308 --- Q200002170006

Scientific Notebook # 343

LABORATORY NOTEBOOK

CNWRA/SwRI

Scientific Notebook # 343 --- Q200002170006

NOTEBOOK NO. 343

ISSUED TO ILYA S. PARTOM I.P.

ON David R. Lunn 19 DBL

DEPARTMENT 20

RETURNED 19

—SCIENTIFIC NOTEBOOK CO.—
2831 LAWRENCE AVE.
P.O. BOX 238
STEVENSVILLE, MI 49127
616-429-8285

INSTRUCTIONS

1. **The primary purpose of this notebook is to protect your and the Company's Patent-Rights by keeping records of all original work in a form acceptable as evidence if any legal conflict arises.**

2. • When starting a page, enter the title, project number, and book number.
   • Use ink for permanence -- avoid pencil.
   • Record your work as you progress, including any spur-of-the-moment ideas which may be developed later.
   • Avoid making notes on loose paper to be recopied.
   • Record your work in such a manner that a co-worker can continue from where you stop. You might be ill and to protect your priority it could be urgent that the work continue while you are absent.

3. • Give a complete account of your experiments and the results, both positive and negative, including your observations.
   • Record all diagrams, layouts, plans, procedures, new ideas, or anything pertinent to your work including the details of any discussions with suppliers, or other people outside the Company.
   • Do not try to erase any incorrect entries; draw lines deleting them, note the corrections, sign and date the changes. This extra care is worthwhile because of the necessity of original data to prove priority of new discoveries.

4. • After entering your data, sign and date the entries.
   • Explain your work to at least two witnesses who are not co-inventors, and have them sign and date the pages in the place provided.
   • Record the names of operators and witnesses present during any demonstration and have at least two witnesses sign the page. If no witnesses are present during an experiment of importance, repeat it in the presence of two witnesses.

5. Since computer programs can be patented these instructions apply to the development of computer software. In this case a description of the structure and operation of the program should be recorded in the notebook, together with a basic flow diagram which illustrates the essential features of the program. In the course of developing the code, the number of lines of code written each day should be recorded in the notebook, together with a statement of the portion of the flow diagram to which the section of code is directed.

6. This notebook and its contents are the exclusive property of the Company. It is confidential and the contents are not to be disclosed to anyone unless authorized by the Company. You must return it when completed, upon request, or upon termination of employment. It should be kept in a protected place. **If loss occurs, notify your supervisor immediately, and make a written report describing the circumstances of the loss.**

Table of Contents	Page
Work performed by Ilya Portam, CNWRA consultant	
Summer 1999	
David R. Turner, Principal Scientist	DRT
WBS 20.01402.761	

SCIENTIFIC NOTEBOOK CO.

TITLE INITIAL ENTRIES Project No. Book No. 1

From Page No.

6/10/1999

We're developing a time dependent 1D code for vertical water, radionuclides, and colloids transport through the unsaturated zone. We call this code COLLOID. We outlined the motivation for developing the code, the purpose of the code, and its content in a letter to Dr. David Turner, dated 5/18/1999. Part of this letter is copied below, after changing the notation to match the notation in the code.

Dear Dr. David Turner.

After some reading I think I now have some understanding of two things:
1- What is the problem.
2- How to try to model the problem.

In what follows I am explaining this.

1- What is the problem.
In field tests at NTS they recently found that radionuclides (mainly Pu) are transported significantly faster than thought before. The standard explanation is that radionuclides are transported by clay colloids to which they are sorbed.

2- How to model the problem.
I propose to develop a 1-D time dependent flow and transport code (a vertical column).

At t=0 the following quantities are given along the column:
S = Degree of saturation.
CNW = Concentration of radionuclides in water.
CCW = Concentration of colloids in water.
CNS = Concentration of radionuclides sorbed to non-colloidal matrix.
CNWC = Concentration of radionuclides sorbed to colloids that are in the water.
CNSC = Concentration of radionuclides sorbed to colloids that are in the matrix.
CCS = Concentration of colloids in matrix.

DRTurner 2/7/00

All these concentrations are given appropriate initial conditions at t=0.

To Page No.

Witnessed & Understood by me,	Date	Invented by	Date
		Recorded by	

At the entrance end, we assume that R(t) (rainfall rate) is given.
There is enough waste so that the water penetrating to the soil is saturated with radionuclides.

The code will include the following features:
Unsaturated /saturated flow of water.
Transport of dissolved radionuclides.
Transport of colloids.
Movement of colloids from the medium to the water.
Sorption of radionuclides from water by colloids.
Sorption of radionuclides from water by the immobile medium.

At the exit end we monitor the outflow of water and radionuclides. From the breakthrough curves obtained we will (hopefully) deduce transport coefficients.

If we are successful with this, we can then go on to do a similar code for flow and transport in fractures.

General description of the code

COLLOID is a 1D time dependent code that simulates vertical flow of water and transport of radionuclides and colloids in a column of unsaturated/saturated medium.

The code uses finite differences, and the time integration is done explicitly.

We write the code in FORTRAN 90 with DIGITAL FORTRAN.

We divide the column (z direction) into nc equal cells of length dz. We take z positive downwards, where z=0 is at ground level.

For each cell along the column we have the following variables:
- z(i) = location of cell upper boundary.
- v(i) = water velocity at cell upper boundary.
- vnw(i) = radionuclides dispersion velocity at cell upper boundary.
- s(i) = degree of saturation, at cell center.

- hp(i) = pressure head, at cell center.
- cnw(i) = radionuclide concentration in water, at cell center.
- cns(i) = radionuclide concentration in matrix, at cell center.
- cnwc(i) = concentration of radionuclides attached to colloids in water, at cell center.
- cnsc(i) = concentration of radionuclides attached to colloids in matrix, at cell center.
- ccw(i) = colloid concentration in water, at cell center.
- ccs(i) = colloid concentration in matrix, at cell center.

The units we use for these variables are:
- Time: year (y).
- Length: meter (m).
- Mass: milligram, kilogram (mg), (kg).

The units of the variables above are given at the beginning of the code.

The first thing we do in the code is initializing all the above variable

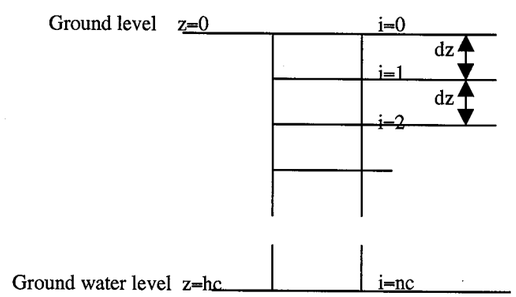


Fig. 1. Column geometry.

6/11/1999

To run the code we also need to input certain fixed data. These are:

- pore = matrix porosity.
- hc = column height.
- dz = cell length.
- dt = integration time increment.
- masb = type of upper boundary condition for the flow. We have (until now three types: 1- constant head; 2- given precipitation rate; 3- finite duration constant head.
- hub = upper boundary head in masb=1 or 3.
- t1 = duration of constant head in masb = 3.
- tmax = maximum running time.
- cnwaq = concentration of radionuclides in the aquifer below the column.
- cnwm = maximum concentration of radionuclides in water.
- cnsm = maximum concentration of radionuclides in matrix.
- ccwout = concentration of colloids in incoming water.
- ccwaq = concentration of colloids in aquifer water.
- ccwm = maximum concentration of colloids in water.
- ccsm = maximum concentration of colloids in matrix.
- tpr = initial output time.
- dtpr = output time step.
- dpor = average pore diameter in the matrix.
- ddn = coefficient of molecular diffusion of radionuclides in water.
- apec, bpec = parameters in effective dispersion versus Peclet number function.

The above fixed data are input to the code by DATA statements. In the future we'll input them from a file using the NAMELIST statement.

The units of the above parameters are given at the beginning of the code.

6/14/1999

To calculate the flow velocities (v) we use Darcy's law.

$$v = -k(s)\left(-1 + \frac{\partial h}{\partial z}\right) \qquad (1)$$

where:
s= saturation.
k= hydraulic conductivity.
h= piesometric head.
The –1 term is for the effect of gravity, taking into account that the z axis is positive downwards.

For unsaturated flow $h=-\psi(s)$, where ψ is the suction (negative pressure).

For $\psi(s)$ we use:

$$\psi = \psi_R\left(\frac{s - s_{min}}{1 - s_{min}}\right) + \psi_L\left(\frac{1 - s}{1 - s_{min}}\right) \qquad (2)$$

where:
s_{min}= minimal saturation.
$\psi_R = \psi(1)$.
$\psi_L = \psi(s_{min})$.

For k(s) we use:

$$k = A(s - s_{min}) + B(s - s_{min})^2 \qquad (3)$$

The conservation equation for fluid volume is:

$$\frac{\partial s}{\partial t} = -\frac{\partial (sv)}{\partial z} \qquad (4)$$

For the dispersion velocity of the nuclides (v_{nw}) use Fick's law:

$$v_{nw} = -D_{eq}\frac{\partial c_{nw}}{\partial z} \qquad (5)$$

where:
D_{eq}= equivalent dispersion coefficient of the radionuclides in the ~~matrix~~ water, and is given by:

$$D_{eq} = a_{pec}D_d + b_{pec}vd_{av} \qquad (6)$$

where:
D_d= molecular diffusion coefficient of the radionuclides in water.
d_{av}= average pore diameter.
a_{pec}, b_{pec}= nondimensional parameters.

For the conservation equation of the radionuclides (ignoring temporarily sorption and decay) we use:

$$\frac{\partial(sc_{nw})}{\partial t} = -\frac{\partial(sv_{nw})}{\partial z} - \frac{\partial(svc_{nw})}{\partial z} \qquad (7)$$

For the colloids in water we use the conservation equation (ignoring temporarily retardation effects):

$$\frac{\partial(sc_{cw})}{\partial t} = -\frac{\partial(svc_{cw})}{\partial z} \qquad (8)$$

where:
c_{cw}= concentration of colloids in water.

6/15/1999

To take into account the precipitation of colloids on the matrix, we assume that there is equilibrium between the colloids in the water and the colloids in the matrix. This is equivalent to assuming that the time scale to reach equilibrium is much shorter compared to the time scale of colloid transport. The equilibrium equation is:

$$\frac{\rho}{\theta s}c_{cs} = k_c c_{cw} \qquad (9)$$

where θ is the porosity, and k_c is a coefficient to be determined from laboratory tests.

Mass conservation for colloids in the water becomes:

$$\frac{\partial(sc_{cw})}{\partial t} = -\frac{\partial(svc_{cw})}{\partial z} - \frac{\partial\left(\frac{\rho}{\theta}c_{cs}\right)}{\partial t} \qquad (10)$$

Substituting Eq. (9) into Eq. (10) we get:

$$R_c\frac{\partial(sc_{cw})}{\partial t} = -\frac{\partial(svc_{cw})}{\partial z} \qquad (11)$$

where:

$$R_c = 1 + k_c \qquad (12)$$

and R_c is the retardation factor for the colloids.

To take into account sorption of the radionuclides to the matrix and to the colloids we assume that radionuclide in water, matrix, and on the colloids are in equilibrium.

From Page No.___

The equilibrium equations are:

$$\frac{\rho}{\theta s}c_{ns} = k_{ns}c_{nw}$$
$$\frac{\rho}{\theta s}c_{nsc} = k_{nsc}c_{nw} \qquad (13)$$
$$c_{nwc} = k_{nwc}c_{nw}$$

where k_{ns}, k_{nsc}, and k_{nwc} are the equilibrium coefficients for c_{ns}, c_{nsc}, and c_{nwc}, to be determined from laboratory tests.

The conservation equation for radionuclides in water including sorption and radioactive decay, with a decay coefficient λ is:

$$\frac{\partial(sc_{nw})}{\partial t} + \lambda sc_{nw} = -\frac{\partial(sv_{nw})}{\partial z} - \frac{\partial(svc_{nw})}{\partial z} - \left[\frac{\partial\left(\frac{\rho}{\theta}c_{ns}\right)}{\partial t} + \lambda\frac{\rho}{\theta}c_{ns}\right] - \left[\frac{\partial\left(\frac{\rho}{\theta}c_{nsc}\right)}{\partial t} + \lambda\frac{\rho}{\theta}c_{nsc}\right] - \left[\frac{\partial(sc_{nwc})}{\partial t} + \lambda sc_{nwc}\right] \qquad (14)$$

Using Eqs. (13), we get:

$$R_n\frac{\partial(sc_{nw})}{\partial t} = -\frac{\partial(sv_{nw})}{\partial z} - \frac{\partial(svc_{nw})}{\partial z} - \lambda R_n sc_{nw} \qquad (15)$$

where R_n is the retardation factor for the radionuclides, and is given by:

$$R_n = 1 + k_{ns} + k_{nsc} + k_{nwc} \qquad (16)$$

To Page No._____

Witnessed & Understood by me, | Date | Invented by / Recorded by | Date

From Page No.___

6/16/1999

The conservation equation for the radionuclides, Eq. (14) above, has something missing. It does not take into account advection of radionuclides attached to colloids in the water. The corrected Eq. (14) reads:

$$\frac{\partial(sc_{nw})}{\partial t} + \lambda sc_{nw} = -\frac{\partial(sv_{nw})}{\partial z} - \frac{\partial(svc_{nw})}{\partial z} - \frac{\partial(svc_{nwc})}{\partial z} - \left[\frac{\partial\left(\frac{\rho}{\theta}c_{ns}\right)}{\partial t} + \lambda\frac{\rho}{\theta}c_{ns}\right] - \left[\frac{\partial\left(\frac{\rho}{\theta}c_{nsc}\right)}{\partial t} + \lambda\frac{\rho}{\theta}c_{nsc}\right] - \left[\frac{\partial(sc_{nwc})}{\partial t} + \lambda sc_{nwc}\right] \qquad (17)$$

As a result, Eq. (15) needs also to be corrected. Its corrected version reads:

$$R_n\frac{\partial(sc_{nw})}{\partial t} = -\frac{\partial(sv_{nw})}{\partial z} - R_{nwc}\frac{\partial(svc_{nw})}{\partial z} - \lambda R_n sc_{nw} \qquad (18)$$

where:

$$R_{nwc} = 1 + k_{nwc} \qquad (19)$$

Description of the COLLOID code.

The code includes several sections. We headline each section with a comment. We now describe the code section by section.

Defining vectors along the column.

We define the vectors using the dimension command. Above that we explain what the vectors are and give their dimensions.

Common /com1/ transfers fixed input data to the functions PSIS and QAYS. These functions evaluate $\psi(s)$ and $k(s)$ respectively.

To Page No._____

Witnessed & Understood by me, | Date | Invented by / Recorded by | Date

From Page No.____

Input

We input fixed data using the data command. This is temporary. Later we intend to input fixed data using the namelist command.

Above the data commands we use comments to explain what the data are, and give their dimensions.

Initial conditions

We use a do loop along the column, and input or compute the initial values of the various variables.

Start time march

We start a time march from time=0 to time=tmax with equal increments dt. We use an explicit integration scheme with constant time steps rather than an implicit scheme although this may require smaller time steps. The system is nonlinear, and using an implicit scheme is very complicated.

All the rest of the code is within the time march do loop.

Determine upper boundary head

We provide several options for prescribing the outside head hout. We choose from them using the goto( ) command, according to the value of masb, which is an input quantity. We now have in the code: masb=1, for constant head, and masb=3, for constant head up to time=t1. We haven't yet implemented masb=2, for given precipitation rate.

Witnessed & Understood by me,	Date	Invented by	Date
		Recorded by	

From Page No.____

6/17/1999

Compute pressure head for saturated sections

We assume that saturated sections may form along the column. In such a case we compute the pressure head in a cell as the distance of the cell center from the upper boundary of the saturated section.

Compute flow velocities

We go along the column with a do loop. In each cell we compute hup and hdn. (We designate by 'up' quantities on the upper side of the cell boundary, and by 'dn' quantities on the lower side of the cell boundary. For unsaturated flow hup=-psiup(sup), and similarly for hdn. The psi(s) relation is evaluated by Eq. (2) in the function PSIS. The relation psi(s) can be easily changed in function PSIS. For saturated flow we use for hup and hdn the piesometric heads computed before.

At the upper boundary we use hup=hout. At the lower boundary we use hdn=0. this was corrected later on. see at p. 21 I.P. 5/15/2000

We compute the hydraulic conductivity at a cell boundary using the function QAYS given in Eq. (3). For s we use an effective value seff, given by:

$$s_{eff}(i) = \min(s(i-1), s(i)) \qquad (20)$$

Finally we compute the flow velocities by:

$$v(i) = -k\left[-1 + \frac{(h_{dn} - h_{up})}{dz}\right] \qquad (21)$$

Witnessed & Understood by me,	Date	Invented by	Date
		Recorded by	

From Page No.___

Compute radionuclides dispersion velocity

We do this in the same way we compute the flow velocities. We compute cnup and cndn, where cn is cnw, the radionuclide concentration in the water. We then compute the equivalent dispersion coefficient Deq by Eq. (6), and the dispersion velocities by:

$$v_{nw}(i) = -D_{eq}\frac{(cn)_{dn} - (cn)_{up}}{dz} \qquad (22)$$

At the upper boundary we use cnup=cnwm, which is the maximum concentration of radionuclides in water. At the lower boundary we use cndn=0.

6/18/1999

Computing new saturations

To compute the new saturations from water volume conservation, we first calculate the products vsup and vsdn. vsup for cell i is either v(i)*s(i-1), or v(i)*s(i), depending on flow direction. Similarly, vsdn is either v(i+1)*s(i) or v(i+1)*s(i+1). For the upper and lower boundaries (i=1, and i=nc+1) we use s=1 when the flow is into the column.

We then calculate the new saturations from the volume conservation equation in finite differences as follows:

$$s_{new}(i) = s(i) + \frac{dt}{dz}(vs_{up} - vs_{dn}) \qquad (23)$$

To Page No._____

Witnessed & Understood by me,	Date	Invented by	Date
		Recorded by	

From Page No.___

Computing new radionuclides concentrations

Similar to the quantities vsup and vsdn above, we first compute here the quantities vscup, vscdn, vnwsup, and vnwsdn. vscup is either v(i)*s(i-1)*cnw(i-1), or v(i)*s(i-1)*cnw(i-1), depending on the flow direction, and similarly with vscdn. Vnwsup is either vnw(i)*s(i-1) or vnw(i)*s(i), and similarly for vnwsdn, depending on the dispersion direction. At the upper and lower boundaries we use cnw(i-1)=cnwm, and cnw(nc+1)=0, when the flow is into the column.

We then use conservation of radionuclides mass in finite differences to compute the new radionuclides concentrations.

$$cnw_{new}(i) = \frac{1}{s_{new}(i)}cnw(i)s(i) + \frac{1}{s_{new}(i)}\frac{dt}{dz}\frac{1}{R_n}\left[R_{nwc}(vsc_{up} - vsc_{dn}) + (vnws_{up} - vnws_{dn})\right] - \frac{dt}{s_{new}}\frac{dv}{dz}\lambda s(i)cnw(i). \qquad (24)$$

R_n and R_{nwc} enter into this equation as explained above with respect to Eqs. (18), and (19).

6/21/1999

Computing new colloid concentrations

Similar to radionuclides above, we compute vscup and vscdn at the upper and lower cell boundaries by. vscup is either v(i)*s(i-1)*ccw(i-1) or v(i)*s(i)*ccw(i), and vscdn is either v(i+1)*s(i)*ccw(i) or v(i+1)*s(i+1)*ccw(i+1), depending on the flow direction. At the upper boundary we use s=1 and ccw(i-1)=ccwout, and at the lower boundary we use s=1 and ccw(i+1)=ccwaq, when the flow direction is into the column.

To Page No._____

Witnessed & Understood by me,	Date	Invented by	Date
		Recorded by	

From Page No.___

We compute the new colloid concentrations using the conservation equation for the colloids in water. In finite differences it reads:

$$ccw_{new}(i) = \frac{1}{s_{new}(i)}\left[ccw(i)s(i) + \frac{1}{R_c}\frac{dt}{dz}\left(vsc_{up} - vsc_{dn}\right)\right] \qquad (25)$$

R_c is the retardation coefficient for the colloids, as defined in Eq. (12).

Avoiding numerical dispersion for saturation, radionuclides concentrations, And colloid concentrations.

Whenever the absolute change in saturation in a cell is less than eps_s, we transfer that change to the uppermost cell.

We the correct the new concentrations in the uppermost cell accordingly.

Whenever the absolute change in Radionuclides or colloids concentrations in a cell is less than eps_n, or eps_c, respectively, we transfer the changes to the uppermost cell.

Adjusting for s>1 and s<s_{min}.

Whenever $s_{new}(i)>s_{sat}$, and $s_{new}(i+1)<1$, we set $s_{new}(i)$ to s_{sat}, and transfer the difference to the cell below. S_{sat} is a value slightly lower than 1. We use s_{sat}=0.999.

Whenever $s_{new}(i)>1$, $s_{new}(i+1)=1$, we set $s_{new}(i)$ to 1, and transfer the difference to the cell below.

In this way we ensure that the column can become saturated only from the aquifer upwards, and not from the upper boundary downwards.

Whenever $s_{new}(i)>1$, $s_{new}(i+1)=1$, we set $s_{new}(i)$ to 1, and transfer the difference to the cell below.

To Page No._____

Witnessed & Understood by me,	Date	Invented by	Date
		Recorded by	

From Page No.___

In this way we ensure that the column can become saturated only from the aquifer upwards, and not from the upper boundary downwards.

Whenever $s_{new}(i)<s_{min}$, we set $s_{new}(i)$ to s_{min}, and transfer the difference to the cell above.

In both cases, we correct the radionuclides concentration and the colloid concentration in the receiving cells accordingly.

6/22/1999

Updating s, cnw, and ccw along the column

After all the corrections and adjustments, we put the new values $s_{new}(i)$, $cnw_{new}(i)$ and $ccw_{new}(i)$, into the old values s(i), cnw(i) and ccw(i).

Computing accumulated mass of radionuclides and of colloids outflowing through a chosen cell boundary

We designate the chosen cell boundary as i=iac. The outflowing colloid mass during the time dt is given by: θv(i)s(i-1)ccw(i-1)dt or θv(i)s(i)ccw(i)dt, depending on the flow direction. The outflowing radionuclides mass during the time dt is the sum of convection flow and dispersion flow. Convection flow is given by: θv(i)s(i-1)cnw(i-1)dt or θv(i)s(i)cnw(i)dt. Dispersion flow is given by: θvnw(i)s(i-1)dt or θvnw(i)s(i)dt, depending on the dispersion flow direction.

We accumulate the outflowing colloid mass in sccw, and the outflowing radionuclides mass in scnw.

We're now ready to output the results of this time cycle and to start a new cycle.

To Page No._____

Witnessed & Understood by me,	Date	Invented by	Date
		Recorded by	

From Page No.____

6/23/1999

All the information below has been taken from: NRC Iterative Performance Assessment Phase 2.

Table-1:

	Grain density (Kg/m**3)	Porosity	Residual saturation	Saturated Conductivity (mm/yr)	Alpha (1/m)	Beta
Topopa Spring w.	2580.	.06 -.16	.080	.11 – 36.	.006	1.4 – 2.2
Calico Hills n.w.v	2370.	.33 -.56	.041	1.2E2 – 6.1E3	.016	1.5 – 4.9
Calico Hills n.w.z	2230.	.20 -.33	.110	.004 - .20	.003	1.2 – 3.3
Prow Pass, w.	2590.	.24 -.40	.066	58. – 300.	.014	2.0 – 3.4
U&M Crater Flat, non-w.	2270.	.18 - .30	.135-.322 U	1.6 – 4.6 Up. 1.3 – 4.8 Md.	.004	1.5 – 2.4
Bullfrog, Welded.	2630.	.19-.32	.056 - .061	110. - 140.	.02	2.3 – 4.2

The Grain Density, Porosity, Saturated Conductivity, Alpha and Beta are copied from Table B-6, page B-24. The Residual Saturation are taken from Table B-1, pp. B-2 – B-9.

Table – 2: K_d (m**3/kg)

Element	Topopa Spring	Calico Hills n.w.v	Calico Hills n.w.z	Prow Pass	Up. & Md. Crater Flat	Bullfrog
Am	.810	.810	1.7	4.5	1.36	.140
Pu	.170	.170	.066	.130	.053	.094
U	.0002	.020	.001	0.0	.0008	.002
Se	.0026	.003	.0045	.0025	.0036	.013
Tc	1.3e-5	0.0	0.0	1.7E-4	-----	.0042
Np	.0045	.0045	.0027	.0051	.0022	.0051
Sr	.08	.034	8.9	.450	7.1	.280
Cs	.36	.24	22.0	2.2	17.6	3.2
Ba	1.2	.6	61.0	3.9	-------	1.1
Ra	1.5	1.5	1.5	1.5	1.2	5.0

This Table has been copied from page B-16, and table B-7 page B-26

To Page No.____

Witnessed & Understood by me, | Date | Invented by | Date | Recorded by

From Page No.____

The Genuchten equations are page 4-27 and page 4-28:

$$s = (s_s - s_r)\left[\frac{1}{1+|\alpha h|^{\beta}}\right]^{(1-\frac{1}{\beta})}, \quad (1)$$

and

$$k_r = \left[1-\left(1-(s_e)^{\frac{1}{\lambda}}\right)^{\lambda}\right]^2\sqrt{s_e} \quad (2)$$

Where: s = saturation.
s_s = saturation at full saturation.
s_r = residual saturation.
h = pressure or suction.
α, β = fitting parameters.
k_r = relative conductivity.
λ = van Genuchten fitting parameter (1-1/β)
s_e = effective saturation given by:

$$s_e = \frac{s - s_r}{s_s - s_r} \quad (3)$$

The retardation factor R_f is calculate from the distribution coefficient (k_d) (page 4-30) by:

$$R_f = 1. + \frac{\rho(1-n)}{\theta}k_d \quad (4$$

Where: θ = moisture content.
ρ = grain density.
n = porosity.

In page 4-53 we find "The system code uses of the present NCR performance assessment uses the k_d approach in estimating retardation of radionuclides. The relationship of k_d to retardation is:

To Page No.____

Witnessed & Understood by me, | Date | Invented by | Date | Recorded by

From Page No.___

$$R_f = 1 + \rho \frac{k_d}{\theta} \qquad (5)$$

From what is written in Appendix B, comes that after all eq.(4) is the one that was used.

7/13/1999

We wanted to use Van-Genutchen functions for $\psi(s)$ and $k(s)$. We first evaluated those functions for some values of the parameters α and β, with the help of a code built for VG equations and given below.

```
! Code for calculating the saturation (s),and the hydraulic
!  conductivity from the VG equatins.

! s= saturation.
! ss= saturation at at full saturation.
! sr= residual saturation.
! h= pressure or suction (m).
! alfa= fitting parameter (1/m).
! beta= fitting parameter.
! kr= relative conductivity.
! ksat= saturated conductivity (m/yr).
! k= conductivitty at given saturation (m/yr).

      data ss,sr/.999,.08/
      data alfa,beta/.006,1.8/
      data hmax/1000./
      data qsat/1.0e-3/

      elamda=1.-1./beta
      dh=hmax/20.

      do h=hmax,dh,-dh
      s=(ss-sr)*(1./(1.+(abs(alfa*h))**beta))**elamda
!     print *,h,s
      seff=(s-sr)/(ss-sr)
      qr=(1.-(1.-seff**(1./elamda))**elamda)**2*sqrt(seff)
      q=qr*qsat
      if(s.le.sr) goto 999
      write (1,*) h,s,seff,qr,q
      enddo

  999 end
```

To Page No._____

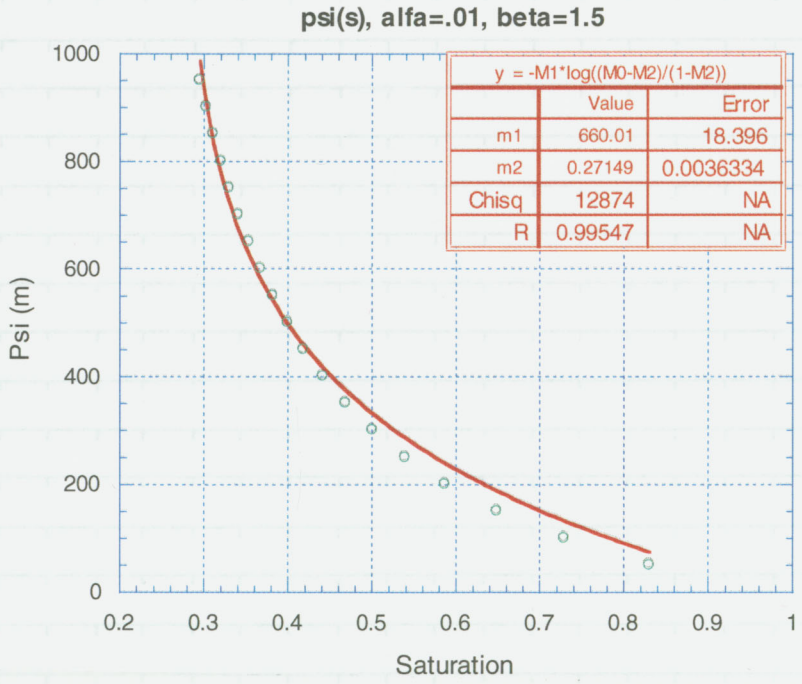
Witnessed & Understood by me,	Date	Invented by	Date
		Recorded by	

We saw that $\psi(s)$ is not defined for values of s close to 1. Because our code is intended for both saturated and unsaturated flow, we need $\psi(s)$ to be defined for all s. We therefore plotted $\psi(s)$ and k(s), and curvefitted them with simple functions. The functions are:

$$\psi = -\psi_0 \ell n \frac{s - s_1}{1 - s_1} \qquad (26)$$

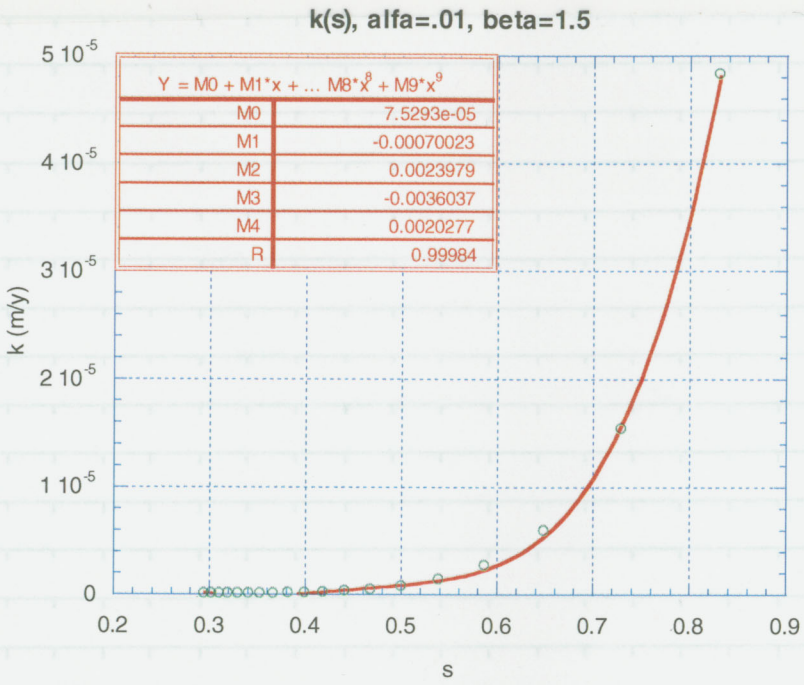
$$k = a_0 + a_1 s + a_2 s^2 + a_3 s^3 + a_4 s^4$$

For α=0.01/m, β=1.5 the, s_{min}=0.08, s_{sat}=0.999, k_{sat}=.001 m/y the Van-Genutchen and the curve fits are given below.



Witnessed & Understood by me, Date | Invented by | Recorded by | Date

From Page No.



We intend to use equations (26) for ψ(s) and k(s) in the code as maspsik=3.

We introduced the reading in of input data through the NAMELIST command. This is convenient because the NAMELIST file is written in free format in the form: a=####, b=####, etc. The input file is fort.11. On file fort.12 we print out all the data input from fort.11.

To be able to identify which input data was used to obtain certain output data, we introduce a sequential run number denoted by nrun. At the beginning of a run we read the previous nrun from fort.13 and rewind fort.13. We then write nrun on all output files. At the end of the run we increase nrun by 1, and write the new nrun on fort.13.

To Page No.

Witnessed & Understood by me, | Date | Invented by | Date | Recorded by

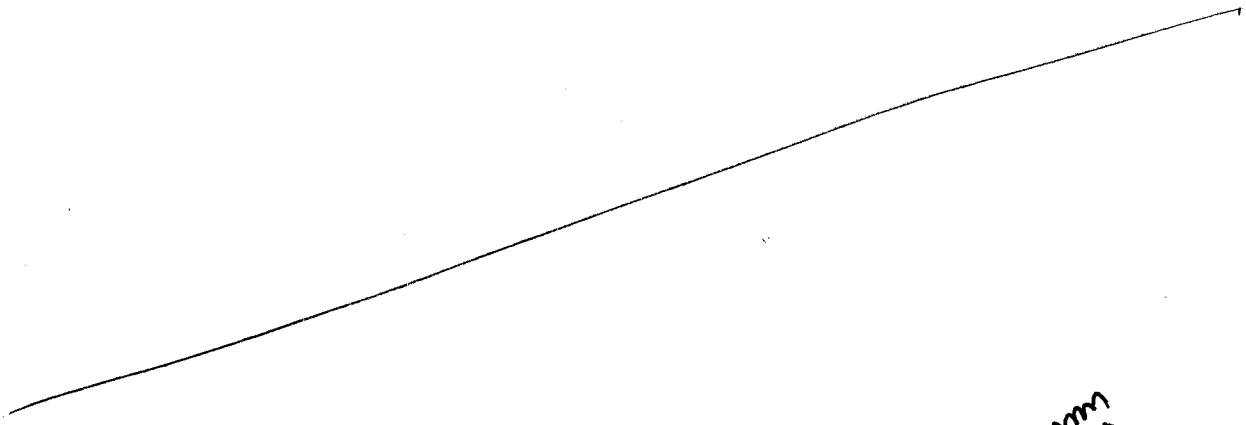
7/29/1999

To observe the effect of colloids on the transport of radionuclides, we introduced a mass accumulation scheme. We input the parameter i_{ac}, which is the number of the cell boundary through which we accumulate the mass of radionuclides and the mass of colloids. Radionuclides going through a cell boundary come from three sources: convected with the water, convected with colloids in the water, and dispersed relative to the flow. At each time step we calculate the rates of these three components, multiply by dt and add to the accumulated mass obtained at the previous time. We do the same for the colloids in the water. We printout the accumulated mass values on file 2.

When the flow reaches the lower boundary (aquifer level), the column becomes saturated from the bottom up. This means that the phreatic level (interface between the saturated and the unsaturated zones) rises. But in a real situation the phreatic level would not rise so fast because of radial flow in the aquifer (which are not included in our model). To account for the radial effect we adjusted the lower boundary condition as follows. When the phreatic level rises by Δh, we apply a velocity boundary condition, and the velocity is given by:

$$v(nc+1)=\alpha_{aq}\Delta h \qquad (27)$$

where α_{aq} is an input coefficient. The effect of this boundary condition is to slow down the rise of the phreatic level.



Project No._____
Book No._____ TITLE_____

From Page No.____

We introduced a velocity boundary condition for the flow at the upper boundary. We prepared space for it before, and we called it masb=2 (see above). According to this boundary condition the inflow velocity is v_{in} during $0<t<t_2$ at the beginning of each year, and $v_{in}=0$ for the rest of the year. The input parameters for this boundary condition are v_{in} and t_2.

Initially we didn't limit the amount of radionuclides that could be adsorped to the colloid particles in the water, assuming that the concentration of colloids was always large enough. But to be able to reduce the colloid concentration down to zero, we introduced a saturation limit for the adsorped radionuclides. We added an input parameter denoted by α_n. The concentration limit is given by:

$$(c_{nwc})_m = \alpha_n c_{cw} \qquad (28)$$

We see that the saturation limit is different for each cell. When c_{nwc} reaches the saturation limit there is no more in equilibrium with the radionuclides in the water, and the conservation equation for the radionuclides has to be changed. We change it as follows:

- We go back to Eq. (17).
- Wherever old values are used in the finite difference equivalent of Eq. (17), we use the known values of c_{nwc}.
- For the new values of c_{nwc} we use a two step procedure.
- In step one we assume equilibrium.
- In step two we check if the saturation limit has been passed. If it has, we set c_{nwc} to $\alpha_n c_{cw}$ and adjust c_{nw} accordingly.

To Page No._____

Witnessed & Understood by me,	Date	Invented by	Date
		Recorded by	


```
!   COLLOID
!   Movement of water, radionuclids and colloids iv a vertical unsaturated column.
!   1D time dependent code.

!   z= vertical location measured downward (m).
!   s= degree of saturation.
!   snew= degree of saturation at end of time step.
!   v= fluid velocity (m/y).
!   hp= pressure hear (m).
!   cnw= mass of radionuclides in water per unit volume of fluid (mg/m**3).
!   cnwnew= mass of radionuclides in water per unit volume of fluid at
!           end of time step (mg/m**3).
!   ccwnew= mass of colloids in water per unit volume of fluid at
!           end of time step (mg/m**3).
!   cns= mass of radionuclides sorbed in non colloidal matrix per unit matrix mass (mg/kg).
!   cnsc= mass of radionuclides sorbed on colloids in matrix per unit matrix mass (mg/kg).
!   cnwc= mass of radionuclides sorbed on colloids in water per unit volume of fluid (mg/m**3).
!   ccs= mass of colloids in matrix per unit matrix mass (mg/kg).
!   ccw= mass of colloids in water per unit volume of fluid (mg/m**3).
!   vnw= dispersion velocity of radionuclides in water (mg/m**2/y).
!   cell bounbary i is above cell i.
!   s,hp,c... defined in cell.
!   z,v,vnw defined at cell boundary.
        common/com1/ssat,smin
        common/com12/maspsik
        common/com2/alfa,beta,gamma,qaysat
        common/psi20/psi0,s1
        dimension z(500),s(500),snew(500),v(500),hp(500)
        dimension cnw(500),cns(500),ccs(500),ccw(500),cnwc(500),cnsc(500)
        dimension cnwnew(500),ccwnew(500)
        dimension vnw(500)

!   INPUT

!   teta= porosity of matrix (volume of pores per unit volume).
!   ro= density of the matrix (kg/m**3).
!   hc= column height (m).
!   dz= cell size (m).
!   dt= time step (y).
!   tmax= maximum runningtime (y).
!   cnwaq= mass of radionuclides in water per unit volume of aquifer water (mg/m**3).
!   cnwm= maximum mass of radionuclides in water per unit volume of fluid (mg/m**3).
!   ccwout= concentration of colloids in incoming water (mg/m**3).
!   ccwaq= concentration of colloids in aquifer water.
!   sint= initial saturation.
!   ccwint= initial concentration of colloids in water (mg/m**3).
!   cnwint= initial concentration of nuclides in water (mg/m**3).
!   masb= flow boundary contition type. masb=1: constant head. masb=2: given precipitation rate.
!         masb=3: constant head up to t=t1.
!   hub= upper boundary given constsnt head in masb=1,3 (m).
!   t1= duration of constant head in masb=3.
!   t21= biginning of irrigation period in masb=2.
!   t22= end of irrigation period in masb=2.
!   t23=biginning of anew irrigation period in masb=2.
!   tpr=initial output time (y).
!   dtpr= printing time step (y).
!   dpor= average tetae diameter of porous medium (m).
!   ddn= molecular diffusion coefficient for the radionuclides (m**2/y).
!   ibt= number of cell for breakthrough curve.
!   apec,bpec= parametersin effective dispersionversus Peclet number function.
!   rc=retardation function for colloids.
!   rn=retardation function for nuclides.
!   qayd=absorption coefficient (m**3/kg)
!   elamda=radioective decay coefficient (1/y).
!   virrig=annual irrigation (m/y)
!   consaq= A constant for calculating v(nc+1) when the phriatic surface rises.
!           The equation is: v(nc+1)=consaq*dh where dh is the rise of the
!           phriatic surface level (1/y).
!   alfan=a non dimensional coeff. determaning the capacity of the colloid to
!           absorb nuclides, used in function: cnwc=alfan*ccw(i).

        namelist/in/nrun
        namelist/in/teta,ro,cnwm,ccwout,qaysat
        namelist/in/sint,ccwint,cnwint,masb,maspsik,hub,t1
        namelist/in/t21,t22,t23,virrig,consaq
        namelist/in/epss,epsnw,epscw,ssat,smin,ccwaq,cnwaq
        namelist/in/dpor,apec,bpec,ddn,rc,elamda,ibt,iac
        namelist/in/alfa,beta,psi0,s1,qayd,ckdns,ckdnsc,ckdnwc,alfan
        namelist/in/hc,dz,dt,tmax,tpr,dtpr
```

Witnessed & Understood by me,	Date

From Page N

```
      data nrun/1/
      data teta,ro/.3,2000./
      data hc,dz/300.,10./
      data dt,tmax/1.,50./
      data tpr,dtpr/0.,1./
      data hub,t1/.01,10000./
      data t21,t22,t23,virrig/0.,.5,1.,.76/
      data cnwm,ccwout/2.4,1.e4/
      data sint,ccwint,cnwint/.7,5.e3,0./
      data masb,maspsik/3,2/
      data dpor,ddn/1.e-4,.038/
      data apec,bpec/.63,1./
      data epss,epsnw,epscw/1.e-6,1.e-5,1.e-6/
      data smin,ssat/.25,.99/
      data ccwaq,cnwaq,consaq/ .1,0.,.25/
      data rc,qayd,elamda/1.0,4.e3,0./
      data ibt/5/
      data iac/5/
      data alfa,beta,qaysat/.016,1.3,10./
      data psi0,s1/66.,.4/
      data ckdns,ckdnsc,ckdnwc,alfan/0.,0.,1.,2.4e-4/


!     write(11,*)
!     write(13,*)
      read(11,in)
      read(13,*)nrun
      rewind 13
      write(12,in)


!  nc= number of cells in column.
!  INITIAL CONDITIONS
      deldt=dt/100.
      sccw=0.
      scnw=0.
      scnwc=0.
      qayc=rc-1.
      qayn=qayd*teta/ro
      qayns=ckdns*qayn
      qaynsc=ckdnsc*qayn
      qaynwc=ckdnwc*qayn
!     rn=1.+qayn
          rn=1.+qaynwc+qayns+qaynsc
      gamma=1.-1./beta
      nc=hc/dz
          cnwcout=min(qaynwc*cnwm,alfan*ccwout)
          cnwcaq=min(qaynwc*cnwaq,alfan*ccwout)
      do i=1,nc
      s(i)=sint
      z(i)=(i-1)*dz
      cnw(i)=cnwint
      ccw(i)=ccwint
      cns(i)=teta/ro*qayns*cnw(i)*s(i)
      cnsc(i)=teta/ro*qaynsc*cnw(i)*s(i)
      cnwc(i)=qaynwc*cnw(i)
      ccs(i)=teta/ro*qayc*ccw(i)*s(i)
      enddo
      z(nc+1)=hc
      s(nc+1)=1.

! header output
      write (1,111),nrun
      write (3,131),nrun,ccwint,rc
      write (2,121),nrun,ibt,iac
      write (2,122) rn,qayns,qaynsc,qaynwc,rc
      write (2,123)

!   START TIME MARCH

      do time=dt,tmax,dt

!   determine upper boundary head.
      goto (10,20,30) masb
!   masb=1, constant head.
10    continue
      hout=hub
      vin=0.
      t1=tmax+dt
      goto 11
```

Witnessed &

To Page No.

Date

TITLE

From Page No.

```
!   masb=2, given precipitation rate.
20    continue
      hout=0.
      if(time.ge.t21.and.time.lt.t22) vin=virrig
      if(time.ge.t22.and.time.lt.t23) vin=0.
      if(time.gt.t23) then
      t21=t21+1.
      t22=t22+1.
      t23=t23+1.
      vin=virrig
      endif
      goto 11

!   masb=3, hout=hub until t=t1, then hout=0.
30    continue
      if(time.lt.t1) then
      hout=hub
      else
      hout=0.
      vin=0.
      endif
      goto 11
11    continue

!   compute pressure head for saturated sections.
      zsat=0.
      isat=1
!     flag=0.
      do i=1,nc+1
!     if(flag.eq.0..and.s(i).eq.1.) then
!     flag=1.
!     hsat=z(i)
!     endif
      if(s(i).lt.1.) then
      hp(i)=0.
      zsat=z(i+1)
      isat=i+1
      else
      hp(i)=z(i)+.5*dz-zsat
      endif
      enddo

!   compute flow velocities
      do i=1,nc+1
      if(i.eq.1) then
      sup=1.
      hup=hout
      else
      sup=s(i-1)
      psiup=psis(sup)
      if(sup.lt.1.) then
      hup=-psiup
      else
      hup=hp(i-1)
      endif
      endif
      if(i.eq.nc+1) then
      sdn=1.
      hdn=hp(nc+1)
      else
      sdn=s(i)
      psidn=psis(sdn)
      if(sdn.lt.1.) then
      hdn=-psidn
      else
      hdn=hp(i)
      endif
      endif
      seff=min(sup,sdn)
      qay=qays(seff)
      if(i.eq.1.and.hout.eq.0.) then
      v(1)=vin
      else
      v(i)=-qay*(-1.+(hdn-hup)/dz)
      endif
      enddo

      vout=consaq*(hc-zsat)
      vout=min(vout,qaysat)
```

Witnessed & Underst

Recorded by

To Page No.

Date

From Page No.____

```
      do i=isat,nc+1
      v(i)=vout
      enddo

!   compute radionuclides dispersion velocities.
      do i=1,nc+1
      if(i.eq.1) then
      cnup=cnwm
      else
      cnup=cnw(i-1)
      endif
      if(i.eq.nc+1) then
      cndn=cnwaq
      else
      cndn=cnw(i)
      endif
      pecle=abs(v(i))*dpor/ddn
      deq=ddn*apec
      deqn=ddn*apec*(1.+pecle/bpec)
      vnw(i)=-deq*(cndn-cnup)/dz
      enddo

!   computing new saturations.
      do i=1,nc
      if(i.eq.1) then
      if(v(1).ge.0.) then
      vsup=v(1)*1.
      else
      vsup=v(1)*s(1)
      endif
      if(v(2).ge.0.) then
      vsdn=v(2)*s(1)
      else
      vsdn=v(2)*s(2)
      endif
      endif

      if(i.gt.1.and.i.lt.nc) then
      if(v(i).ge.0.) then
      vsup=v(i)*s(i-1)
      else
      vsup=v(i)*s(i)
      endif
      if(v(i+1).ge.0.) then
      vsdn=v(i+1)*s(i)
      else
      vsdn=v(i+1)*s(i+1)
      endif
      endif

      if(i.eq.nc) then
      if(v(i).ge.0.) then
      vsup=v(i)*s(i-1)
      else
      vsup=v(i)*s(i)
      endif
      if(v(i+1).ge.0.) then
      vsdn=v(i+1)*s(i)
      else
      vsdn=v(i+1)*1.
      endif
      endif
      snew(i)=s(i)+dt/dz*(vsup-vsdn)
      enddo

!   computing new radionuclides concentrations.
      do i=1,nc
      if(i.eq.1) then
      if(v(1).ge.0.) then
      vscup=v(1)*1.*cnwm
          vsccup=v(1)*1.*cnwcout
      else
      vscup=v(1)*s(1)*cnw(1)
          vsccup=v(1)*s(1)*cnwc(1)
      endif
      if(v(2).ge.0.) then
      vscdn=v(2)*s(1)*cnw(1)
          vsccdn=v(2)*s(1)*cnwc(1)
      else
      vscdn=v(2)*s(2)*cnw(2)
```

To Page No.____


```
          vsccdn=v(2)*s(2)*cnwc(2)
      endif
      if(vnw(1).gt.0.) then
      vnwsup=vnw(1)*1.
      else
      vnwsup=vnw(1)*s(1)
      endif
      if(vnw(2).ge.0.) then
      vnwsdn=vnw(2)*s(1)
      else
      vnwsdn=vnw(2)*s(2)
      endif
      endif

      if(i.gt.1.and.i.lt.nc) then
      if(v(i).ge.0.) then
      vscup=v(i)*s(i-1)*cnw(i-1)
          vsccup=v(i)*s(i-1)*cnwc(i-1)
      else
      vscup=v(i)*s(i)*cnw(i)
          vsccup=v(i)*s(i)*cnwc(i)
      endif
      if(v(i+1).ge.0.) then
      vscdn=v(i+1)*s(i)*cnw(i)
          vsccdn=v(i+1)*s(i)*cnwc(i)
      else
      vscdn=v(i+1)*1.*cnwaq
          vsccdn=v(i+1)*1.*cnwcaq
      endif
      if(vnw(i).ge.0.) then
      vnwsup=vnw(i)*s(i-1)
      else
      vnwsup=vnw(i)*s(i)
      endif
      if(vnw(i+1).ge.0.) then
      vnwsdn=vnw(i+1)*s(i)
      else
      vnwsdn=vnw(i+1)*s(i+1)
      endif
      endif

      if(i.eq.nc) then
      if(v(i).ge.0.) then
      vscup=v(i)*s(i-1)*cnw(i-1)
          vsccup=v(i)*s(i-1)*cnwc(i-1)
      else
      vscup=v(i)*s(i)*cnw(i)
          vsccup=v(i)*s(i-1)*cnwc(i-1)
      endif
      if(v(i+1).ge.0.) then
      vscdn=v(i+1)*s(i)*cnw(i)
          vsccdn=v(i+1)*s(i)*cnwc(i)
      else
      vscdn=v(i+1)*1.*cnwaq
          vsccdn=v(i+1)*1.*cnwcaq
      endif
      if(vnw(i).ge.0.) then
      vnwsup=vnw(i)*s(i-1)
      else
      vnwsup=vnw(i)*s(i)
      endif
      if(vnw(i+1).ge.0.) then
      vnwsdn=vnw(i+1)*s(i)
      else
      vnwsdn=vnw(i+1)*1.
      endif
      endif
      cnwnew(i)=((cnw(i)*(1.+qayns+qaynsc)+cnwc(i))*s(i)/rn+dt/dz*(vscup-vscdn +vsccup-vsccdn  &
&         +vnwsup-vnwsdn)/rn-elamda*s(i)*dt/rn*(cnw(i)*(1.+qayns+qaynsc)+cnwc(i)))/snew(i)
      enddo

!   computing new colloids concentrations.
      do i=1,nc
      if(i.eq.1) then
      if(v(1).ge.0.) then
      vscup=v(1)*1.*ccwout
      else
      vscup=v(1)*s(1)*ccw(1)
      endif
      if(v(2).ge.0.) then
```

From Page No.

```
            vscdn=v(2)*s(1)*ccw(1)
            else
            vscdn=v(2)*s(2)*ccw(2)
            endif
            endif

            if(i.gt.1.and.i.lt.nc) then
            if(v(i).ge.0.) then
            vscup=v(i)*s(i-1)*ccw(i-1)
            else
            vscup=v(i)*s(i)*ccw(i)
            endif
            if(v(i+1).ge.0.) then
            vscdn=v(i+1)*s(i)*ccw(i)
            else
            vscdn=v(i+1)*s(i+1)*ccw(i+1)
            endif
            endif

            if(i.eq.nc) then
            if(v(i).ge.0.) then
            vscup=v(i)*s(i-1)*ccw(i-1)
            else
            vscup=v(i)*s(i)*ccw(i)
            endif
            if(v(i+1).ge.0.) then
            vscdn=v(i+1)*s(i)*ccw(i)
            else
            vscdn=v(i+1)*1.*ccw(i)*ccwaq
            endif
            endif
            ccwnew(i)=(ccw(i)*s(i)+dt/dz*(vscup-vscdn)/rc)/snew(i)
            enddo

!   avoiding numerical dispersion in s.
            do i=2,nc
            if(abs(snew(i)-s(i)).lt.epss) then
            dels=snew(i)-s(i)
            ccwnew(1)=(ccwnew(1)*snew(1)+dels*ccwnew(i))/(snew(1)+dels)
            cnwnew(1)=(cnwnew(1)*snew(1)+dels*cnwnew(i))/(snew(1)+dels)
            snew(i)=s(i)
            snew(1)=snew(1)+dels
            endif
            enddo

!   adjusting for s(i)>1.
            do i=1,nc-1
            if(snew(i).gt.ssat.and.s(i+1).lt.1.) then
            dels=snew(i)-ssat
            ccwnew(i+1)=(ccwnew(i+1)*snew(i+1)+dels*ccwnew(i))/(snew(i+1)+dels)
            cnwnew(i+1)=(cnwnew(i+1)*snew(i+1)+dels*cnwnew(i))/(snew(i+1)+dels)
            snew(i+1)=snew(i+1)+dels
            snew(i)=ssat
            endif
            if(snew(i).gt.1..and.s(i+1).eq.1.) then
            dels=snew(i)-1.
            ccwnew(i+1)=(ccwnew(i+1)*snew(i+1)+dels*ccwnew(i))/(snew(i+1)+dels)
            cnwnew(i+1)=(cnwnew(i+1)*snew(i+1)+dels*cnwnew(i))/(snew(i+1)+dels)
            snew(i+1)=snew(i+1)+dels
            snew(i)=1.
            endif
            enddo
            if(snew(nc).gt.1.) snew(nc)=1.

!   adjusting for s(i)<smin.
            do i=nc,2,-1
            if(snew(i).lt.smin) then
            dels=snew(i)-smin
            ccwnew(i-1)=(ccwnew(i-1)*snew(i-1)+dels*ccwnew(i))/(snew(i-1)+dels)
            cnwnew(i-1)=(cnwnew(i-1)*snew(i-1)+dels*cnwnew(i))/(snew(i-1)+dels)
            snew(i-1)=snew(i-1)+dels
            snew(i)=smin
            endif
            enddo
            if(snew(1).lt.smin) snew(1)=smin

!   avoiding numerical dispersion for nuclides.
            do i=2,nc
            if(abs(cnwnew(i)-cnw(i)).lt.epsnw) then
            delnw=cnwnew(i)-cnw(i)
```

Witnessed & Under

TITLE

From Page I

```
            cnwnew(i)=cnw(i)
            cnwnew(1)=cnwnew(1)+delnw*snew(i)/snew(1)
            endif
            enddo
            if(abs(cnwnew(1)-cnw(1)).lt.epsnw) cnwnew(1)=cnw(1)

adjusting for cnw>cnwm.
            do i=1,nc
            if(cnwnew(i).gt.cnwm) then
            cnwnew(i)=cnwm
            endif
            enddo

avoiding numerical dispersion for colloids.
            do i=2,nc
            if(abs(ccwnew(i)-ccw(i)).lt.epscw) then
            delcw=ccwnew(i)-ccw(i)
            ccwnew(i)=ccw(i)
            ccwnew(1)=ccwnew(1)+delcw*snew(i)/snew(1)
            endif
            enddo
            if(abs(ccwnew(1)-ccw(1)).lt.epscw) ccwnew(1)=ccw(1)

computing mass of colloids and nuclides accumulating at i=iac
i=iac
if (v(i).gt.0.) then
sccw=sccw+teta*s(i-1)*ccw(i-1)*v(i)*dt
scnwc=scnwc+teta*s(i-1)*cnwc(i-1)*v(i)*dt
            if (vnw(i).gt.0.) then
            scnd=teta*vnw(i)*s(i-1)
            else
            scnd=teta*vnw(i)*s(i)
            endif
scnw=scnw+(teta*s(i-1)*cnw(i-1)*v(i)+ scnd)*dt
sntot=scnwc+scnw
else
sccw=sccw+teta*s(i)*ccw(i)*v(i)*dt
scnwc=scnwc+teta*s(i)*cnwc(i)*v(i)*dt
            if (vnw(i).gt.0.) then
            scnd=teta*vnw(i)*s(i-1)
            else
            scnd=teta*vnw(i)*s(i)
            endif
scnw=scnw+(teta*s(i)*cnw(i)*v(i)+scnd)*dt
sntot=scnwc+scnw
endif


!   update saturations.
            do i=1,nc
            s(i)=snew(i)
            enddo

!   update nuclides concentrations in water (cnw).
            do i=1,nc
            cnw(i)=cnwnew(i)
            enddo

!   update colloids concentrations in water (cnw).
            do i=1,nc
            ccw(i)=ccwnew(i)
            enddo

!   computing of radionuclides mass sorbed on non colloidal matrix, on
!   colloids in matrix, on colloids in water, and of mass of colloids in matrix,
!   and correcting for saturation of amount of nuclides sorped to colloids in the water.
            do i=1,nc
            cns(i)=teta/ro*qayns*cnw(i)*s(i)
            cnsc(i)=teta/ro*qaynsc*cnw(i)*s(i)
            cnwc(i)=qaynwc*cnw(i)
            ccs(i)=teta/ro*qayc*ccw(i)*s(i)

            cnwcm=alfan*ccw(i)
            if(cnwc(i).gt.cnwcm) then
            delcnwc=cnwc(i)-cnwcm
            cnwc(i)=cnwcm
            delcnw=delcnwc/(1.+qayns+qaynsc)
            cnw(i)=cnw(i)+delcnw
            if(cnw(i).gt.cnwm) cnw(i)=cnwm
            cns(i)=cns(i)+teta/ro*s(i)*qayns*delcnw
```

Witnesse

Recorded by

From

```
        cnsc(i)=cnsc(i)+teta/ro*s(i)*qaynsc*delcnw
        endif

        enddo


!    output.

        if(time.ge.(tpr-deldt))then
        write (1,113)tpr,zsat
        write (3,133)tpr,zsat
        write (1,112)
        write (3,132)
        tpr=tpr+dtpr
        do i=1,nc+1
        write(1,114) z(i),s(i),v(i),vnw(i),cnw(i)
        write(3,134) z(i),s(i),v(i),ccw(i)
        enddo
        write(2,124)time,cnw(ibt),scnw,scnwc,sntot,sccw
        endif
!   END TIME MARCH
        enddo

        nrun=nrun+1
        write(13,*) nrun

!   headlines for output
111     format(5x,'no. of run=',i3)
112     format(3x,'z(i)',8x,'s(i)',11x,'v(i)',11x,'vnw(i)',10x,'cnw(i)')
113     format(/5x,'time=',f7.3,5x,'z(sat)='f7.2/)
114     format(f8.3,4e15.5)
121     format(5x,'no. of run=',i3,10x,'ibt= ',i3,5x,'iac= ',i3)
122     format(5x,'rn=',f7.3,5x,'qayns=',f6.2,5x,'qaynsc=',f6.2,5x,'qaynwc=',f6.2,5x,'rc=',f6.2/)
123     format(3x,'time',10x,'cnw',11x,'scnw',10x,'scnwc',11x,'sntot',11x,'sccw')
124     format(f8.2,5e15.5)
131     format(5x,'no. of run=',i3,5x,'ccwint=',f7.2,5x,'rc=',f7.2)
132     format(3x,'z(i)',8x,'s(i)',11x,'v(i)',11x,'ccw(i)')
133     format(/5x,'time=',f7.3,5x,'z(sat)='f7.2/)
134     format(f8.3,3e15.5)

        end
```

Witne

From Page No.

```
!   PSI(S) FOR UNSATURATED FLOW
        function psis(s)
        common/com1/ssat,smin
        common/com12/maspsik
        common/com2/alfa,beta,gamma,qaysat
        common/psi20/psi0,s1
!   maspsik= functions used to clculate psi(s), k(s).
        goto (10,20) maspsik

 10     continue
!   smin= minimal saturation.
!   psi1= suction at s=smin (m).
!   psir= suction at s=1-epsilon (m).
        data psi1,psir/0.9,0.7/
        psis=psir*(s-smin)/(1.-smin)+psi1*(1.-s)/(1.-smin)
        goto 100

 20     continue
!   curve fitting for result from calculations with vG equations.
        psis=-psi0*log((s-s1)/(1.-s1))
        goto 100


 100    return
        end

!   QAY(S) FOR UNSATURATED FLOW.
        function qays(s)
        common/com1/ssat,smin
        common/com12/maspsik
        common/com2/alfa,beta,gamma,qaysat

        goto (10,20) maspsik

 10     continue
!   qay1,qay2= parameters in K(S) eqution, k=k1(s-smin)+k2(s-smin)**2.
        data qay1,qay2/465.5,344.1/
        qays=qay1*(s-smin)+qay2*(s-smin)**2
        go to 100

 20     continue
! Genuchten equation
!       k=(1-(1-(seq)**(1/gamma))gamma)**2*sqrt(seq)
!   where:
!       seq=(s-smin)(ssat-smin)

        seq=(s-smin)/(1.-smin)
        seq=min(seq,1.)
        qay=(1.-(1.-(seq)**(1./gamma))**gamma)**2*sqrt(seq)
        qays=qay*qaysat
        go to 100

 100    return
        end
```

fort.11

```
&in
teta=.33,ro=2260.,qaysat=2.,alfa=.016,beta=1.3,psi0=66.5,s1=0.4,
smin=.041,ssat=.99,rc=1.,
qayd=4.e3,ckdns=1.,ckdnsc=0.,ckdnwc=0.,cnwm=2.4,elamda=0.,alfan=1.e-4,
ddn=.038,dpor=1.e-4,apec=.63,bpec=1.,
sint=.7,ccwint=1.e5,cnwint=0.,ccwout=1.e5,
hc=100.,dz=1.,dt=0.01,tmax=3001.,tpr=300.,dtpr=30.,maspsik=2,masb=2,
hub=.001,t1=10000.,epss=1.e-6,epsnw=1.e-6,epscw=1.e-6,
ccwaq=1.e5,cnwaq=0.,ibt=25,iac=26,
consaq=.1,t21=0.,t22=.2,t23=1.0,virrig=0.05
/
CHnwv_Am
```

To Page No.

Witnessed & Understood by me,	Date	Invented by	Date
		Recorded by	

From Page No.

This scientific notebook records experimental work conducted to evaluate the importance of radionuclides being transported in colloidal form. The test codes described here were not developed for use in preparing regulatory products. Algorithms tested in the codes may be incorporated, at a later date, into the TPA code.

This notebook appears to be have properly maintained.

Gordon Wittmeyer

2/8/2000

To Page No.

Witnessed & Understood by me,	Date	Invented by	Date
		Recorded by	

From Page No.

I have reviewed this scientific notebook and find it in compliance with QAP-001. There is sufficient information used for conducting tests and developing software in this notebook for another qualified individual could repeat the work described herein.

Gordon Wittmeyer

2/15/2000

To Page No.

Witnessed & Understood by me,	Date	Invented by	Date
		Recorded by	

ADDITIONAL INFORMATION FOR SCIENTIFIC NOTEBOOK #: 343

Document Date:	06/10/1999
Availability:	Southwest Research Institute® Center for Nuclear Waste Regulatory Analyses 6220 Culebra Road San Antonio, Texas 78228
Contact:	Southwest Research Institute® Center for Nuclear Waste Regulatory Analyses 6220 Culebra Road San Antonio, TX 78228-5166 Attn.: Director of Administration 210.522.5054
Data Sensitivity:	■"Non-Sensitive" □ Sensitive □"Non-Sensitive - Copyright" □ Sensitive - Copyright
Date Generated:	1999
Operating System: (including version number)	NA
Application Used: (including version number)	Fortran 90
Media Type: (CDs, 3 1/2, 5 1/4 disks, etc.)	1 CD
File Types: (.exe, .bat, .zip, etc.)	Various
Remarks: (computer runs, etc.)	Media contains backup of UNIX system directory