CIVL 7170 HW-11 Richards' Equation
(Developed by Dr. Clement; Email: clement@auburn.edu)

Revised (May, 2013)

1. In this exercise we will simulate soil moisture movement in a one-dimensional 100 cm long column that received sufficient recharge to maintain the top layer at fully saturated conditions (i.e., ψ=0) for a period of 0.1 days. The bottom boundary is assumed to at the water table (i.e., ψ=0). After 0.1 days the recharge was cut down and no flow boundary conditions was maintained at the top up to 5 days. During this 5-day period the column was allowed to drain and moisture was redistributed over the column. Predict the transient distribution of capillary pressure and water content within the column at various times. Results are shown in Figs below, recreate these figures. Assume the column was initially at equilibrium with the bottom water table boundary (i.e., initial condition is: ψ=-Z). Use saturated conductivity of 1 cm/day, saturated water content (porosity) of 0.4, residual water content of 0.1, van Genuchten alpha of 0.01 cm^{-1}, and van Genuchten n of 2.

1. **Flow equation**

<u>Pressure Head from the Richards' equation:</u>

$$C(\psi)\frac{\partial \psi}{\partial t} = \frac{\partial}{\partial Z}\left[k(\psi)\frac{\partial \psi}{\partial Z}\right] + \frac{\partial k(\psi)}{\partial Z}$$

Define

$$F_1 = \frac{C_i^{n+1,m}}{\Delta t}$$

$$F_2 = \frac{k_i + k_{i+1}}{2\Delta z^2}$$

$$F_3 = -\frac{k_i + k_{i-1}}{2\Delta z^2}$$

$$F_4 = \frac{k_{i+1} - k_{i-1}}{2\Delta z}$$

Using F notation, the differential equation reduce to

$$F_1\psi_i^{n+1,m+1} - F_1\psi_i^n = F_2\left[\psi_{i+1}^{n+1,m+1} - \psi_i^{n+1,m+1}\right] + F_3\left[\psi_i^{n+1,m+1} - \psi_{i-1}^{n+1,m+1}\right] + F_4$$

$$-F_1\psi_i^n - F_4 = -F_3\psi_{i-1}^{n+1,m+1} + (-F_2 + F_3 - F_1)\psi_i^{n+1,m+1} + F_2\psi_{i+1}^{n+1,m+1}$$

2. **Boundary conditions**
   2.1 wetting cycle upto 0.1 days,
   @z=0, ψ=0, water table boundary

$$\psi_1^{n+1,m+1} = 0$$

@z=100, ψ=0, sufficient flow to maintain θ =n, when t ≤0.1days

$$\psi_n^{n+1,m+1} = 0$$

2.2 drainage cycle after 0.1 days, z=100cm, q_x=0

$$-\mathbf{k}(\boldsymbol{\psi})\left[\frac{\partial\boldsymbol{\psi}}{\partial\mathbf{z}}+\mathbf{1}\right]=\mathbf{0}$$

$$\frac{\partial\boldsymbol{\psi}}{\partial\mathbf{z}}=-\mathbf{1}$$

$$\boldsymbol{\psi}_{\mathbf{n+1}}-\boldsymbol{\psi}_{\mathbf{n}}=-\Delta\mathbf{z}$$

$$\boldsymbol{\psi}_{\mathbf{n+1}}=\boldsymbol{\psi}_{\mathbf{n}}-\Delta\mathbf{z}$$

$$-F_1\psi_n^n-F_4=-F_3\psi_{n-1}^{n+1,m+1}+(-F_2+F_3-F_1)\psi_n^{n+1,m+1}+F_2\psi_{n+1}^{n+1,m+1}$$

$$-F_1\psi_n^n-F_4=-F_3\psi_{n-1}^{n+1,m+1}+(-F_2+F_3-F_1)\psi_n^{n+1,m+1}+F_2(\psi_n^{n+1,m+1}-\Delta z)$$

$$-F_1\psi_n^n-F_4=-F_3\psi_{n-1}^{n+1,m+1}+(F_3-F_1)\psi_n^{n+1,m+1}-F_2\Delta z$$

$$-F_1\psi_n^n-F_4+F_2\Delta z=-F_3\psi_{n-1}^{n+1,m+1}+(F_3-F_1)\psi_n^{n+1,m+1}$$

<u>Code Details:</u>

```
Option Explicit
Private Sub unsaturated_Click()
Dim i As Integer, n As Integer, it As Integer, ntime, j As Integer
Dim a(600) As Double, b(600) As Double, c(600) As Double, d(600) As Double, v(600) As Double
Dim delx As Double, length As Double, delt As Double, t As Double
Dim f1 As Double, f2 As Double, f3 As Double, f4 As Double
Dim u() As Double, pu() As Double
Dim Kr() As Double, CC() As Double, theta() As Double

delx = 2#
length = 100#
n = length / delx + 1  'no of nodes
ReDim u(n), pu(n+1) , kr(n+1),CC(n+1),theta(n+1) As Double
delt = 1 / 100
ntime = 0.1 / delt

'Initial conditions
t = 0
Cells(1, 1) = "time/distance"
For i = 1 To n
   u(i) = -(i - 1) * delx
   Cells(1, i + 1) = (i - 1) * delx
   Cells(2, 1) = 0
```

```
   Cells(2, i + 1) = u(i)
Next i
Call water_content(n, u, theta)
For i = 1 To n
   Cells(5 + 5 / delt, 1) = 0
   Cells(5 + 5 / delt, i + 1) = theta(i)
Next i
'+++++++++++  start wetting (recharge conditions for 0.1 days)  +++++++++++++++++
For it = 1 To ntime
      t = t + delt
      Cells(2 + it, 1) = t
      Cells(5+5/delt+it,1) = t

       For i = 1 To n
             pu(i) = u(i)
       Next i

      For j = 1 To 5  'Note, we are using 5 Picard iterations
           ' The bottom boundary condition
           a(1) = 0
           b(1) = 1#
           c(1) = 0
           d(1) = 0

           'the upper boundary condition: sufficient flow to maintain saturated
           a(n) = 0
           b(n) = 1#
           c(n) = 0
           d(n) = 0

         call conduc(n , pu, Kr, CC)

           For i = 2 To (n - 1)
               f1 = 1 / (delt) * CC(i)
               f2 = 1 / (2 * delx ^ 2) * (Kr(i) + Kr(i + 1))
               f3 = -1 / (2 * delx ^ 2) * (Kr(i) + Kr(i - 1))
               f4 = 1 / (2 * delx) * (Kr(i + 1) - Kr(i - 1))

              a(i) = -f3
              b(i) = -f2 + f3 - f1
              c(i) = f2
              d(i) = -f1 * pu(i) - f4
           Next i

            Call tridia(n, a, b, c, d, v)

            For i = 1 To n ' transferring new values
```

```
                u(i) = v(i)
            Next i

      Next j
      Call water_content(n, u, theta)

      'Output the result
       For i = 1 To n
             Cells(2 + it, i + 1) = u(i)
             Cells(5+5/delt+it,i+1) = theta(i)
        Next i
Next it

'+++++++++++  start drainage (no flow BC at top)  +++++++++++++++++
ntime2 = 5 / delt
For it = (1 + ntime) to ntime2
   t = t + delt
   Cells(2 + it, 1) = t
   Cells(5+5/delt+it,1) = t

   For i = 1 To n
      pu(i) = u(i)
   Next i

   For j = 1 To 5
      ' The bottom boundary condition
      a(1) = 0
      b(1) = 1#
      c(1) = 0
      d(1) = 0

     pu(n+1) = pu(n)-delx           ' No flow at the n_th node

     Call conduc(n +1, pu, Kr, CC)

      For i = 2 To (n - 1)
             f1 = ??
             f2 = ??
             f3 = ??
             f4 = ??
         a(i) = ??
         b(i) = ??
         c(i) = ??
         d(i) = ??
      Next i

     'the upper boundary condition:
```

```
        a(n) = -f3
        b(n) = f3 - f1
        c(n) = 0
        d(n) = -f1 * pu(i) - f4 + f2 * delx

        Call tridia(n, a, b, c, d, v)

        For i = 1 To n ' transferring new values
           u(i) = v(i)
        Next i
   Next j
   Call water_content(n, u(), theta)

   'Output the result
   For i = 1 To n
      Cells(2 + it, i + 1) = u(i)
      Cells(5+5/delt+it,i+1) = theta(i)
   Next i

Next it
End Sub
```

Sub conduc(n as Integer, pu() as Double, Kr() as Double, CC() as Double)

```
'Se : van-Genuchten water retention'
'Kr : Mualem relative hydraulic conductivity function
'CC : water capacity fuction
Dim teta As Double, tetar As Double, tetas As Double
Dim alpha As Double, nn As Double, mm As Double
Dim i As Integer
Dim Se As Double

Call para(nn, mm, alpha, tetas, tetar)
For i = 1 To n
    If (pu(i) > 0) Then
       Se = 1#
    Else
       Se = (1 + (alpha * (-pu(i))) ^ nn) ^ (-mm)
    End If
    Kr(i) = Se ^ (1 / 2) * (1 - (1 - Se ^ (1 / mm)) ^ mm) ^ 2
    CC(i) = (alpha) * mm * nn * (tetas - tetar) * Se ^ (1 / mm) * (1 - Se ^ (1 / mm)) ^ mm
Next i
End Sub
```

Sub water_content(n as integer, u() as double, theta() as double)

```
'compute water content,θ using capillary pressure head
```

```
Dim teta As Double, tetar As Double, tetas As Double
Dim alpha As Double, nn As Double, mm As Double
Dim i As Integer
Dim Se As Double

Call para(nn, mm, alpha, tetas, tetar)
For i = 1 To n
    If (u(i) > 0) Then
        Se = 1#
    Else
        Se = (1 + (alpha * (-u(i))) ^ nn) ^ (-mm)
    End If
   Theta(i) = Se*(tetas-tetar)+tetar
Next i
End sub
```

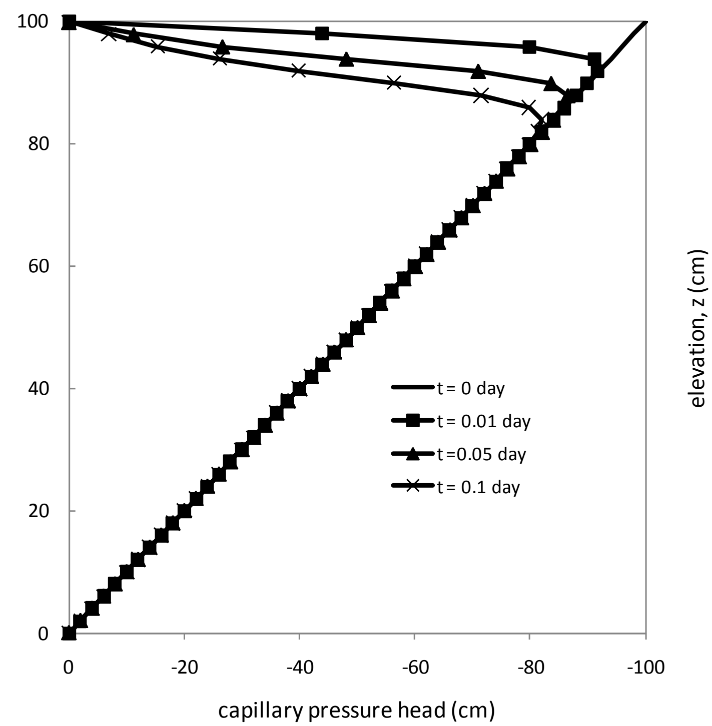
Sub para(nn as double, mm as double, alpha as double, tetas as double, tetar as double)

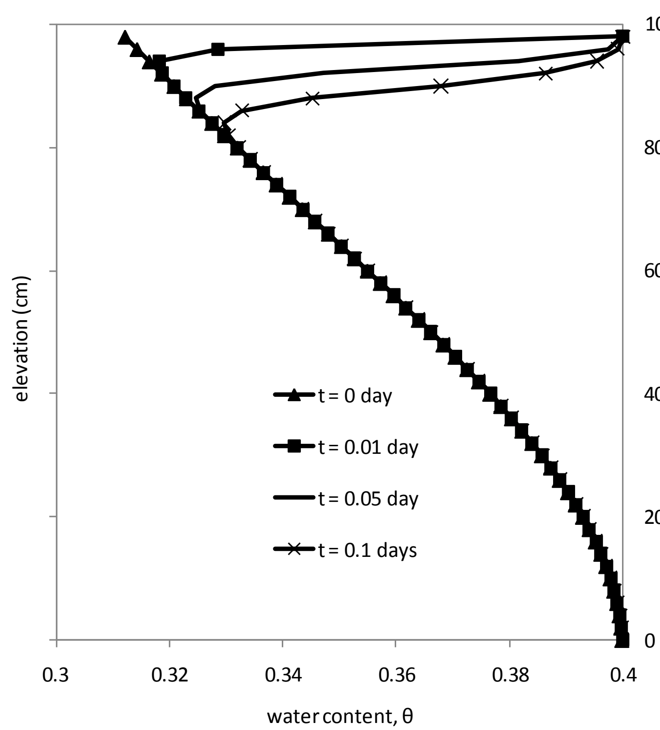
```
'This subroutine derines the soil parameters
alpha = 0.01
nn = 2#
mm = 1 - 1 / nn
tetas = 0.4
tetar = 0.1
End Sub
```

Sub tridia(n as integer, a() as Double, b() as Double , c() as Double, d() as Double, v() as Double)

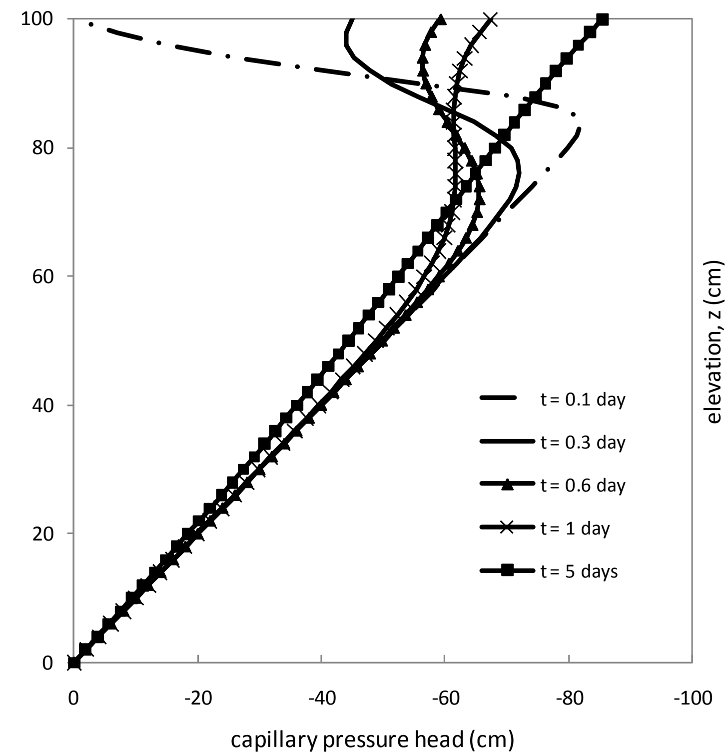
```
'solve system of tridiagonal matrix
Dim i, j As Integer
Dim ff(10000) As Double, b1(10000) As Double, d1(10000) As Double
For i = 1 To n
      b1(i) = b(i)
      d1(i) = d(i)
Next i
For i = 2 To n
      ff(i) = a(i) / b1(i - 1)
      b1(i) = b1(i) - c(i - 1) * ff(i)
      d1(i) = d1(i) - d1(i - 1) * ff(i)
Next i

v(n) = d1(n) / b1(n)

For i = 1 To n - 1
      j = n - i
      v(j) = (d1(j) - c(j) * v(j + 1)) / b1(j)
Next i
End Sub
```

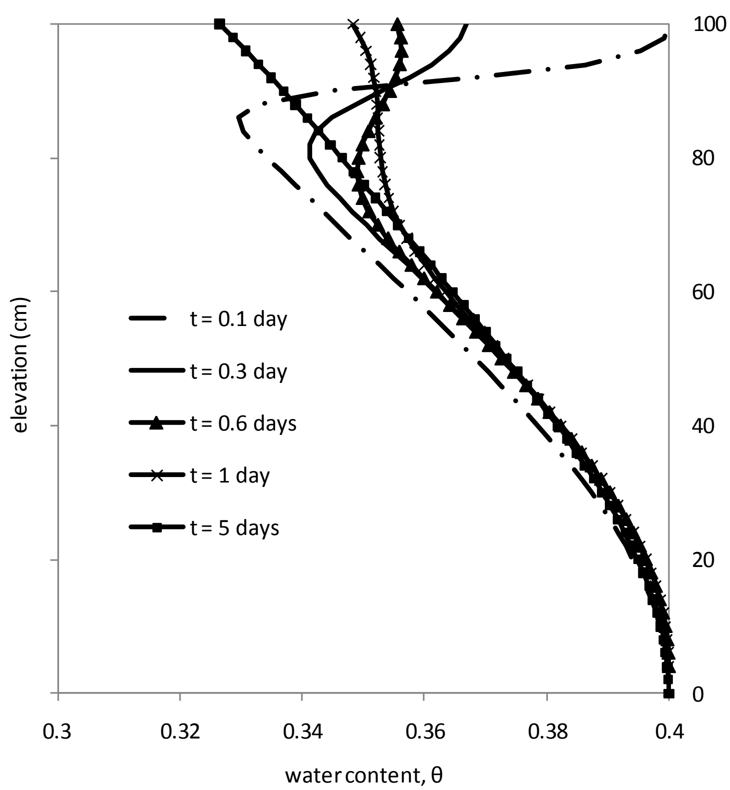
<u>Output: Wetting with rainfall</u>





Drainage with no flow boundary condition on the top





REFERENCE MATERIAL

(The following text provides the details of the water retention function (van Genuchten function) and relative permeability function, and soil water capacity function used in this exercise.

Van Genuchten' function

One of the most popular is van Genuchten's equation(van Genuchten, 1980):

$$\theta_e = [1 + (\alpha|\varphi|)^n]^{-m}$$

where:_ $\theta_e = (\theta - \theta_r)/(\theta_s - \theta_r)$, effective water content or effective saturation;
θ – water content; α, n, θ_r,m– equation parameters; θ_s – saturation
soil moisture; θ_r – residual soil moisture, and m = 1-1/n.

Soil Water Capacity (C)

$$\frac{\partial\theta}{\partial t} = \frac{\partial\theta}{\partial\varphi}\frac{\partial\varphi}{\partial t} = C(\varphi)\frac{\partial\varphi}{\partial t}$$

$$C(\varphi) = \frac{\partial\theta}{\partial\varphi}$$

Derivation of a model functional form for C(φ) using van Genuchten's function

$$\theta_e = [1 + (\alpha|\varphi|)^n]^{-m}$$

$$\frac{\theta - \theta_r}{\theta_s - \theta_r} = [1 + (\alpha|\varphi|)^n]^{-m}$$

$$\theta = \theta_r + (\theta_s - \theta_r)[1 + (\alpha|\varphi|)^n]^{-m}$$

Analytical form

$$C(\varphi) = \frac{\partial\theta}{\partial\varphi} = (\theta_s - \theta_r)\frac{d}{dh}[1 + (\alpha|\varphi|)^n]^{-m}$$

$$= \alpha mn\,(\theta_s - \theta_r)\theta_e{}^{1/m}(1 - \theta_e{}^{1/m})^m$$

Relative Hydraulic conductivity derivation

K = K_{sat} * K_r, where Ks = hydraulic conductivity at saturation, Kr = relative hydraulic conductivity.

$$K_r(\varphi) = \frac{\left[1 - (\alpha\varphi)^{n-1}(1 + (-\alpha\varphi)^n)^{-m}\right]^2}{\left[1 + (\alpha\varphi_i)^n\right]^{m/2}}$$

And

$$\theta_e = [1 + (-\alpha\varphi)^n]^{-m}$$

$$(-\alpha\varphi)^n - 1 = (\theta_e{}^{-\frac{1}{m}} - 1)^{1-\frac{1}{n}} = (\theta_e{}^{-\frac{1}{m}} - 1)^m$$

From these two equations,

$$K_r(\varphi) = \frac{\left[1-(-\alpha\varphi)^{n-1}(1+(-\alpha\varphi)^n)^{-m}\right]^2}{[1+(\alpha\varphi_i)^n]^{\frac{m}{2}}} = \theta_e{}^{1/2}\left[1-\left(\theta_e{}^{-\frac{1}{m}}-1\right)^m\theta_e\right]^2$$
$$= \theta_e{}^{1/2}\left[1-(\theta_e{}^{-\frac{1}{m}}-1)^m\left(\theta_e{}^{1/m}\right)^m\right]^2$$
$$= \theta_e{}^{1/2}\left[1-(1-\theta_e{}^{1/m})^m\right]^2$$

Therefore

$$K_r(\theta_e) = \theta_e{}^{1/2}\left[1-\left(1-\theta_e{}^{1/m}\right)^m\right]^2$$

Water retention curve , θ(ψ) and relative hydraulic conductivity, K(θ) from Van Genuchten model.

Below are the Van Genuchten model parameters used for the test problem.
$\alpha = 0.01$; $n = 2$; $m = 1 - 1/n$; $\theta s = 0.4$; $\theta r = 0.1$
where, θs is a saturated soil-water content and θs is a saturated soil-water content

$$\theta = \theta_r + (\theta_{s-}\theta_r)[1+(\alpha|\varphi|)^n]^{-m}$$

$$K_r(\theta_e) = \theta_e{}^{1/2}\left[1-\left(1-\theta_e{}^{1/m}\right)^m\right]^2$$