CHAPTER 8

Finite Difference Solutions for Transient Problems

Numerical Methods in the Hydrological Sciences
Special Publication Series 57

doi:10.1029/057SP13

8. Finite Difference Solutions for Transient Problems

8.1. Background

Hydrological problems often involve time as a variable. The propagation of flood waves through river systems, the decline in water levels in aquifers as water is pumped from wells, the speading of a contaminant through a waterway or aquifer, and the drying of surface layers of soil by evapotranspiration are examples of problems where the variablesof interest – river stage, ground-water head, or soil moisture – vary temporally.

A particularly important type of equation used to describe *transient* (i.e., time-varying) problems in hydrology is the *heat equation* (Chapter 6.1). This equation describes not only heat flow in the natural environment (e.g., see Domenico and Schwartz 1998), but such things as diffusion of chemicals in sediments (e.g., Boudreau 1997), the movement of flood waves in channels using the diffusion analogy (e.g., Bras 1990), and the change in groundwater head in an aquifer (e.g., Fetter 2001).

8.2. Unidirectional flow in an aquifer

To illustrate the numerical methods used to solve the heat equation, consider a horizontal aquifer with constant formation properties, T (transmissivity) and S (storativity)[1]. The head in the aquifer is taken to vary with only one spatial coordinate, x. The equation that describes the variation of head in time and space is then

$$\frac{\partial h}{\partial t} = \frac{T}{S}\frac{\partial^2 h}{\partial x^2}. \qquad (8.1)$$

Further suppose that the aquifer is bounded by constant-head streams at x=0 and 2 km and that at some initial time the head in the aquifer is elevated at the center of the aquifer relative to the streams because of a recharge event. The head distribution is given by

$$h = 0.1x, \qquad \text{for } 0 \le x \le 1000 \text{ m}$$
$$h = 0.1(2000 - x), \quad \text{for } 1000 \text{ m} \le x \le 2000 \text{ m}.$$

The parameter T/S for the aquifer is 151.5 $m^2 s^{-1}$. The problem is to determine heads in the aquifer as a function of time assuming that recharge has ceased.

8.3. A forward difference (or *explicit*) approximation

Finite difference approximations for transient problems involve differencing both time and space derivatives. We will adopt the convention of using a "j" superscript to denote grid points in time and an "i" subscript to denote space grid points. Thus h_i^j denotes head at spatial grid point "i" and time grid point "j".

[1] For transient conditions, the derivation of the conservation equation shown in Box 6.1 must be adjusted to accommodate changes in time within the control volume. This leads to inclusion of the time derivative of head multiplied by a storage coefficient. See a text on hydrogeology, e.g., Fetter (2001), for more details.

One finite-difference approximation to equation (8.1) is formed by using the heads at time j in the approximation of the right-hand side:

$$\frac{h_i^{j+1} - h_i^j}{\Delta t} = \frac{T}{S}\left(\frac{h_{i+1}^j - 2h_i^j + h_{i-1}^j}{(\Delta x)^2} \right). \tag{8.2}$$

The way that we solve time-varying problems is to start at an initial time when conditions are known and then calculate conditions one time step into the future. Conditions at this step in the future are then "known" and one can proceed to calculate conditions at $2\Delta t$. And so forth. Thus, in general terms, conditions at time "j" are known and conditions at time "j+1" are to be calculated. Because the heads on the right side of equation (8.2) are at the "j" time, the approximation to the time derivative is a forward-difference approximation.

The unknowns in equation (8.2) form a vector $\left[h_1^{j+1}\ h_2^{j+1}\ h_3^{j+1} \ldots\ h_n^{j+1} \right]'$. We can write equation (8.2) in matrix-vector form as

$$h^{j+1} = W h^j \tag{8.3}$$

where W has diagonal entries equal to $\left(1 - 2\left(\frac{T\Delta t}{S(\Delta x)^2} \right)\right)$ and sub- and super-diagonal entires equal to $\frac{T\Delta t}{S(\Delta x)^2}$. The method is *explicit* because each component equation of the set of equations (8.3) is independent of the others. The equations are not linked directly, so no solution of a system of equations is required. Equation (8.3) requires only matrix multiplication.

Let's look at how the solution works. We choose a spacing in the x direction of 250 m and a time step of 206 s $\left[= (\Delta x)^2 S / 2T \right]$.

```
% Explicit solution to example one-D groundwater problem
% Set parameter values and grid spacing
dx=250; x=250:dx:1750; ToverS=151.5; dt=dx^2/(2*ToverS);
h_old=[0.1*x(1:4) 0.1*(2000-x(5:7))]';  %Initial head values
alpha=ToverS*dt/dx^2;   %Transmissivity divided by storage coefficient
M_diag=sparse(1:7,1:7,1-2*alpha,7,7);  %Diagonal of W matrix
L_diag=sparse(2:7,1:6,alpha,7,7);  %Off-diagonal of W matrix
W=M_diag+L_diag+L_diag';   %W matrix
hh=zeros(7,10);  %Pre-allocate matrix for solution
for j=1:10   %Do ten time steps
      h_new=W*h_old;          % The explicit solution
      hh(:,j)=h_new;
      h_old=h_new;
end
hh'
ans =
   25.0000   50.0000   75.0000   75.0000   75.0000   50.0000   25.0000
   25.0000   50.0000   62.5000   75.0000   62.5000   50.0000   25.0000
   25.0000   43.7500   62.5000   62.5000   62.5000   43.7500   25.0000
```

```
   21.8750   43.7500   53.1250   62.5000   53.1250   43.7500   21.8750
   21.8750   37.5000   53.1250   53.1250   53.1250   37.5000   21.8750
   18.7500   37.5000   45.3125   53.1250   45.3125   37.5000   18.7500
   18.7500   32.0312   45.3125   45.3125   45.3125   32.0312   18.7500
   16.0156   32.0312   38.6719   45.3125   38.6719   32.0312   16.0156
   16.0156   27.3438   38.6719   38.6719   38.6719   27.3438   16.0156
   13.6719   27.3438   33.0078   38.6719   33.0078   27.3438   13.6719
```

We can do the comparison with the analytical solution as well (Figure 8.1) and show that the approximate nature of the numerical solution is evident with the relatively coarse spatial grid spacing used in this example.[2]

```
% Analytical solution is an infinite Fourier series; see, e.g.,
% Carslaw and Jaeger 1959
time=0:dt:10*dt; fdtime=time(2:11);  % set time vector for solution
x1=250; x2=750;  % choose values of x for solution
n=0:25;   % use 25 terms in Fourier series solution
h250=zeros(1,11); h750=zeros(1,11);  % preallocate solution vectors
% compute the exact solution for 11 times
for j=1:11
     h250(j)=800*sum((ones(size(n))./(pi^2*(2*n+1).^2)).*cos((2*n+1).* ...
        pi*(x1/1000-1)/2).*exp(0.3738*(2*n+1).^2*time(j)/1000));
     h750(j)=800*sum((ones(size(n))./(pi^2*(2*n+1).^2)).*cos((2*n+1).*...
        pi*(x2/1000-1)/2).*exp(-0.3738*(2*n+1).^2*time(j)/1000));
end
plot(fdtime,hh(1,:),'or',fdtime,hh(3,:),'or',time,h250,'b',time,h750,'b')
xlabel('Time, seconds')
ylabel('Heads at 250m (lower curve) and 750m (upper curve)')
```

[2] The analytical solution is in the form of an infinite Fourier series. See Carslaw and Jaeger (1959).

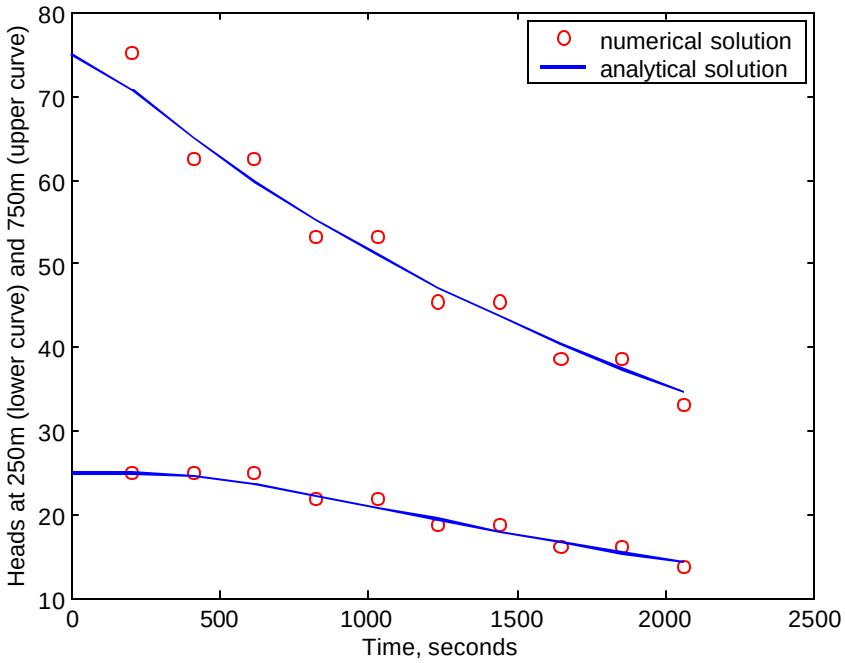

Figure 8.1. Heads calculated using explicit method with the analytical solution.

8.4. Stability problems with the explicit method

The explicit method is attractive because of its simplicity. To be useful, however, the time steps in the explicit method must be kept small. To illustrate the problem, let's look at the solution above for a Δt double that of our original choice, i.e. $\Delta t = (\Delta x)^2 S / T$.

```
% set time and space steps and parameters
dx=250; x=250:dx:1750; ToverS=151.5; dt=dx^2/(ToverS);
alpha=ToverS*dt/dx^2;
% place initial heads in vector h_old
h_old=[0.1*x(1:4) 0.1*(2000-x(5:7))]';
% construct the finite-difference matrix
M_diag=sparse(1:7,1:7,1-2*alpha,7,7);
L_diag=sparse(2:7,1:6,alpha,7,7);
W=M_diag+L_diag+L_diag';
% preallocate solution matrix
hh=zeros(7,10);
for j=1:10              % solve the equations for 10 time steps
      h_new=W*h_old;
      hh(:,j)=h_new;
      h_old=h_new;
end
hh(1:4,:)'

ans =
          25          50          75          50
```

```
   25          50          25         100
   25           0         125         -50
  -25         150        -175         300
  175        -350         625        -650
 -525        1150       -1625        1900
 1675       -3300        4675       -5150
-4975        9650      -13125       14500
14625      -27750       37275      -40750
-42375      79650     -105775      115300
```

Obviously, something is drastically wrong! The calculated heads are oscillating with absolute values getting ever larger with simulation time. This is a classic example of instability.

The way to study stability problems is to look at how errors (in computation there will *always* be small round-off errors) propagate through the solution. If the true solution to the finite-difference equations is given by H, the best that we can calculate is $H+e$, where e is the error. For one of the individual equations of (8.3), the true computation should be

$$H_i^{j+1} = rH_{i-1}^{j} + (1-2r)H_i^{j} + rH_{i-1}^{j} \tag{8.4}$$

where $r = \dfrac{T\Delta t}{S(\Delta x)^2}$. Our computation is imperfect, however, and is actually given by the equation

$$H_i^{j+1} + e_i^{j+1} = r\left(H_{i-1}^{j} + e_{i-1}^{j}\right) + (1-2\,r)\left(H_i^{j} + e_i^{j}\right) + r\left(H_{i+1}^{j} + e_{i+1}^{j}\right). \tag{8.5}$$

Subtracting (8.4) from (8.5), we find that the errors are governed by exactly the same equation as are the true heads:

$$e_i^{j+1} = r\left(e_{i-1}^{j}\right) + (1-2r)\left(e_i^{j}\right) + r\left(e_{i+1}^{j}\right). \tag{8.6}$$

A rigorous analysis of errors can be made in a number of ways. It turns out that we can get the right answer using a very simple analysis. We want a way to calculate a bound on the error at time j+1. Looking at equation (8.6), we ask, "What is the *worst* thing that could happen?" (It turns out that in computation, the worst thing always does happen.) Clearly, the error at time j+1 will be largest if e at the i-1 and i+1 spatial nodes have the opposite sign to the error at node i. Without loss of generality, we can assume that the errors are also equal in magnitude. In that case (8.6) becomes:

$$e_i^{j+1} = (1-4r)e_i^{j}. \tag{8.7}$$

Noting that this equation would serve as a bound for all values of j, we find that, after m time steps, the error is related to any initial error by

$$e_i^{j+1} = (1-4r)^m e_i^{0}. \tag{8.8}$$

We see that, if (1-4r) > 1 in magnitude, the error will grow as a power of the number of time steps. Such unbounded growth in error is what we mean by instability. The stability criterion for the explicit method is read directly from (8.8).

$$|1-4r| \le 1, \quad \text{or}$$

$$r \le \frac{1}{2}.$$

The stability restriction renders the explicit method cumbersome for many applications. For any approximation to be reasonably good, Δx must be kept small. Because Δt must be scaled by $(\Delta x)^2$, each time step is very small and a large number of computational steps may be required. If the stability restriction on the time step could be removed, Δt could be selected independently of Δx and the computation could be much more efficient. A modification to the finite-difference approximation accomplishes just that aim.

8.5. A backward difference (or *implicit*) approximation

To show how to avoid the rather strict stability criterion for explicit methods, let us revisit the finite difference equation (8.2) and see what happens if we use the unknowns at the j+1 time step to approximate the spatial derivative.

$$\frac{h_i^{j+1} - h_i^j}{\Delta t} = \frac{T}{S}\left(\frac{h_{i+1}^{j+1} - 2h_i^{j+1} + h_{i-1}^{j+1}}{(\Delta x)^2} \right). \qquad (8.9)$$

With this approximation, we are "looking backward" in time from the time step at which we take the spatial derivative, so the finite difference approximation in time is a backward difference. With this equation, if we place the "unknowns" (all heads at the j+1 time step) on the left-hand side of the equation and all the "knowns" (the heads at time j) on the right-hand side, we get

$$Ah^{j+1} = h^j, \qquad (8.10)$$

where the diagonal elements of A are (1+2r) and the sub- and super-diagonal elements are -r. The solution is implicit because a system of matrix-vector equations must be solved.

How about the stability? The same type of analysis that we used for the explicit equation gives the equation for the error as

$$-re_{i-1}^{j+1} + (1+2r)e_i^{j+1} - re_{i+1}^{j+1} = e_i^j.$$

The absolute worst that can happen in this case is for the errors at time j+1 for the i-1 and i+1 nodes to be equal to the error for node i. Then the error at the j+1 time step would be equal to the error at the j time step. No explosive growth. No instability. Even under the worst conditions. Thus, the implicit method is unconditionally stable for the ground-water equation. We can use any time steps that we want. Of course, large time steps will not result in a very accurate solution to the equation, but the solution will be stable.

How does the implicit method work in practice? The modification to the *MATLAB* code for the explicit method isn't very difficult.

```
% set time and space steps and parameters
dx=250; x=250:dx:1750; ToverS=151.5; dt=dx^2/(2*ToverS);
alpha=ToverS*dt/dx^2;
```

```
% place initial heads in vector h_old
h_old=[0.1*x(1:4) 0.1*(2000-x(5:7))]';
% construct the finite-difference matrix
M_diag=sparse(1:7,1:7,1+2*alpha,7,7);
L_diag=sparse(2:7,1:6,-alpha,7,7);
A=M_diag+L_diag+L_diag';
% preallocate solution matrix
hh=zeros(7,10);
for j=1:10                    % solve the equatios for 10 time steps
      h_new=A\h_old;          % solution uses the backslash command
      hh(:,j)=h_new;
      h_old=h_new;
end
hh'

ans =
   24.7423   48.9691   71.1340   85.5670   71.1340   48.9691   24.7423
   24.1471   47.1038   66.3301   75.9486   66.3301   47.1038   24.1471
   23.2616   44.7521   61.5390   68.7438   61.5390   44.7521   23.2616
   22.1756   42.1791   57.0369   62.8903   57.0369   42.1791   22.1756
   20.9760   39.5530   52.8777   57.8840   52.8777   39.5530   20.9760
   19.7308   36.9713   49.0482   53.4661   49.0482   36.9713   19.7308
   18.4872   34.4872   45.5190   49.4926   45.5190   34.4872   18.4872
   17.2755   32.1276   42.2606   45.8766   42.2606   32.1276   17.2755
   16.1137   29.9039   39.2467   42.5616   39.2467   29.9039   16.1137
   15.0115   27.8186   36.4551   39.5083   36.4551   27.8186   15.0115
```

The numerical solution is "smoother" for the implicit method than for the explict method (compare Figure 8.2 with Figure 8.1).

To illustrate that the solution is stable for larger values of r, rerun the example with r=1, the value that led to explosive instability with the explicit method. As the results on the next page show, the calculated heads "behave" even when r exceeds 0.5. It is the ability to take relatively large time steps that leads to the preference for the use of implicit methods in finite-difference solutions of ground-water-flow problems.

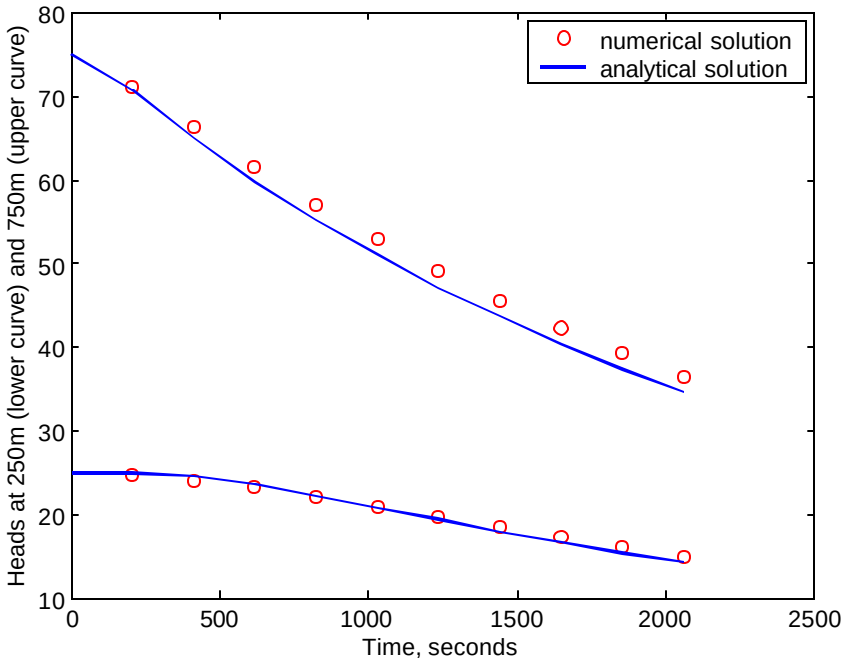

Figure 8.2. Heads calculated using implicit method with the analytical solution.

```
dx=250; x=250:dx:1750; ToverS=151.5; dt=dx^2/(ToverS);
h_old=[0.1*x(1:4) 0.1*(2000-x(5:7))]';
alpha=ToverS*dt/dx^2;
M_diag=sparse(1:7,1:7,1+2*alpha,7,7);
L_diag=sparse(2:7,1:6,-alpha,7,7);
A=M_diag+L_diag+L_diag';
hh=zeros(7,10);
for j=1:10
      h_new=A\h_old;
      hh(:,j)=h_new;
      h_old=h_new;
end
hh'

ans =
   23.9362   46.8085   66.4894   77.6596   66.4894   46.8085   23.9362
   22.0349   42.1684   57.6618   64.3278   57.6618   42.1684   22.0349
   19.7685   37.2706   49.8750   54.6926   49.8750   37.2706   19.7685
   17.4690   32.6386   43.1762   47.0150   43.1762   32.6386   17.4690
   15.3076   28.4537   37.4150   40.6150   37.4150   28.4537   15.3076
   13.3526   24.7502   32.4444   35.1679   32.4444   24.7502   13.3526
   11.6191   21.5048   28.1451   30.4860   28.1451   21.5048   11.6191
   10.0979   18.6745   24.4207   26.4425   24.4207   18.6745   10.0979
    8.7699   16.2120   21.1915   22.9418   21.1915   16.2120    8.7699
    7.6140   14.0721   18.3904   19.9075   18.3904   14.0721    7.6140
```

8.6. The g method

We can "mix and match" the explicit and implicit methods. That is, we can approximate the spatial derivative as g times the finite-difference approximation for the spatial derivative at the j+1 time step plus (1-g) times the approximation at time j. This results in a set of equations

$$Gh^{j+1} = Hh^{j},$$

where the diagonal elements of G are (1+2rg) and the sub- and super-diagonal elements are $-rg$, and the diagonal elements of H are [1-2r(1-g)] and the sub- and super-diagonal elements are r(1-g). When g=0, we recover the explicit method; if g=1, we get the implicit method.

For g=1/2, this method is known as the Crank-Nicolson method. Recall that the forward- and backward-difference approximations to a first derivative in time have truncation errors O(Δt) whereas the central-difference approximation has truncation error O((Δt)^2). The Crank-Nicolson approximation is, in a sense, a central-difference approximation in time. In fact, one can show that the truncation error for this case is O((Δt)^2).

8.7. Flow in an unsaturated soil

Problems involving flow in an unsaturated soil generally are attacked by solving the Richards equation [Box 8.1]. Such problems are quite difficult because the relationships between matric potential, y, and moisture content, q, and between hydraulic conductivity, K, and moisture content are highly nonlinear. The material in the following section is meant to serve as an introduction to how such problems can be approached.

Consider the problem of vertical flow of soil moisture. The Richards equation for this problem can be written (with "z" measured vertically downward from the surface)

$$\frac{\partial q}{\partial t} = \frac{\partial}{\partial z}\left[K(q)\frac{\partial y}{\partial z}\right] - \frac{\partial K(q)}{\partial z}. \qquad (8.11)$$

There are a number of ways for expressing the dependence of y and K on q. One of these is as follows.

$$s = \frac{q}{n}, \quad \text{where } n \text{ is the porosity;}$$

$$y(s) = y_{sat} s^{-1/m}, \quad \text{where } m \text{ is a parameter;}$$

$$K(s) = K_{sat} s^{c}, \quad \text{where } c \text{ is a parameter.}$$

where K_{sat} and y_{sat} are the values of these parameters for s=1. Given these forms for the nonlinear relationships, equation (8.11) can be written

$$\frac{\partial q}{\partial t} = \frac{\partial}{\partial z}\left[K_{sat}\left(\frac{q}{n}\right)^{c}\left(\frac{-y_{sat}}{m}\right)\left(\frac{1}{n}\right)\left(\frac{q}{n}\right)^{-1-1/m}\frac{\partial q}{\partial z}\right] - \left(K_{sat}c\left(\frac{1}{n}\right)\left(\frac{q}{n}\right)^{c-1}\right)\frac{\partial q}{\partial z}.$$

The equation is obviously nonlinear. There are no "standard" ways to solve such equations. Suffice it to say that solution is often achieved with considerable difficulty.

Solutions to nonlinear equations can be obtained by "linearizing" the nonlinear equations and accepting the solution to the linear equations as an approximation to the solution to the nonlinear equations. Often, iterative methods are used. One non-iterative approach for solving the nonlinear finite-difference soil moisture equations is a "predictor-corrector" method. In predictor-corrector methods, the nonlinear terms are taken out of the equations by substituting known values in these terms. The basic idea behind predictor-corrector methods is to eliminate the nonlinear terms by approximating them with values at the j^{th} time level, solve these equations to get a "predicted" value of the dependent variable at, say, time level $j+1/2$, use these "predicted" values to linearize the nonlinear terms in the original equation, and obtain "corrected" values by solving these equations.

To implement a predictor-corrector method for the soil-moisture equation, it is rewritten as:

$$\frac{\partial^2 q}{\partial z^2} = \frac{1}{\gamma}\frac{\partial q}{\partial t} - \left[\frac{u}{\gamma}\frac{\partial q}{\partial z} - \frac{w}{\gamma}\right]\frac{\partial q}{\partial z}$$

where γ, u, and w are functions of moisture content:

$$\gamma = K\frac{dy}{dq}, \quad w = \frac{dK}{dq}, \text{ and } u = \frac{dK}{dq}\frac{dy}{dq} + K\frac{d^2 y}{dq^2}.$$

Note that the γ, w, and u terms are the nonlinear terms in the equation; if these were *known* and constant (i.e., not functions of q), the soil-moisture equation would be linear. The equation is solved from t^j to t^{j+1} in two steps. First a step (the "predictor") is made to an intermediate point half way between t^j and t^{j+1}. Then, using the results from this half step, the solution is advanced to time t^{j+1} (the "corrector"). The predictor equation is:

$$\frac{q_{i-1}^{j+1/2} - 2q_i^{j+1/2} + q_{i+1}^{j+1/2}}{\Delta z^2} = \frac{1}{\gamma_i^j}\frac{q_i^{j+1/2} - q_i^j}{\Delta t/2} - \left[\frac{u_i^j}{\gamma_i^j}\left(\frac{q_{i+1}^j - q_{i-1}^j}{2\Delta z}\right) - \frac{w_i^j}{\gamma_i^j}\right]\left(\frac{q_{i+1}^j - q_{i-1}^j}{2\Delta z}\right)$$

and the corrector equation is:

$$\frac{1}{2}\left[\frac{q_{i-1}^{j+1} - 2q_i^{j+1} + q_{i+1}^{j+1}}{\Delta z^2} + \frac{q_{i-1}^{j} - 2q_i^{j} + q_{i+1}^{j}}{\Delta z^2}\right] =$$

$$\frac{1}{\gamma_i^j}\frac{q_i^{j+1} - q_i^j}{\Delta t} - \left[\frac{u_i^{j+1/2}}{\gamma_i^{j+1/2}}\left(\frac{q_{i+1}^{j+1/2} - q_{i-1}^{j+1/2}}{2\Delta z}\right) - \frac{w_i^{j+1/2}}{\gamma_i^{j+1/2}}\right]\left(\frac{1}{2}\right)\left(\frac{q_{i+1}^{j+1} - q_{i-1}^{j+1}}{2\Delta z} + \frac{q_{i+1}^{j} - q_{i-1}^{j}}{2\Delta z}\right)$$

The predictor uses an implicit method covering a time step $\Delta t/2$, linearized by approximating γ,w, and u at the j^{th} time level. The corrector is a Crank-Nicholson method covering the full time step – level j to level $j+1$ – and linearized by approximating γ, w and u using the results from the predictor, i.e., the $j+1/2$ values. (See Remson et al., 1971 for details.)

An implementation of the predictor-corrector in *MATLAB* may be helpful. Consider the problem of infiltration into an initially dry sandy loam soil (K_s=3.4 x 10^{-3} cm s^{-1}, ψ_s=-25 cm, n=0.25, m=5.4, c=3.4). Moisture content at the start of the infiltration event is equal to 0.10

everywhere. The moisture content at the soil surface is instantaneously raised to 0.25 and held there (e.g., by applying a pond of water in an infiltrometer). The time evolution of the moisture profile can be studied by integrating the Richards equation numerically.

```
% 3 function files precede the main code
function y=Kdpsi(param,theta)    % function "y"
Ks=param(1); c=param(2); n=param(3); psis=param(4); m=param(5);
y=(Ks*(-psis/m)*(theta/n).^(c-1-1/m))/n;
function w=dk(param,theta)       % function "w"
Ks=param(1); c=param(2); n=param(3); psis=param(4); m=param(5);
w=(Ks*c*(theta/n).^(c-1))/n;
function u=dKdpsi(param,theta)      % function "u"
Ks=param(1); c=param(2); n=param(3); psis=param(4); m=param(5);
u1c=Ks*c*(-psis/m);
u2c=Ks*psis*(1/m)*(1+1/m);
u=((u1c+u2c)*(theta/n).^(c-2-1/m))/n^2;
% main predictor-corrector loop (using the y, w, and u functions above)
Ks=3.4e-3; psis=-25; n=0.25; m=3.3; c=3.6;   %Values of parameters
param=[Ks c n psis m]; nz=30;
dz=100/nz; dt=60;                       %set time and space steps
ysat=-Ks*psis/m; wsat=Ks*c;
theta_old=ones(nz+1,1)*0.1;
theta_old(1)=0.25;
output=zeros(15,nz+1)';
time=zeros(19,1);
outkount=1;
for j=1:75                           % time loop
% predictor to advance one-half of the time step
   y=Kdpsi(param,theta_old(2:nz));
   w=dK(param,theta_old(2:nz));
   u=dKdpsi(param,theta_old(2:nz));
   bracketterm=((u./y).*(theta_old(3:nz+1)-theta_old(1:nz-1))/(2*dz)-w./y);
   mterm=-2*ones(nz-1,1)/dz^2-2*ones(nz-1,1)./(y*dt);
   M_diag=sparse(1:nz-1,1:nz-1,mterm,nz-1,nz-1);
   Lterm=ones(nz-1,1)./dz^2;
   Uterm=ones(nz-1,1)./dz^2;
   L_diag=sparse(2:nz-1,1:nz-2,Lterm(2:nz-1),nz-1,nz-1);
   U_diag=sparse(1:nz-2,2:nz-1,Uterm(1:nz-2),nz-1,nz-1);
   A=M_diag+L_diag+U_diag;
   rhs=-2*theta_old(2:nz)./(y*dt)-(bracketterm/(2*dz)).*...
      (theta_old(3:nz+1)-theta_old(1:nz-1));
   rhs(1)=rhs(1)-low(1)*theta_old(1);
   rhs(nz-1)=rhs(nz-1)-up(nz-1)*theta_old(nz+1);
   soln=A\rhs;
   thetahalf=[theta_old(1);soln;theta_old(nz+1)];
% corrector to advance to the next time
   y=Kdpsi(param,thetahalf(2:nz));
   w=dK(param,thetahalf(2:nz));
   u=dKdpsi(param,thetahalf(2:nz));
   bracketterm=((u./y).*(thetahalf(3:nz+1)-thetahalf(1:nz-1))/(2*dz)-w./y);
   Mterm=-ones(nz-1,1)/dz^2-ones(nz-1,1)./(y*dt);
   M_diag=sparse(1:nz-1,1:nz-1,d,nz-1,nz-1);
   Lterm=ones(nz-1,1)./(2*dz^2)-bracketterm./(4*dz);
```

```
    Uterm=ones(nz-1,1)./(2*dz^2)+bracketterm./(4*dz);
    L_diag=sparse(2:nz-1,1:nz-2,Lterm(2:nz-1),nz-1,nz-1);
    U_diag=sparse(1:nz-2,2:nz-1,Uterm(1:nz-2),nz-1,nz-1);
    A=M_diag+L_diag+U_diag;
    rhs=-theta_old(2:nz)./(y*dt)-(theta_old(1:nz-1)-2*theta_old(2:nz)...
       +theta_old(3:nz+1))/(2*dz^2)-bracketterm.*(theta_old(3:nz+1)...
       -theta_old(1:nz-1))/(4*dz);
    rhs(1)=rhs(1)-low(1)*theta_old(1);
    rhs(nz-1)=rhs(nz-1)-up(nz-1)*theta_old(nz+1);
    soln=A\rhs;
    theta_new=[theta_old(1);soln;theta_old(nz+1)];theta_old=theta_new;
    if rem(j,15)==0
       output(:,outkount)=theta_new;
       time(outkount)=dt*15*outkount; outkount=outkount+1;
    end
end
zz=0:-1/nz:-1; axis([0 0.26 -1 0]);
plot(output(:,1),zz,output(:,2),zz,output(:,4),zz)
xlabel('Volumetric moisture content');ylabel('Depth, meters')
time=time/60;
text(0.11,-0.6,['profiles for times ' num2str(time(1)) ' ' num2str(time(2))
' ' num2str(time(4)) ' minutes'])
```

The results of the computation (Figure 8.3) show that the moisture propagates into the soil with a fairly sharp front. In cases with a sharp front, time and space increments need to be kept small to avoid "glitches" in the solution. (Try the code above with nz=20, instead of 30, for example.) You should keep your skepticism intact when you are generating solutions to complex equations and not believe everything immediately as it comes out of the computer.

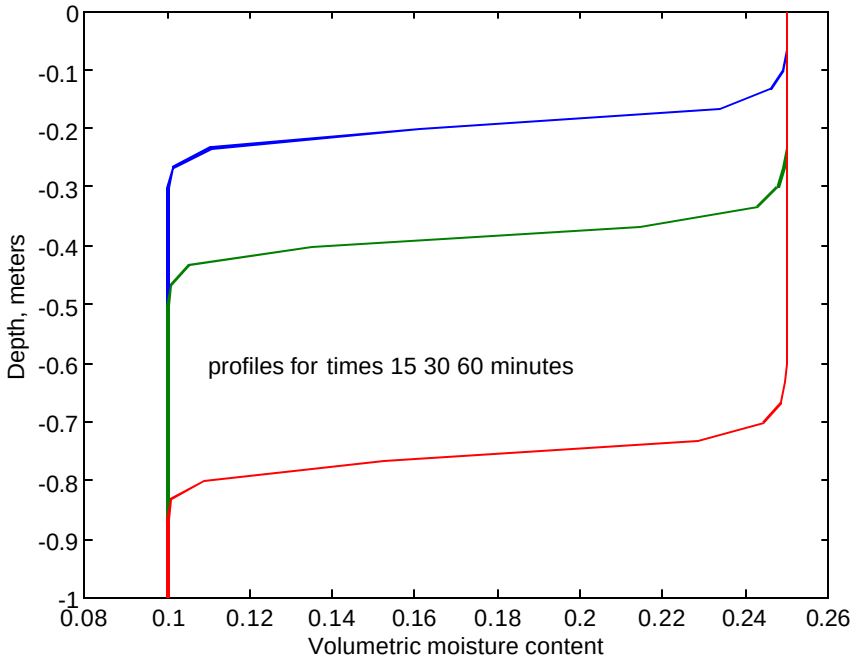

Figure 8.3 Solution to the Richards equation.

8.8. Problems

1. Heat flow in the earth is governed by the processes of conduction and convection. In regions where water is free to move, heat flow in the near surface (the top several hundred meters of the earth's surface) is strongly affected by convection and the analysis of temperature changes is quite complicated. In the arctic, however, permafrost essentially renders water motion meaningless as a heat-flow mechanism. In these areas, conduction is the primary mechanism by which heat is transported in the crust and a relatively simple analysis may be appropriate. The equation for such a problem is:

$$\frac{\partial T}{\partial t} = \frac{K}{\rho c}\frac{\partial^2 T}{\partial z^2}$$

where T is temperature, z is depth below the surface, t is time, K is thermal conductivity, c is heat capacity, and ρ is density.

Consider heat flow in the top 1km of the crust. The surface boundary condition is a specified temperature. The bottom boundary condition (at 1km) is that the upward heat flux, q, be equal to KG_0, where G_0 is the geothermal gradient, about 3°C per 100m. The thermal conductivity of rock and of permafrost is about 0.5 cal m^{-1} s^{-1} °C^{-1} and ρc is about 0.5 cal cm^{-3} °C^{-1}.

a) Write a code, using the γ method, to solve the temperature problem for permafrost regions. Explore the effect of changing grid spacing and γ.

b) Use your code to calculate the steady-state temperature profile for a surface temperature of -15°C. Start the computation with T=0°C everywhere.

c) Starting with the steady-state temperature profile as the initial condition, calculate the temperature profile every decade under conditions of a steadily increasing surface temperature at a rate of 3.5°C per century.

d) Mann et al. (1998) suggest that global surface temperature remained relatively steady for several centuries prior to the 20th century and then the temperature rose at a rate of about 0.5°C per century. Of course surface temperature trends at any locale can depart from the global mean trend. The table below gives measured temperatures from a borehole on the north slope of Alaska in 1984. Using your code, offer an interpretation of these data. By asking you to use your code, the intention is that you should be quantitative in your answer. You will want to run your code emphasizing the top tens of meters of the temperature profile – otherwise the fine detail of the changing temperature profile may be obscured by the coarse spatial discretization. [After you have finished this problem, you may want to look at the article by Lachenbruch and Marshall (1986) which is the source for these data. For a more up-to-date discussion of the analysis of borehole temperature profiles relative to climate, see Pollack and Huang (1998). For data from boreholes around the world, see http://www.ngdc.noaa.gov/paleo/borehole/borehole.html. Finally, in case you are prone to accept the results of the analysis of borehole temperatures uncritically, see Mann and Schmitt (2003) for a discussion of difficulties associated with inferring climate change from borehole temperature profiles.]

Depth below surface, meters	Temperature, degrees C
700	11.99
600	9.00
500	5.99
400	3.04
300	-0.01
200	-3.00
150	-4.49
125	-5.24
100	-5.92
90	-6.11
80	-6.37
75	-6.46
70	-6.50
65	-6.60
60	-6.69
55	-6.71
50	-6.75
45	-6.76
40	-6.78
35	-6.73

2. Explore the solution of the Richards equation for unsaturated flow. How are results sensitive to the time and space steps? How does the infiltration front progress for various soil types? (The values below are from Bras, 1990.)

SOIL TYPE	K_{sat} (cm s^{-1})	y_{sat} (cm)	n	m	c
Clay	3.4 x 10-5	-90	0.45	0.44	7.5
Silty loam	3.4 x 10-4	-45	0.35	1.2	4.7
Sand	8.6 x 10-3	-15	0.2	5.4	3.4

8.9. References

Boudreau, BP., *Diagenetic Models and their Implementation*, 414 pp., Springer-Verlag, Berlin, 1997.

Bras, R.L., *Hydrology*, 643 pp., Addison-Wesley, Reading, MA, 1990.

Carslaw, H.S. and J.C. Jaeger, *Conduction of Heat in Solids*, 2nd ed., 510 pp, Clarendon Press, Oxford, 1959.

Domenico, P.A. and F.W. Schwartz, *Physical and Chemical Hydrogeology*, 2nd ed., 506 pp., Wiley, New York, 1998.

Fetter, C.W. Jr., *Applied Hydrogeology*, 598 pp., Prentice-Hall, Upper Saddle River, NJ, 2001.

Lachenbruch, A.H. and B.V. Marshall, Changing climate: geothermal evidence from permafrost in the Alaskan Arctic, *Science, 234*: 689-696, 1986.

Mann, ME, Bradley, RS, and MK Huges, Global-scale temperature patterns and climate over the past six centuries. *Nature, 392*: 779-787. 1998.

Mann ME and GA Schmidt, Ground vs. surface air temperature trends: Implications for borehole surface temperature reconstructions, *Geophys. Res. Let. 30*: art. no. 1607, 2003.

Pollack, HN and S. Huang, Underground temperatures reveal changing climate, *Geotimes, 43*:16-19, 1998.

Remson, I., Hornberger, G.M., and F.J. Molz, *Numerical Methods in Subsurface Hydrology*. 389pp., Wiley-Interscience, New York, 1971.

Box 8.1. Equation for flow in an unsaturated soil

Water in the subsurface can be conveniently divided into *groundwater*, or water under pressure > 0, and *soil moisture*, or water under pressure < 0. That is, generally speaking, soil moisture is water above the water table (the surface defined by $p = 0$) that is under "tension" (under pressure less than atmospheric).

Water in the unsaturated zone, like water in the saturated zone, moves down a gradient in *head*. For soil moisture, the total head, h, is taken as the sum of the head due to gravity, $-z$ (for the z axis directed downward), and capillary pressure head or matric head, y.

$$h = y - z.$$

Matric head is a function of the moisture content. That is, in a very dry soil, capillary (and other) forces hold the water very strongly (high y) while in a moist soil the water is held less strongly (lower y). Matric head also depends on pore size, which generally scales with grain size, so that finer-grained sediment or soil has a larger matric head at a given level of saturation than coarse-grained soils . The relationship between matric head, y, and moisture content, q, for a given soil is known as the *matric characteristic*.

Darcy's law is used to describe the flow of water in the unsaturated zone. The law has the same form as for flow in saturated soils – specific discharge is proportional to the gradient in total head – but in the unsaturated zone, the hydraulic conductivity is a function of moisture content: $K=K(q)$. Darcy's law is then written

$$q = -K(q)\frac{dh}{dl}$$

where "l" represents a distance variable in the direction of flow.

For many applications we are concerned with the flow of soil moisture in the vertical, or "z" direction. In this case Darcy's law is (for z measured downward)

$$q_z = -K(q)\frac{dh}{dz} = -K(q)\frac{d}{dz}(y - z)$$

$$q_z = -K(q)\left[\frac{dy}{dz} - 1\right]$$

For the case of vertical flow of water, the appropriate continuity equation [Box 6.1] is:

$$\frac{\partial q}{\partial t} = -\frac{\partial q_z}{\partial z}$$

The left side of this equation represents the rate of change of mass in a small control volume, and the right side is the difference between the inflow rate and the outflow rate, each expressed on a per unit volume basis. Combining the continuity equation with Darcy's law results in the *Richards equation*.

$$\frac{\partial q}{\partial t} = \frac{\partial}{\partial z}\left[K(q)\frac{\partial y}{\partial z}\right] - \frac{\partial K(q)}{\partial z}.$$

Box 6.1. Groundwater flow equations

The basis of the equations used to describe the flow of groundwater is the conservation of mass equation, which, when applied to a fixed control volume, basically says that the rate of mass inflow minus rate of mass outflow equals rate of change of mass storage – what goes in minus what goes out equals the change in what's inside.

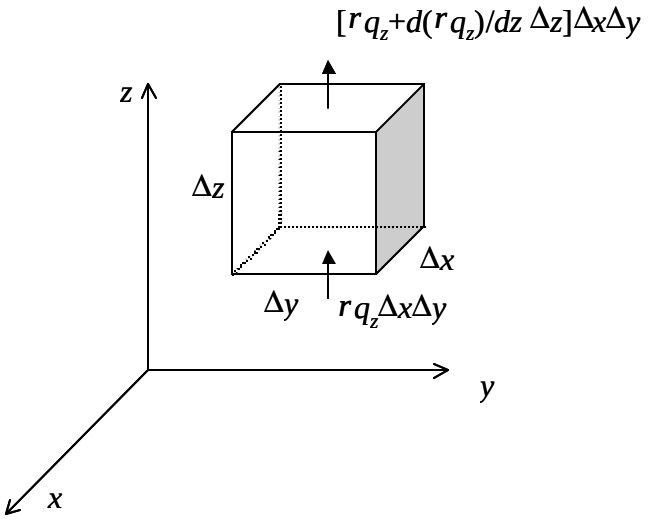

Figure B6.1.1. Control volume for deriving the conservation equation.

One way to derive the general equation of continuity for pore-fluid flow is to specify an arbitrary control volume to be a small rectangular parallelepiped in a fixed, Cartesian coordinate frame with sides of length Δx, Δy, and Δz (Fig. B6.1.1). Without any loss of generality, we take the directions designated by the arrows on the axes as positive (Fig. B6.1.1) and consider the case of positive flows. Consider first the inflow of mass into the control volume. The inflow into the parallelepiped in the z-direction is $\rho q_z \Delta x \Delta y$, where ρ is the density of water and q_z is the specific discharge (volumetric discharge per unit area) in the z direction. Density times the specific discharge gives the mass flux (mass per area per time) so multiplication by the area, $\Delta x \Delta y$, yields the mass inflow in the z direction. The mass flows in the x- and y-directions can be similarly calculated. Because specific discharge can change with distance, the value of q_z at the top face need not be the same as that at the bottom face. We can estimate the specific discharge at the top face using the Taylor series (Chapter 3.2). Because the distance separating the two faces (Δz) is as small as we wish to make it, we can get an acceptable approximation of the flux at the top face by just retaining the first two terms of the series (i.e., a linear extrapolation):

$$q_z(z+\Delta z) = q_z(z) + \frac{\partial q_z}{\partial z}\Delta z \qquad \text{(B6.1.1)}$$

The expression for the mass outflow in the z direction is then

$$\left[r\, q_z + \frac{\partial\, rq_z}{\partial z} \Delta z \right] \Delta x \Delta y \qquad \text{(B6.1.2)}$$

Now the expression that we need for the continuity equation is the **net** inflow of mass – the difference between the inflow and the outflow. For the z direction,

$$\text{net mass flow}_z = (rq_z(z) - rq_z(z+\Delta z))\Delta x \Delta y = -\frac{\partial rq_z}{\partial z} \Delta x \Delta y \Delta z$$

Using similar expressions for the x- and y-directions, the total net mass inflow can be obtained:

$$\text{total net mass inflow} = -\left[\frac{\partial\, rq_x}{\partial x} + \frac{\partial\, rq_y}{\partial y} + \frac{\partial\, rq_z}{\partial z} \right] \Delta x \Delta y \Delta z \qquad \text{(B6.1.3)}$$

For *steady* flow there can be no change of mass in the control volume so the net inflow must be zero. If we further assume that the density is constant, equation (B6.1.3) implies that

$$\frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} + \frac{\partial q_z}{\partial z} = 0 \qquad \text{(B6.1.4)}$$

The forces driving flow through an aquifer are due to gravity and pressure gradients. These forces, expressed on a per unit weight basis, are represented in groundwater flow equations in terms of a *head gradient*. Groundwater head is defined as pressure per unit weight plus elevation, which is the head due to gravity.

Darcy's law relates the specific discharge to the head gradient. Darcy's law states that this relationship is linear, with the constant of proportionality between specific discharge and head gradient being the *hydraulic conductivity, K*. If we make the assumption that the aquifer is *isotropic*, i.e., that K is independent of direction, Darcy's law can be written as follows.

$$\begin{aligned} q_x &= -K \frac{\partial h}{\partial x} \\ q_y &= -K \frac{\partial h}{\partial y} \\ q_z &= -K \frac{\partial h}{\partial z} \end{aligned} \qquad \text{(B6.1.5)}$$

Combining equations (B6.1.4) and (B6.1.5), we obtain an equation for steady groundwater flow.

$$\frac{\partial}{\partial x}\left(K \frac{\partial h}{\partial x} \right) + \frac{\partial}{\partial y}\left(K \frac{\partial h}{\partial y} \right) + \frac{\partial}{\partial z}\left(K \frac{\partial h}{\partial z} \right) = 0 \qquad \text{(B6.1.6)}$$

For the case of a *homogeneous* aquifer, one for which K is constant, equation (B6.1.6) reduces to the Laplace equation.

$$\frac{\partial^2 h}{\partial x^2}+\frac{\partial^2 h}{\partial y^2}+\frac{\partial^2 h}{\partial z^2}=0 \qquad \text{(B6.1.7)}$$

For the case of a horizontal aquifer of constant thickness, b, we assume that there is no vertical flow so equation B6.1.4 takes the form

$$b\frac{\partial q_x}{\partial x}+b\frac{\partial q_x}{\partial x}=0 \qquad \text{(B6.1.8)}$$

When (B6.1.8) is combined with Darcy's law, we obtain

$$\begin{aligned}&\frac{\partial}{\partial x}\left(Kb\frac{\partial h}{\partial x}\right)+\frac{\partial}{\partial x}\left(Kb\frac{\partial h}{\partial x}\right)=0\\ &\frac{\partial}{\partial x}\left(T\frac{\partial h}{\partial x}\right)+\frac{\partial}{\partial x}\left(T\frac{\partial h}{\partial x}\right)=0\end{aligned} \qquad \text{(B6.1.9)}$$

where T is the *transmissivity* of the aquifer. If there is recharge to the aquifer, e.g., by slow flow through an overlying confining layer, the equation is modified accordingly:

$$\frac{\partial}{\partial x}\left(T\frac{\partial h}{\partial x}\right)+\frac{\partial}{\partial x}\left(T\frac{\partial h}{\partial x}\right)=-w \qquad \text{(B6.1.10)}$$

where w is the recharge rate.

If the aquifer is homogeneous, T is constant and can be brought outside the derivative in (B6.1.9). The result is

$$\frac{\partial^2 h}{\partial x^2}+\frac{\partial^2 h}{\partial y^2}=0 \qquad \text{(B6.1.11)}$$

so again we find that the Laplace equation describes the steady flow of groundwater through a horizontal aquifer.

Finally, note that Darcy's law indicates that flow is down the *gradient* in head. This implies that flow lines for groundwater in an isotropic aquifer are perpendicular to lines of constant head. In *MATLAB* this means that if groundwater heads are computed and contour lines drawn, the `gradient` and `quiver` commands can be used to depict the flow.