.POR(J)) GO TO 120                                    SOI01970
      XWRE(L,N)=POR(J)*(X(L,N)-XWR)/(POR(J)-XWR)                     SOI01980
      GO TO 130                                                      SOI01990
  120 XWRE(L,N)=POR(J)                                               SOI02000
```

```
 130  CONTINUE                                                          SOI02010
 140  XOLD(L)=EX                                                        SOI02020
      RETURN                                                            SOI02030
C*******************************************************************SOI02040
C                                                                       SOI02050
C        CALCULATE MOISTURE CONTENT AND SPECIFIC MOISTURE CAPACITY      SOI02060
C   (IF KODE EQUALS 2). IF J5 IS EQUAL TO ONE. THE MOISTURE             SOI02070
C   CHARACTERISTIC MAY BE SPECIFIED BY FORTRAN STATEMENTS. AND THE      SOI02080
C   USUAL HYSTERESIS AND MAIN WETTING CURVE ARE IGNORED.                SOI02090
C                                                                       SOI02100
C*******************************************************************SOI02110
      ENTRY SOIL22(KODE)                                                SOI02120
      DO 160 I=1,NN                                                     SOI02130
      IF(PP(I).GT.-1.D0) GO TO 150                                      SOI02140
      PFU(I)=DLOG10(-PP(I))                                             SOI02150
      IF(I24.NE.3) PFU(I)=PFU(I)-CPLGE*(TT(I)-TZERO)                    SOI02160
      GO TO 160                                                         SOI02170
 150  PFU(I)=-1.D0                                                      SOI02180
 160  CONTINUE                                                          SOI02190
      DO 230 L=1,NL                                                     SOI02200
      DO 230 N=1,2                                                      SOI02210
      I=L+N-1                                                           SOI02220
      J=IS(L)                                                           SOI02230
      IF(J5.EQ.1) GO TO 210                                             SOI02240
      IF(PFU(I).LT.PFR(J,NY)) GO TO 170                                 SOI02250
      ARG1=DEXP(EM(J)*(A1(J)-S1(J)*PFU(I)))                             SOI02260
      ARG2=DEXP(EM(J)*(A2(J)-S2(J)*PFU(I)))                             SOI02270
      XWE=DLOG(ARG1+ARG2)/EM(J)                                         SOI02280
      IF(KODE.EQ.1) GO TO 180                                           SOI02290
      PUE=-10.**PFU(I)                                                  SOI02300
      DXWEDP=(-ARG1*EM(J)*S1(J)/DLN10/PUE-ARG2*EM(J)*S2(J)/DLN10/       SOI02310
     *PUE)/EM(J)/(ARG1+ARG2)                                            SOI02320
      GO TO 180                                                         SOI02330
 170  XWE=POR(J)                                                        SOI02340
      DXWEDP=0.0                                                        SOI02350
 180  IF(IH(L).EQ.2) GO TO 190                                          SOI02360
      XX(L,N)=XWE+XWRE(L,N)*(1.-XWE/POR(J))                             SOI02370
      IF(KODE.EQ.1) GO TO 200                                           SOI02380
      DXDP(L,N)=DXWEDP*(1.-XWRE(L,N)/POR(J))                            SOI02390
      GO TO 200                                                         SOI02400
 190  XX(L,N)=XWE*(1.+(XWRE(L,N)-XWE)/POR(J))                           SOI02410
      IF(KODE.EQ.1) GO TO 200                                           SOI02420
      DXDP(L,N)=DXWEDP*(XX(L,N)/XWE-XWE/POR(J))                         SOI02430
 200  IF(XX(L,N).GT.POR(J)) XX(L,N)=POR(J)                              SOI02440
      GO TO 230                                                         SOI02450
 210  IF(PFU(I).LT.DLOG10(PB(J))) GO TO 220                             SOI02460
C*******************************************************************SOI02470
C                                                                       SOI02480
C   USER-SUPPLIED ALTERNATIVE EQUATIONS FOR RETENTION CURVE AND ITS     SOI02490
C   SLOPE FOLLOW:                                                       SOI02500
```

```
      XX(L,N)=274.169/(739.+(DLN10*PFU(I))**4.)+0.124                    SOI02510
      IF(KODE.EQ.1) GO TO 230                                           SOI02520
      DXDP(L,N)=274.169/(739.+(DLN10*PFU(I))**4.)**2.*4.*(PFU(I)*DLN10) SOI02530
     1**3./10.**PFU(I)                                                  SOI02540
C*******************************************************************SOI02550
      GO TO 230                                                         SOI02560
 220  XX(L,N)=POR(J)                                                    SOI02570
      DXDP(L,N)=0.0                                                     SOI02580
 230  CONTINUE                                                          SOI02590
      RETURN                                                            SOI02600
C*******************************************************************SOI02610
C                                                                       SOI02620
      ENTRY SOIL32                                                      SOI02630
C                                                                       SOI02640
C*******************************************************************SOI02650
C                                                                       SOI02660
C   CALCULATE RELATIVE HYDRAULIC CONDUCTIVITY AND THE FACTOR, EF.       SOI02670
C                                                                       SOI02680
C*******************************************************************SOI02690
      DO 320 L=1,NL                                                     SOI02700
      DO 320 N=1,2                                                      SOI02710
      J=IS(L)                                                           SOI02720
      IF(XX(L,N).GE.POR(J)) GO TO 290                                   SOI02730
      IF(J5.EQ.1) GO TO 280                                             SOI02740
      DO 240 K=1,NSS                                                    SOI02750
      NK=NSS+1-K                                                        SOI02760
      IF(XX(L,N).GT.XR(J,NK)) GO TO 250                                 SOI02770
      IF(K.EQ.NSS) GO TO 300                                            SOI02780
 240  CONTINUE                                                          SOI02790
 250  PFM=PFR(J,NK)-SS(J,NK)*(XX(L,N)-XR(J,NK))                         SOI02800
      SSS=SS(J,NK)                                                      SOI02810
260   XXX=XWET(PFM,J)                                                   SOI02820
      IF(DABS(XXX-XX(L,N)).LT.1.D-08) GO TO 270                         SOI02830
      PFM=PFM+(XXX-XX(L,N))*SSS                                         SOI02840
      GO TO 260                                                         SOI02850
 270  PWET=-10.**PFM                                                    SOI02860
      CK1=0.                                                            SOI02870
      IF(NK.GT.1) CK1=ENT(J,NK-1)                                       SOI02880
      CKPART=(CK1+(1./PWET-1./PR(J,NK))                                 SOI02890
     */SS(J,NK))/ENT(J,NSS)                                             SOI02900
      CKPART=CKPART*CKPART                                              SOI02910
      CK(L,N)=CKSAT(J)*DSQRT((XX(L,N)-XK(J))/(POR(J)-XK(J)))*CKPART     SOI02920
      GO TO 310                                                         SOI02930
C*******************************************************************SOI02940
C                                                                       SOI02950
C USER-SUPPLIED FORMULA FOR HYDRAULIC CONDUCTIVTY FOLLOWS:              SOI02960
C                                                                       SOI02970
 280  CK(L,N)=CKSAT(J)*124.6/(124.6+(-PP(L+N-1))**1.77)                 SOI02980
C                                                                       SOI02990
C*******************************************************************SOI03000
```

```
      GO TO 310                                                          SOI03010
 290  CK(L,N)=CKSAT(J)                                                   SOI03020
      GO TO 310                                                          SOI03030
 300  CK(L,N)=0.                                                         SOI03040
      IF(I24.EQ.3) CK(L,N)=CKSAT(J)*1.D-20                               SOI03050
      EF(L,N)=1.11*POR(J)                                                SOI03060
      GO TO 320                                                          SOI03070
 310  EF(L,N)=(1.11*POR(J)-XX(L,N))*(1.+XX(L,N)/(1.11*POR(J)-XK(J)))     SOI03080
 320  CONTINUE                                                           SOI03090
      RETURN                                                             SOI03100
C*************************************************************************SOI03110
C                                                                        SOI03120
C   FORMAT STATEMENTS                                                    SOI03130
C                                                                        SOI03140
C*************************************************************************SOI03150
1010  FORMAT(6G10.0,20X)                                                 SOI03160
1020  FORMAT(///5X,'INPUT SOIL PARAMETERS'/5X,21(1H=)//10X,'SOIL TYPE',  SOI03170
     15X,'POR',5X,'PB',5X,'XWILT',4X,'EM',6X,'CKSAT',5X,'PFINT1',5X,     SOI03180
     2/10X,65(1H-)/10(/14X,I1,8X,F4.2,3X,F5.0,4X,F4.2,4X,F4.0,           SOI03190
     32X,D9.2,4X,F6.4))                                                 SOI03200
1030  FORMAT(///5X,'COMPUTED SOIL PARAMETERS'/5X,24(1H=)//10X,           SOI03210
     1'SOIL TYPE',9X,'A1',9X,'A2',9X,'S1',9X,'S2',9X,'XK',              SOI03220
     2/10X,67(1H-)/10(14X,I1,7X,5(F10.5,1X)))                           SOI03230
1040  FORMAT(//5X,'SOIL TYPE',I3/5X,12(1H-)/6X,'XR',9E12.2/5X,'PFR',9E12SOI03240
     1.2/6X,'PR',9E12.2/6X,'SS',8E12.2/5X,'ENT',8E12.2)                 SOI03250
      END                                                                SOI03260
C*************************************************************************SOI03270
```

```
C*******************************************************************RHO00010
C                                                                    RHO00020
      FUNCTION RHOZZ2(TEE)                                           RHO00030
C                                                                    RHO00040
C*******************************************************************RHO00050
C                                                                    RHO00060
C         THE SUBPROGRAM CALCULATES THE VARIOUS PHYSICAL             RHO00070
C   PROPERTIES OF WATER AS A FUNCTION OF TEMPERATURE. THIS INCLUDES   RHO00080
C   SATURATED VAPOR DENSITY, SLOPE OF SATURATION VAPOR DENSITY CURVE. RHO00090
C   AND THE TEMPERATURE CORRECTION FOR HYDRAULIC CONDUCTIVITY.       RHO00100
C                                                                    RHO00110
C*******************************************************************RHO00120
      IMPLICIT REAL*8(A-H,O-Z)                                       RHO00130
      COMMON /CKTEE/ CKTN                                            RHO00140
      DIMENSION RZ(81),SIGMA(9)                                      RHO00150
C*******************************************************************RHO00160
C                                                                    RHO00170
C   RZ(I) CONTAINS THE VALUE OF THE SATURATION VAPOR DENSITY AT      RHO00180
C   TEMPERATURE I. SIGMA CONTAINS THE SURFACE TENSION OF WATER AT    RHO00190
C   TEN DEGREE INCREMENTS (NOT CURRENTLY IN USE).                    RHO00200
C                                                                    RHO00210
C*******************************************************************RHO00220
       DATA RZ/ 0.4831D-05,0.5178D-05,0.5545D-05,0.5933D-05,0.6344D-05. RHO00230
     1          0.6779D-05,0.7240D-05,0.7728D-05,0.8246D-05.0.8793D-05. RHO00240
     2          0.9373D-05,0.9984D-05,0.1063D-04,0.1131D-04.0.1203D-04. RHO00250
     3          0.1279D-04,0.1358D-04,0.1443D-04,0.1532D-04.0.1626D-04. RHO00260
     4          0.1724D-04,0.1827D-04,0.1936D-04,0.2050D-04.0.2170D-04. RHO00270
     5          0.2297D-04,0.2429D-04,0.2567D-04,0.2713D-04.0.2865D-04. RHO00280
     6          0.3025D-04,0.3193D-04,0.3368D-04,0.3552D-04.0.3744D-04. RHO00290
     7          0.3945D-04,0.4154D-04,0.4374D-04,0.4602D-04.0.4841D-04. RHO00300
     8          0.5091D-04,0.5352D-04,0.5624D-04,0.5907D-04.0.6203D-04. RHO00310
     9          0.6511D-04,0.6831D-04,0.7165D-04,0.7512D-04.0.7874D-04. RHO00320
     A 0.8250D-04,0.8641D-04,0.9049D-04,0.9472D-04,0.9910D-04.          RHO00330
     1          0.1036D-03,0.1084D-03,0.1133D-03,0.1184D-03.0.1237D-03. RHO00340
     2          0.1292D-03,0.1349D-03,0.1408D-03,0.1469D-03.0.1533D-03. RHO00350
     3          0.1598D-03,0.1666D-03,0.1737D-03,0.1809D-03.0.1884D-03. RHO00360
     4          0.1963D-03,0.2042D-03,0.2126D-03,0.2212D-03.0.2301D-03. RHO00370
     5          0.2393D-03,0.2488D-03,0.2585D-03,0.2686D-03.0.2791D-03. RHO00380
     6          0.2898D-03/                                         RHO00390
      DATA RHOLG/979./                                               RHO00400
      DATA SIGMA/75.6D+00,74.22D+00,72.75D+00.71.18D+00,69.56D+00,    RHO00410
     167.91D+00,66.18D+00,64.4D+00,62.6D+00/                         RHO00420
C*******************************************************************RHO00430
C                                                                    RHO00440
C   FIND SATURATION VAPOR DENSITY (G/CM3).                           RHO00450
C                                                                    RHO00460
C*******************************************************************RHO00470
      L=TEE                                                          RHO00480
      EXTRA=TEE-L                                                    RHO00490
      RHOZZ2=RZ(L+1)+EXTRA*(RZ(L+2)-RZ(L+1))                         RHO00500
```

```
      RETURN                                                            RHO00510
C*****************************************************************************RHO00520
C                                                                       RHO00530
C   COMPUTE SLOPE OF SATURATION VAPOR DENSITY CURVE (G/CM3-DEGK).       RHO00540
C                                                                       RHO00550
C*****************************************************************************RHO00560
      ENTRY DRZDT2(TEE)                                                 RHO00570
      L=TEE                                                             RHO00580
      DRZDT2=RZ(L+2)-RZ(L+1)                                            RHO00590
      RETURN                                                            RHO00600
C*****************************************************************************RHO00610
C                                                                       RHO00620
C                                                                       RHO00630
C   COMPUTE TEMPERATURE CORRECTION FOR HYDRAULIC CONDUCTIVITY.          RHO00640
C                                                                       RHO00650
C*****************************************************************************RHO00660
      ENTRY CKTT2(TEE)                                                  RHO00670
      IF (TEE.GT.20.) GO TO 10                                          RHO00680
      VISCC=10.**((1301./(998.333+8.1855*(TEE-20.)+0.00585*(TEE-20.)*(TERHO00690
     *E-20.)))-3.30233)                                                 RHO00700
      GO TO 20                                                          RHO00710
   10 VISCC=0.01002*10.**((1.3272*(20.-TEE)-0.001053*(TEE-20.)**2.)/(TEERHO00720
     **+105.))                                                          RHO00730
   20 CKTT2=1./VISCC/CKTN                                               RHO00740
      RETURN                                                            RHO00750
      END                                                               RHO00760
C*****************************************************************************RHO00770
```

```
C*************************************************************************XWE00010
C                                                                         XWE00020
      FUNCTION XWET(P,J)                                                  XWE00030
C                                                                         XWE00040
C*************************************************************************XWE00050
C                                                                         XWE00060
C   THIS FUNCTION SUBPROGRAM COMPUTES THE VALUE OF THE MAIN               XWE00070
C   WETTING CURVE USING EQATION 3.38.                                     XWE00080
C                                                                         XWE00090
C*************************************************************************XWE00100
      IMPLICIT REAL*8(A-H,O-Z)                                            XWE00110
      COMMON /SOIL/ A1(44),A2(44),EM(44),S1(44),S2(44)                    XWE00120
      XWET=DLOG(DEXP(EM(J)*(A1(J)-S1(J)*P))+DEXP(EM(J)*(A2(J)-            XWE00130
     *S2(J)*P)))/EM(J)                                                    XWE00140
      RETURN                                                              XWE00150
      END                                                                 XWE00160
C*************************************************************************XWE00170
```