$$K(h) = k_r K_s \qquad (3\text{-}2)$$

where k_r is the relative permeability, ranging in value from 0.0 to 1.0, and K_s is the saturated hydraulic conductivity (L/T). The saturated hydraulic conductivity is a flow property of the porous medium and fluid which is determined by tests performed under saturated conditions. It represents a maximum possible value of effective hydraulic conductivity. The relative permeability term describes the influence of water content on the magnitude of the effective hydraulic conductivity. Values of relative permeability range from a minimum value reflecting the reduction of effective conductivity at residual water content to a maximum of 1.0 reflecting saturated conditions.

The change in relative permeability is caused by changes in moisture content, which result in the preferential movement of water through certain pathways, due to the influence of capillary forces. As the soil becomes less saturated, water drains more readily from large radius pore structures and water flow becomes restricted to pore sequences of smaller radii (Figure 3.2) as well as that held in layers close to the soil particles. The result of water becoming increasingly restricted to the smaller radius pathways is a reduction in the spatially-averaged effective hydraulic conductivity.

The decrease in effective hydraulic conductivity, as reflected in the relative permeability term, is described by pairs of empirical soil-moisture curves. These curves detail the relationships between water content and pressure head and between hydraulic conductivity and water content. Soil-moisture curves are often described as coefficients and exponents of standard analytical functions (Brooks and Corey, 1966; Mualem, 1976; van Genuchten, 1980). The 3DFEMWATER code allows the user to define the curves using the van Genuchten functions (1980) or as sets of paired values of relative permeability versus moisture content and moisture content versus pressure head given in lookup table format. The van Genuchten relationships found in 3DFEMWATER are as follows:

$$k_r = S_e^{1/2}\left[1 - \left(1 - S_e^{1/(}\right)^{(}\right]^2 \qquad (3\text{-}3a)$$

and

$$S_e = \begin{cases} [1 + (\alpha h - \alpha h_a)^{n}]^{-(} & \text{for } h < h_a \\ 1 & \text{for } h \geq h_a \end{cases} \qquad (3\text{-}3b)$$

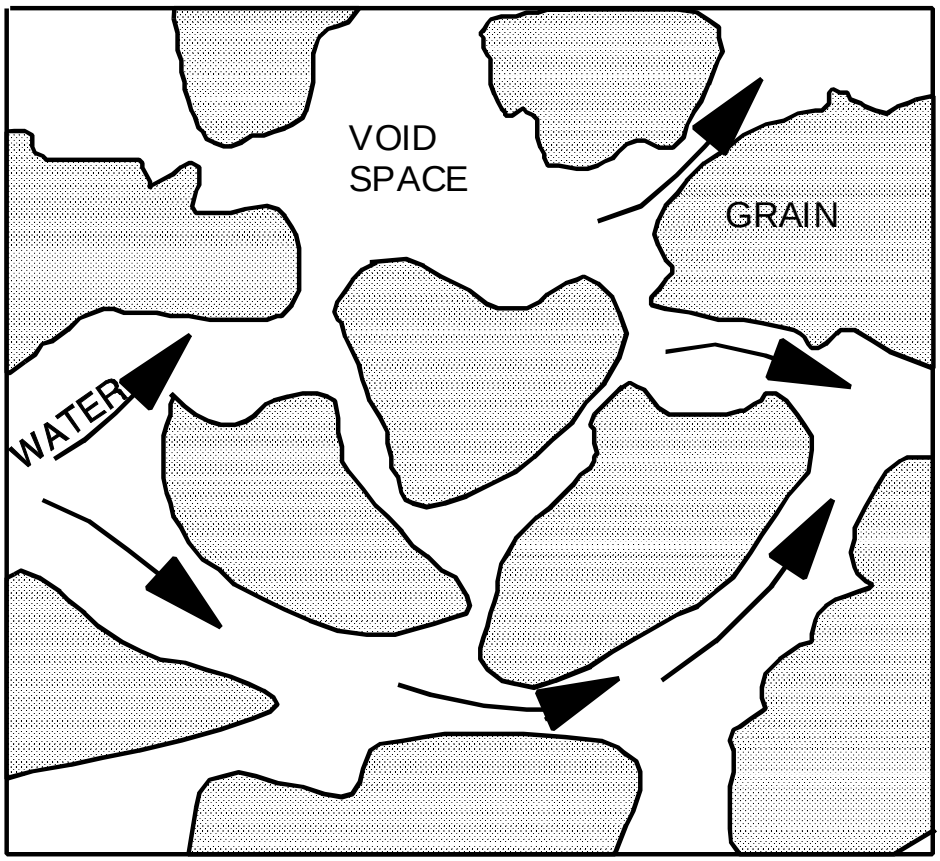


Figure 3.2 Variable pore spacing in soil under saturated flow conditions.

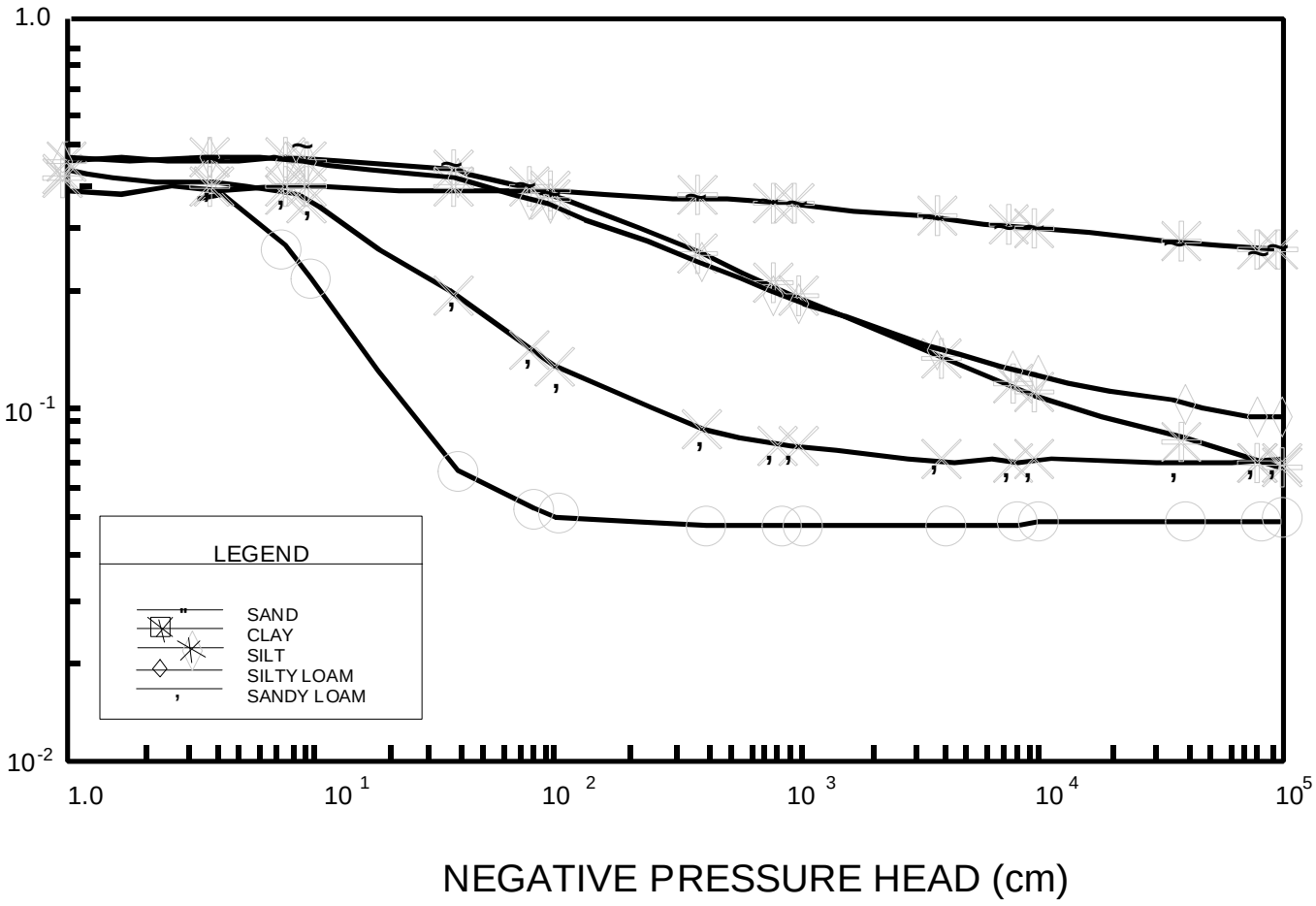
where

$$2_e \; ' \; \frac{2_w \; \& \; 2_{wr}}{N \; \& \; 2_{wr}} \qquad (3\text{-}3c)$$

$$(\; ' \; 1 \; \& \; 1/\$ \qquad (3\text{-}3d)$$

and

2_w = moisture content (dimensionless)



N = porosity (dimensionless)
2_{wr} = residual moisture content (dimensionless)
$, (= soil-specific exponents (dimensionless)
" = soil-specific coefficient (1/L)
h_a = air entry pressure head (L)
2_e = effective moisture content (dimensionless)

Note that the soil-moisture content is defined as the porosity times the degree of saturation. Typical soil-moisture curves generated from Equations 3-3a and 3-3b are presented in Figures 3.3a and 3.3b.

Figure 3.3. Logarithmic plot of constitutive relations for sand, clay, silty loam, and sandy loam: (a) moisture content vs. pressure head and (b) relative permeability vs. moisture content (based on data presented in Carsel and Parrish, 1988).

The water capacity term or storage term used in 3DFEMWATER can be written in the form:

$$F(h) = d\theta_w/dh \qquad (3\text{-}4)$$

It should be noted that due to the relatively small influence of compressibility on water capacity in the unsaturated zone (with respect to the drainage potential), soil and water compressibility have been ignored in the storage term. When analytical functions are used to describe the nonlinearity of the relative conductivity, the derivative with respect to pressure head of the water content versus pressure head function must also be analytically defined.

The equation governing saturated flow represents a limiting case of Richard's equation where the relative permeability is a constant of 1.0 and the water capacity is a constant equal to the specific yield for an unconfined aquifer or specific storage for a confined aquifer.

3.1.2 Boundary Conditions and Transient Source/Sink Terms

Unique solutions to variably-saturated flow problems are generated by solving Richard's equation in conjunction with 1) a set of boundary conditions defined at the physical edges of the modeled system and 2) where appropriate, source/sink terms applied within the system (Figure 3.4). Boundary conditions available in the 3DFEMWATER model include fixed-head (Dirichlet) boundaries, specified-flux (Cauchy) boundaries, specified-pressure-head gradient (Neumann) boundaries and variable (head-dependent flow) boundaries. *Fixed-head* or *Dirichlet* boundaries are boundaries defined by prescribing pressure heads at specified boundary nodes

so that:

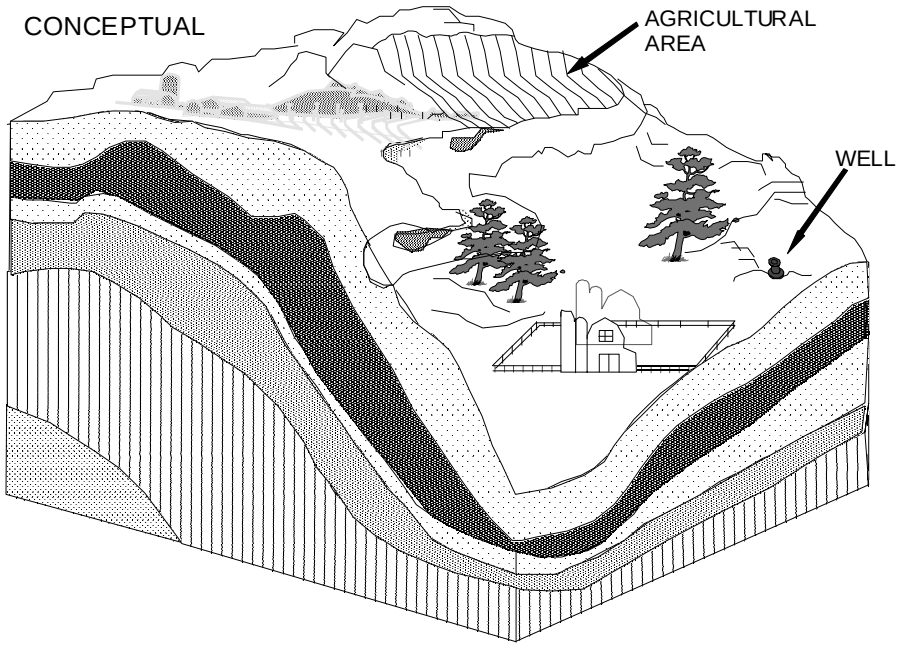
$$h = h_d(x_b, y_b, z_b, t) \text{ on } B_d \qquad (3\text{-}5)$$

where

h_d	= specified pressure head (L)
B_d	= portion of the system boundary subject to a Dirichlet boundary condition
x_b, y_b, z_b	= spatial coordinates on the boundary (L)

Dirichlet boundaries are typically used to define the perimeters of bodies of water, the water table location, and leaking surface impoundments or other waste disposal facilities containing specified levels of water. Specified pressure heads may be constant or allowed to vary with time reflecting physical processes such as water level fluctuations associated with seasonal changes in rainfall and evapotranspiration rates.

The *specified-flux (Cauchy)* boundary represents the portions of the system boundary where infiltration or evapotranspiration rates can be quantified. The specified-flux boundary condition can be written:


CONCEPTUAL
AGRICULTURAL AREA
WELL

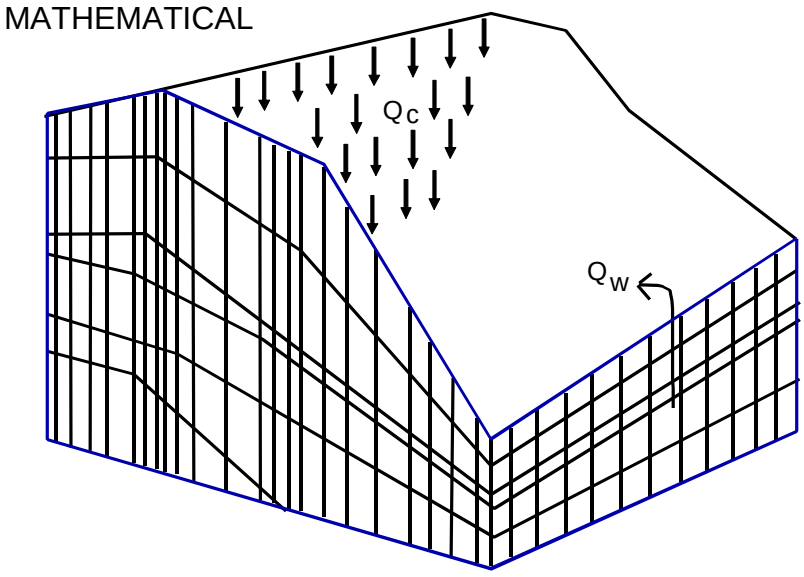

MATHEMATICAL
Qc
Qw

Figure 3.4. Conceptual model and mathematical approximation for variably-saturated flow system. Within the modeled system, transient source/sink terms may be applied as point sources/sinks or as distributed sources/sinks.

$$-n \cdot k_r K_s \cdot (\nabla h + \nabla z) = q_c(x_b, y_b, z_b, t) \text{ on } B_c \qquad (3\text{-}6)$$

where

- n = outward unit vector normal to the boundary
- ∇h = pressure head gradient
- ∇z = gravity gradient
- q_c = specified flux rate (L/T)
- B_c = portion of the system boundary subject to a specified-flux boundary condition
- k_r = relative permeability
- K_s = saturated hydraulic conductivity (L/T)

The specified-flux condition is analogous to a *Neumann* boundary condition for saturated flow problems differing only in the nonlinear nature of the effective hydraulic conductivity. The specified boundary is simulated by assigning water flux rates along specified element sides. Flux rate versus time profiles can be input to account for seasonal or other time-variant changes in rainfall and evapotranspiration rates. The default boundary condition for 3DFEMWATER is a zero specified-flux boundary condition, q_c=0.

Also available in 3DFEMWATER is a *specified-pressure-head gradient (Neumann)* boundary condition of the form:

$$-n \cdot k_r K_s \cdot \nabla h = q_n(x_b, y_b, z_b, t) \text{ on } B_n \qquad (3\text{-}7)$$

where q_n (L/T) is the portion of the boundary flux attributable to the pressure-head gradient and B_n is the portion of the system boundary subject to a specified-pressure-head gradient boundary condition. For unsaturated flow problems, the presence of this option provides the user an efficient way of evaluating systems with vertically extensive vadose zones. As long as the area of interest in a study is above the capillary fringe, the specified-pressure-head gradient boundary condition allows the user to truncate the system above the water table without knowing fluxes or pressure heads a priori (Figure 3.5). By choosing the specified-pressure-head gradient boundary condition option for element faces defining the bottom boundary of the system, and setting the flux q_n equal to zero, the bottom boundary becomes a gravity drainage boundary. This is equivalent to the code allowing the user to specify a flux along a horizontal bottom boundary of $q_c=k_rK_s$. This assumption of zero vertical change in pressure head near the bottom boundary is a reasonable assumption for slowly varying flow conditions and represents the outflow boundary condition that is usually assumed for field drainage experiments. This boundary condition is not appropriate for use in modeling the saturated zone.

The *variable composite* boundary condition represents a combined Dirichlet/ specified-flux boundary. It allows for time-variant infiltration/ evapotranspiration rates with limits set on the maximum and minimum pressure heads which the boundary nodes may attain. The variable boundary conditions during periods of precipitation are:

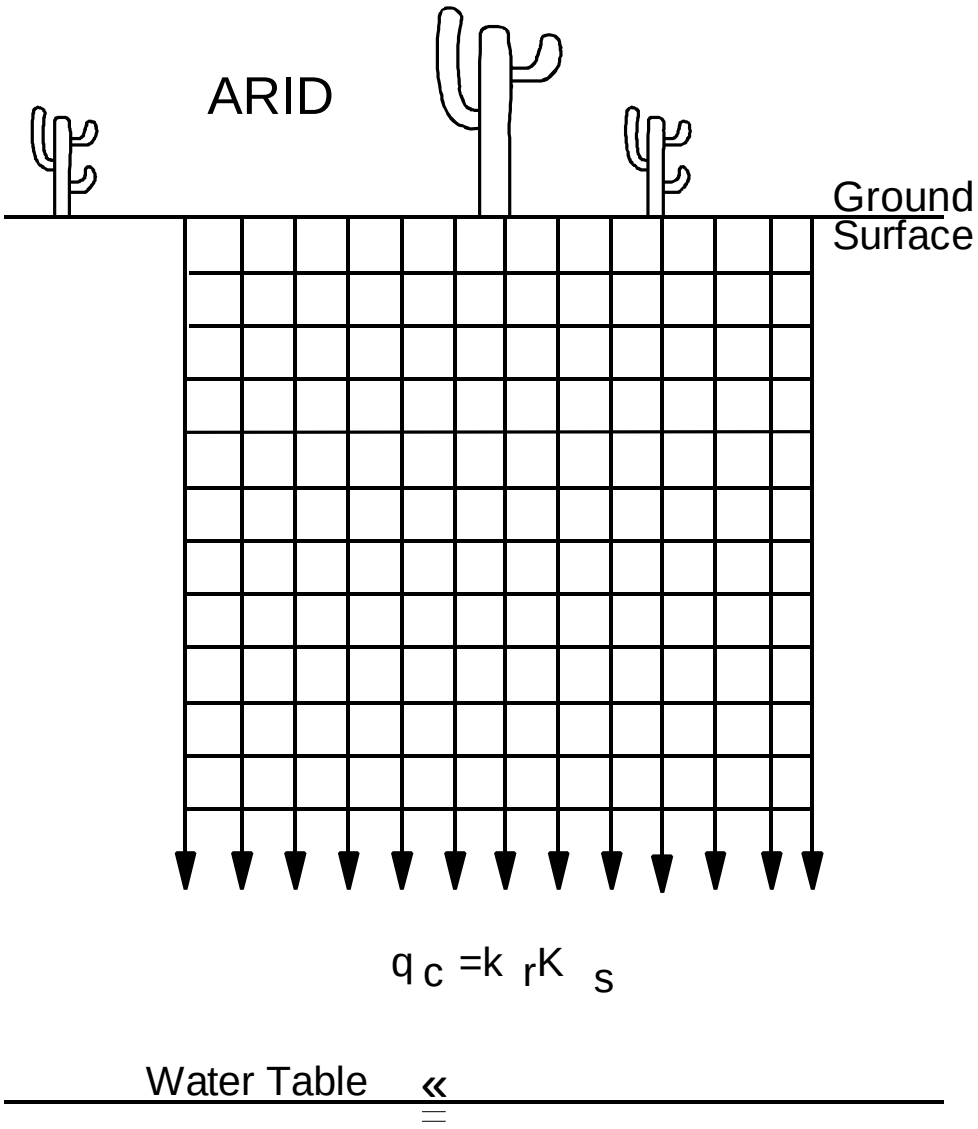


Figure 3.5. Use of a pressure-head gradient boundary condition to simulate a portion of the

unsaturated zone.

$$h = h_p(x_b, y_b, z_b, t) \text{ on } B_v \tag{3-8a}$$

or

$$-n\theta k_r K_s \theta (\partial h / \partial z) = q_p(x_b, y_b, z_b, t) \text{ on } B_v \tag{3-8b}$$

and during periods of nonprecipitation are:

$$h = h_p(x_b, y_b, z_b, t) \text{ on } B_v \tag{3-8c}$$

or

$$h = h_m(x_b, y_b, z_b, t) \text{ on } B_v \tag{3-8d}$$

or

$$-n\theta k_r K_s \theta (\partial h / \partial z) = q_e(x_b, y_b, z_b, t) \text{ on } B_v \tag{3-8e}$$

where

h_p = maximum pressure head (L)
q_p = maximum infiltration rate (L/T)
h_m = minimum pressure head (L)
q_e = maximum evapotranspiration rate (L/T)
B_v = portion of the system boundary subject to a variable boundary condition

Physically, the maximum pressure head limit on the boundary prevents the generation of inappropriate surface water mounding. The minimum pressure headrestraint keeps the evaporation process from drying the soil near the boundary to moisture levels lower than residual saturation levels. The variable boundary condition can be used to approximate seepage faces within the studied area.

Internal source/sink terms, as represented by the term q ($L^3/T/L^3$) in Equation 3-1 are also accounted for in 3DFEMWATER. As with the boundary conditions, the source/sink terms can be constant or allowed to vary with time. Two source/sink options are available in the code. The first is a distributed source/sink option and the second is a point source/sink option.

The distributed source option is a source intensity that is integrated over the volume of an element. The user prescribes a source intensity, q_2 ($L^3/T/L^3$), or flux rate per unit volume for each distributed source element. This option allows a user modeling a large area to

approximate the influence of several wells within an element.

The point source/sink option is generally used to represent production or injection wells. Wells are represented as volumetric water fluxes, q_1 (L^3/T), applied at a nodal point or to better represent a screening interval, a column of nodal points (Figure 3.6). If vertically

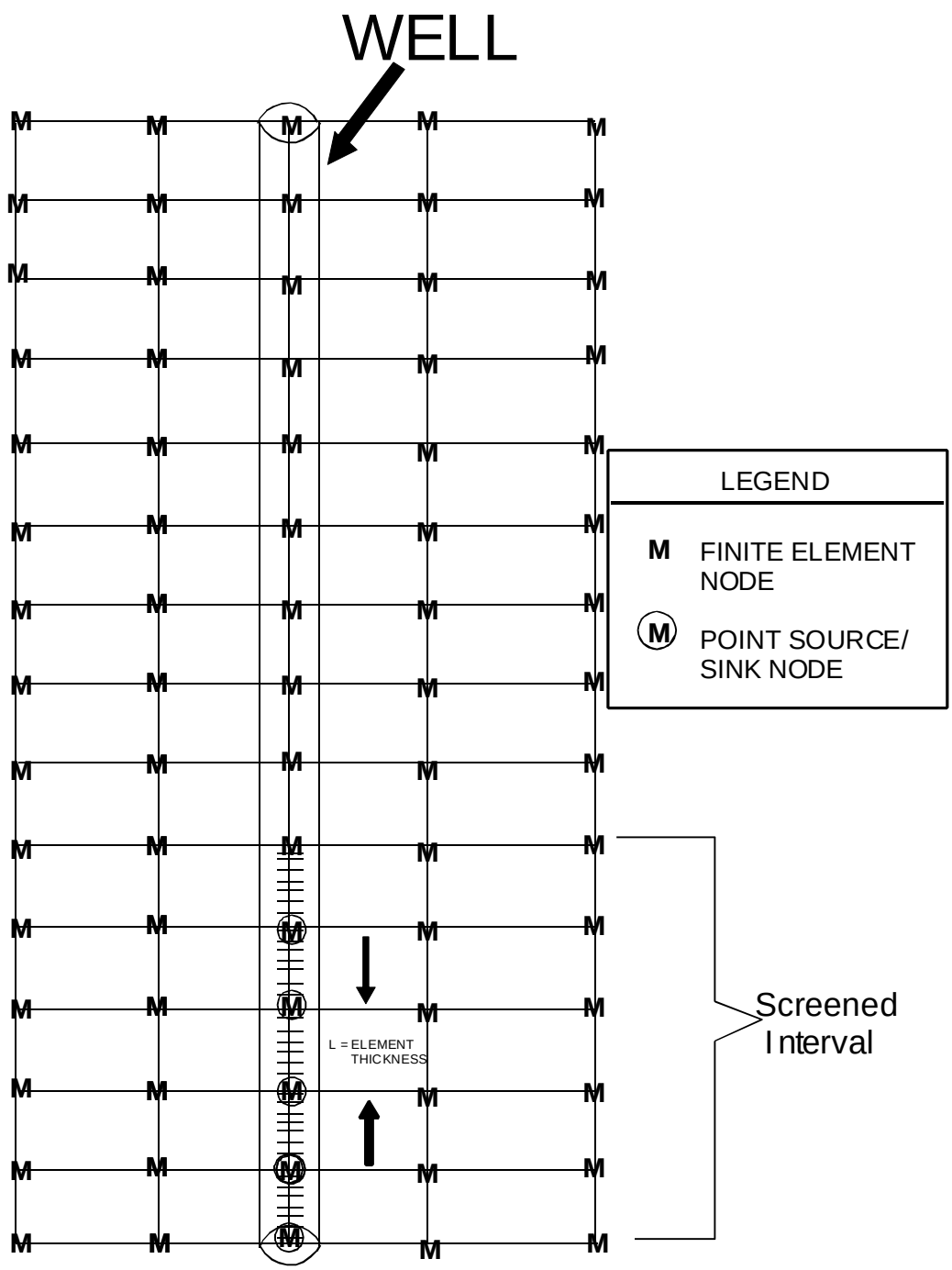

WELL
LEGEND
M FINITE ELEMENT NODE
M POINT SOURCE/ SINK NODE
L = ELEMENT THICKNESS
Screened Interval

Figure 3.6. Using a series of nodes to represent a screened well interval.

adjacent nodes are used to represent the screened interval of a well, the volumetric flux must be distributed among the nodes. The most appropriate distribution of the total flux is in proportion to the effective conductances, C_e, of the individual nodes where the effective conductance of each node is defined as:

$$C_e = 0.5\left[(K_s)_{n-1} L_{n-1} + (K_s)_n L_n\right] \qquad (3\text{-}9)$$

where n-1 and n are indices referring to the element below the node and the element above the node respectively and 0.5L is half the thickness of an element.

Time-variant boundary conditions and source/sink flux or flux intensity rates are defined by a series of paired time and value points. This paired data is used to assemble a look-up table from which appropriate values are obtained using linear interpolation at specified times of analysis. Constant values can be specified by assigning the same value to a set of two time/data point pairs, making sure that the simulation time is fully spanned.

3.1.3 Initial Conditions

The solution of Richard's equation also requires the initialization of pressure head values such that:

$$h = h_i(x,y,z,t=0) \text{ in } R \qquad (3\text{-}10)$$

where h_i is the initial pressure head distribution (L), and R is the region of interest (Figure 3.7). Besides providing a frame of reference for transient analyses, the initial conditions are used to set the nonlinear parameters at the beginning of a simulation. For transient problems, an appropriate set of initial pressure head values may either be input directly or derived from a steady-state simulation. For more information on these options see Section 4.1.11.

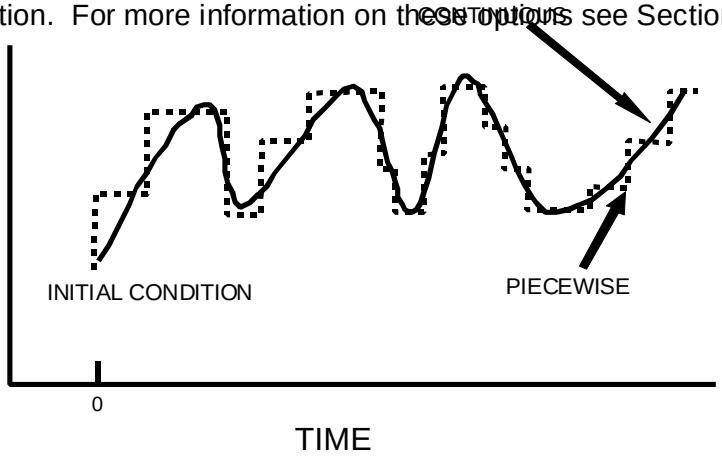


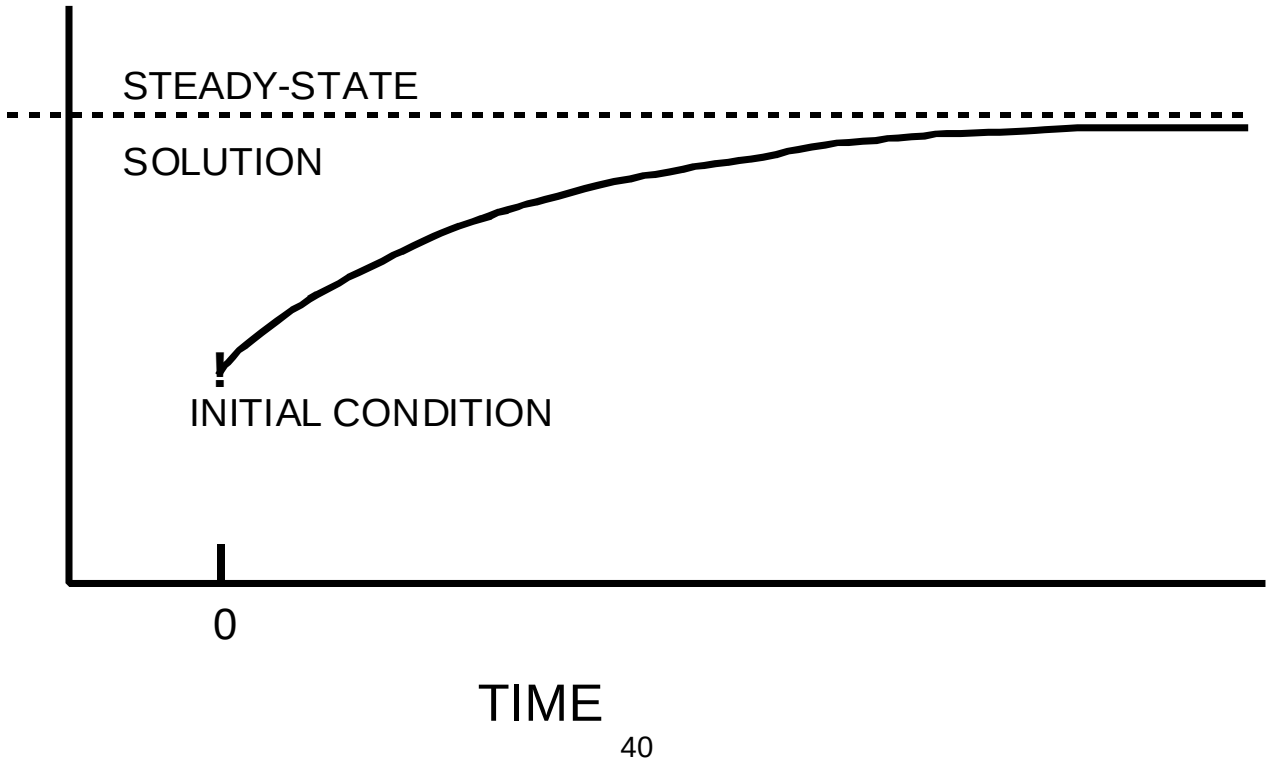
Figure 3.7. Pressure head versus time at a nodal point on the finite element grid.

3.1.4 Steady-State

When analyzing the influence of transient stresses, such as well production schemes and drought conditions, on the flow system, a starting point must be assumed. The user defines boundary conditions and flow parameters as best he/she can, then does an initial simulation to allow the system to reach an equilibrium or steady-state (Figure 3.8). The steady-state simulation then defines the pressure head at all points in the system and it is from this initial condition that a transient simulation is started. Although the actual system is never really in steady-state, by using averaged conditions (i.e., rainfall, etc.) a reasonable starting point is generated. If the steady-state simulation fails to converge or the results poorly match field data, flow parameters and/or boundary conditions should be adjusted to improve the starting conditions. The steady-state or equilibrium condition is generated by removing the temporal term from Equation 3-1. The system is then defined as the equilibrium reached under the average conditions.

Besides being used for initial conditions for a transient simulation, the steady-state flow option can also be used in conjunction with a transient transport simulation. Since the flow system will generally reach equilibrium under non-changing stresses faster than an associated solute transport problem, using a steady-state flow field and average conditions to define the advective portion of solute transport will often give a good approximation of the change of solute distribution over time. The savings in computational effort can be considerable and, given the uncertainty of parameters in the system, an acceptable approximation may be reached.

Figure 3.8. Pressure head versus time at nodal point where steady-state solution is being approached.



3.2 NUMERICAL APPROXIMATION IN 3DFEMWATER

The 3DFEMWATER model was developed to solve the variably-saturated flow equation described in Section 3.1. In the model, Richard's equation (Equation 3-1) is approximated using the Galerkin finite element technique. The time integral term in Equation 3-1 is approximated using backwards or central (Crank-Nicholson) difference in time. The nonlinearity of the system is treated using Picard iteration and the generated set of linearized equations is solved using a block iterative method.

3.2.1 Galerkin Formulation

In 3DFEMWATER, Richard's equation is approximated using the Galerkin finite element method (Pinder and Gray, 1977) where the dependent variable, pressure head, is approximated by a trial function of the form:

$$h \approx N_j(x_i,t)h_j(t) \qquad j = 1,2,\ldots,n \tag{3-11}$$

where $N_j(x_i,t)$ are the three-dimensional shape functions and $h_j(t)$ are nodal values of pressure head at time t for the n nodes of which the finite element grid is comprised (Figure 3.9).

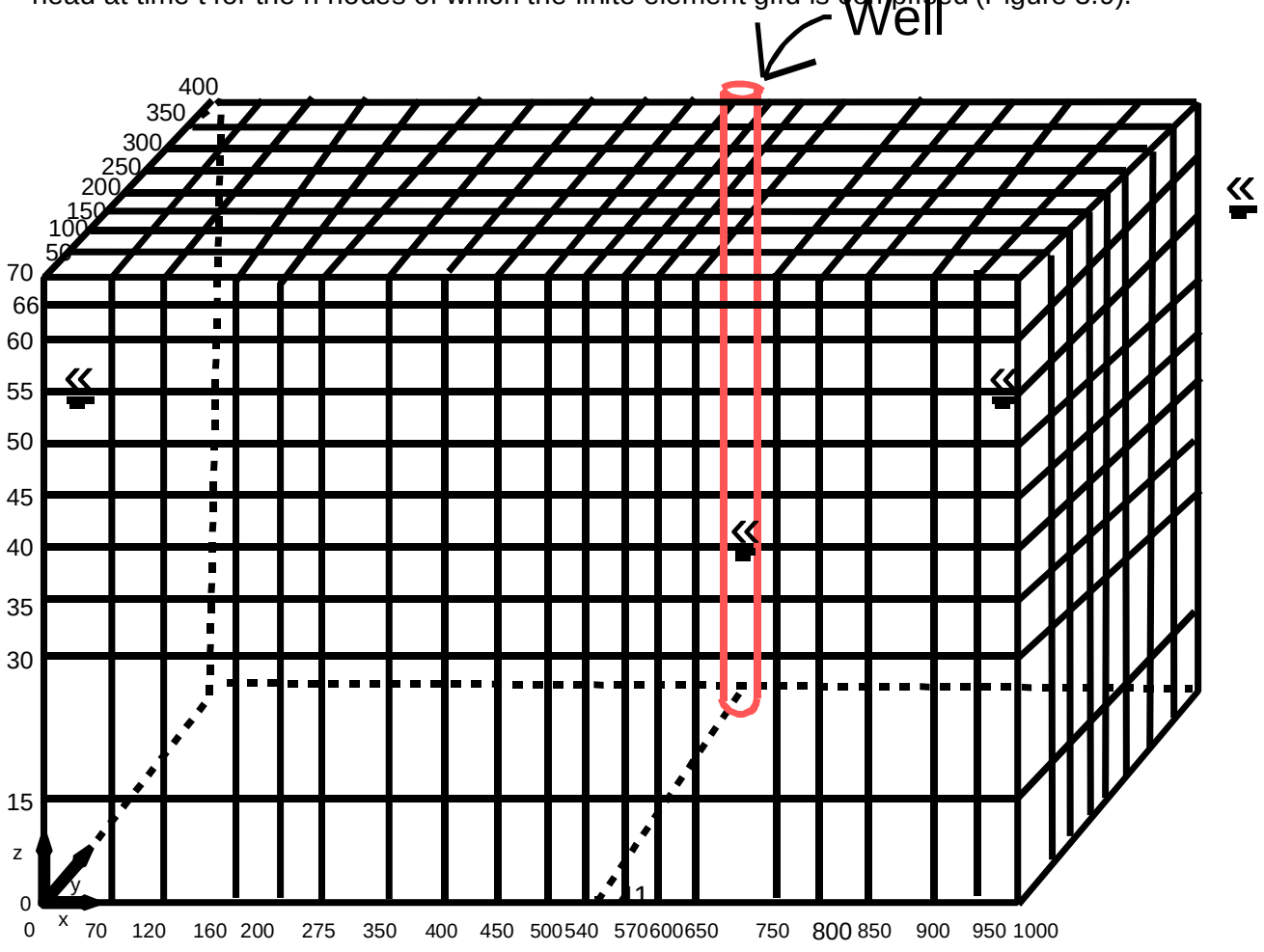


Figure 3.9. Finite element grid for production from a single well in a variably-saturated porous medium.

Substituting the trial functions into Equation 3-1 and applying the Galerkin criterion, a set of weighted residual minimization equations are generated of the form:

$$\int_{R_s} W_i \left[F(h) \frac{\partial \hat{h}}{\partial t} - \nabla \cdot [k_r K_s \nabla (L\hat{h} + Lz)] - q \right] dR = 0 \qquad (3\text{-}12)$$

where W_i are the weighting functions and R_s is the volume being simulated.

For the Galerkin method, the weighting functions are the same as the shape functions. Substituting $W_i = N_i$ and Equation 3-1 into Equation 3-12 results in:

$$(h) \frac{\partial (N_j h_j)}{\partial t} - \nabla \cdot \left[k_r K_s \nabla (h_j L N_j + Lz) \right] - q \Big] dR = 0, \quad \begin{matrix} i = 1,2, \\ j = 1,2, \end{matrix} \qquad (3\text{-}13)$$

where n is the number of nodes. Integration by parts can be used to rid Equation 3-13 of all second order derivatives, leaving a set of equations of the form:

$$\begin{aligned} &\int_{R_s} F(h) N_i \frac{\partial (N_j h_j)}{\partial t} dR + \int_{R_s} k_r K_s L N_i \nabla (h_j L N_j + Lz) dR \\ &- \int_{B_s} N_i \mathbf{n} \nabla k_r K_s \nabla (h_j L N_j + Lz) dB - \int_{R_s} N_i q dR = 0 \end{aligned} \qquad (3\text{-}14)$$

where B_s is the region boundary. The integrals given in Equation 3-14, which are taken over the entire region being modeled, can be replaced by the summation of integrals taken over the individual elements of which the finite element grid (Figure 3.9) consists. This finite element approximation generates a set of n nodal equations of the form:

$$A_{ij} \frac{dh_j}{\partial t} + B_{ij} h_j = C_i, \quad \begin{matrix} i = 1,2,...n \\ j = 1,2,...n \end{matrix} \qquad (3\text{-}15a)$$

where

$$A_{ij} = \sum_{k=1}^{m} \int_{R_e} F(h) N_i^e N_j^e dR \qquad (3\text{-}15b)$$

$$B_{ij} \;' \sum_{k'1}^{m} \int_{R_e} LN_i^e \partial k_r K_s \partial LN_j^e dR \qquad (3\text{-}15c)$$

and

$$\zeta_i = \sum_{k=1}^{m} \left[-\int_{R_e} k_r K_s \nabla N_i^e \cdot \nabla z \, dR + \int_{R_e} N_i^e q \, dR + \int_{B_s} N_i^e \eta \cdot k_r K_s \cdot (h_j \nabla N_j^e + \nabla z) \right] dt \qquad (3\text{-}15d)$$

where m is the number of elements into which the system is discretized and N^e denotes elemental shape functions.

3.2.2 Solution Techniques

To solve the series of linearized ordinary differential equations presented in Equation 3-15a, the time differential is replaced by a finite difference formulation, resulting in working equations for 3DFEMWATER of the form:

$$\frac{A_{ij}}{\Delta t_k} \left(h_j^{k+1} - h_j^{k} \right) + w B_{ij}^{k+w} h_j^{k+1} + (1-w) B_{ij}^{k+w} h_j^{k} = C_i^{k+w} \qquad (3\text{-}16)$$

where k+1 represents the current time level, k the previous time level, Δt the length of the current time step and w the time weighting function (1.0=backwards in time; 0.5=Crank Nicholson or centered in time). Note that the associated transport code, 3DLEWASTE, utilizes a backwards-in-time scheme. Therefore, when using 3DFEMWATER to generate a flow field for a 3DLEWASTE simulation, the backwards-in-time option should be used. This prevents the possibility of a mismatch in the interpolation of time-variant boundary condition and source/sink flux values.

For each time step, the solution method involves an outer and inner iterative scheme (Figure 3.10) where the outer iterations control convergence of the nonlinear terms in the equations and the inner iterative scheme controls the block-iterative method of solving the linearized set of equations. For each nonlinear iteration, the linearized set of equations is solved using relative permeability and storage terms updated using pressure head values generated during the previous nonlinear (outer) iteration. Relative permeability and storage terms for the first iteration in a time step are based on pressure head values from the previous time step, or for the first time step, from the initial conditions.

Because of the strong nonlinear nature of the soil moisture curves, the outer iterative scheme may become unstable. To help circumvent this problem it is often helpful to damp the iterative changes in the pressure head. One method of damping the iterative changes is through the use of an under-relaxation factor. Implementation of the under-relaxation factor for the outer iterations in 3DFEMWATER is as follows:

$$h_i^{r+1} = (1-u)h_i^r + uh_i^{r+1} \qquad (3\text{-}17)$$

where u is the outer under-relaxation factor and r is the iteration number. If damping is needed, values below one should be used. Acceleration or over-relaxation (1.0<u<2.0) is generally not recommended for the nonlinear iterations

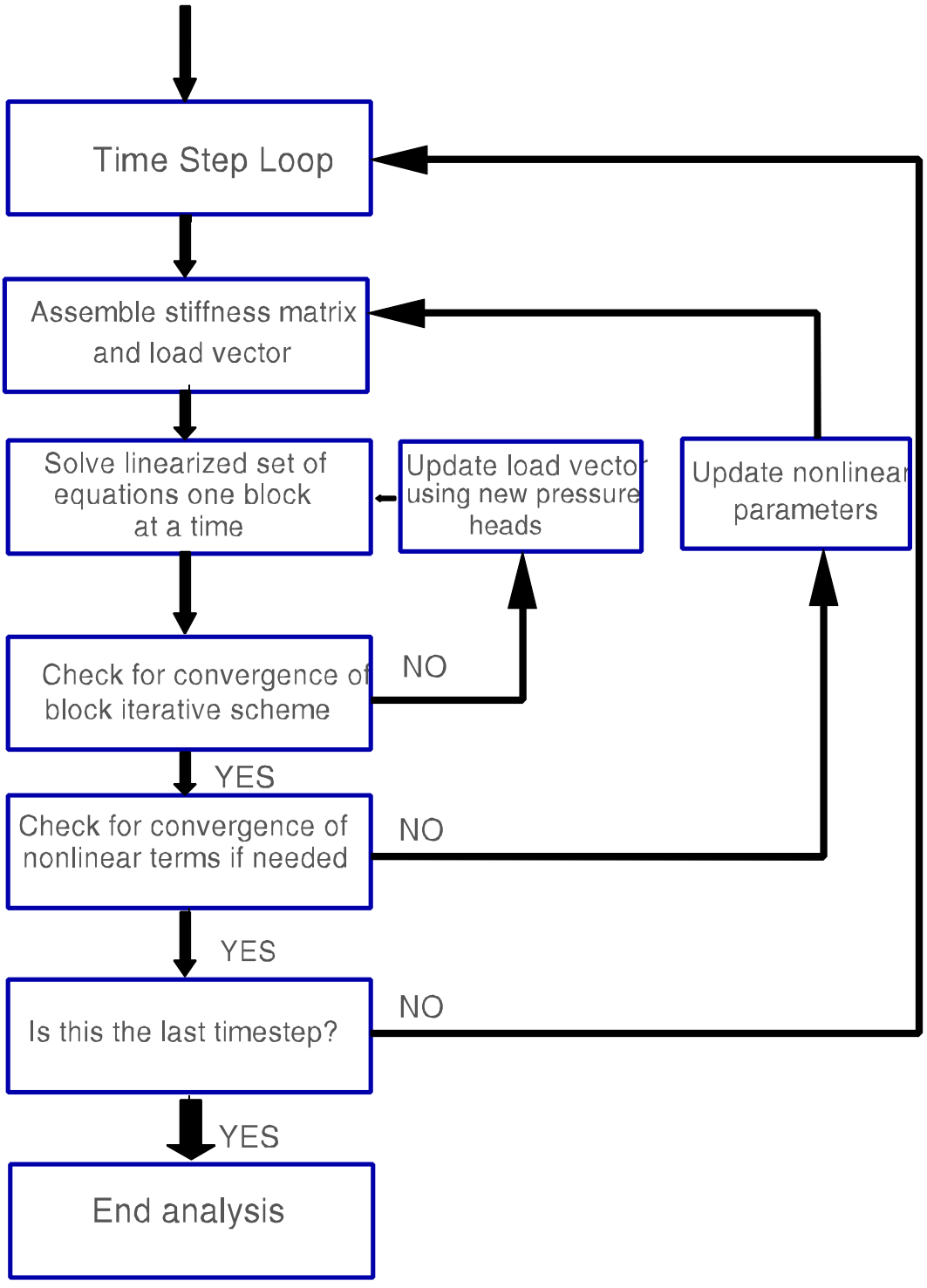

Time Step Loop
Assemble stiffness matrix and load vector
Solve linearized set of equations one block at a time
Update load vector using new pressure heads
Update nonlinear parameters
Check for convergence of block iterative scheme
NO
YES
Check for convergence of nonlinear terms if needed
NO
YES
Is this the last timestep?
NO
YES
End analysis

Figure 3.10.Solution scheme for unsaturated flow analysis.

as it may make the solution become unstable. For transient simulations, reduction of the time-step size can also help increase the stability of the solution scheme. Note that sometimes steady-state problems will be difficult to solve. In this case, it is often worth trying a transient solution approach, using expanding time steps to approach the steady-state solution.

For each nonlinear iteration, a set of linearized simultaneous equations is solved using a block iterative scheme. The user defines a set of subregions (or blocks) by prescribing the nodes contained in each subregion (Figure 3.11). The code then generates a series of connectivity arrays indicating: 1) the nodes contained in each subregion, 2) for each node, all other nodes found in elements it is part of, and 3) which of these adjacent nodes are located in the same subregion. The nodal equations for each subregion are solved directly using a Gaussian solver. For each nodal equation defined in Equation 3-15a, contributions from adjacent nodes falling outside the subregion being solved for are generated by multiplying the matrix terms with the appropriate nodal pressure heads. These pressure heads are generated during the last direct solution for the subregion containing the adjacent nodes.

Subregions are generally defined as nodal planes (Figure 3.11) allowing the user to work with a minimal half-bandwidth when the direct solver is invoked. The half-bandwidth is defined as one plus the largest difference between the node number associated with the nodal equation and the other nodes found in elements the node is part of and which are in the same block as the node. As a general rule, subregions comprised of vertical or sub-vertical nodal slices provide the smallest half-bandwidth and will perform well in the block iterative method, although this may not always be the case. For some problems, horizontal slicing may be advantageous. The block iterative logic contains a relaxation factor which can be used to over-relax the solution and help accelerate the rate of convergence. Implementation of the inner over-relaxation scheme is as follows:

$$h_i^{s+1} = (1-\omega)h_i^{s} + \omega h_i^{s+1} \qquad (3\text{-}18)$$

where s denotes the inner iteration number and o is the over-relaxation factor. The optimal value of the over-relaxation factor usually falls between 1.5 and 1.9. A good starting point is o = 1.72.

3.3 3DLEWASTE

3DLEWASTE is designed to simulate the movement of dissolved species through a variably-saturated porous medium. Typical applications for 3DLEWASTE include the examination of: 1) leachate migration from landfills and surface impoundments, 2) the influence on water quality of pesticide and fertilizer applications, and 3) the environmental impact of leaky containment structures such as underground and above ground storage tanks (Figure 3.12). Velocity fields needed to define the advective pathways of water bearing the chemicals are provided by associated 3DFEMWATER simulations.