Scientific Notebook
473

Contents

Upper Split Wash Runoff Modeling

RES 9/29/01

This notebook documents adaptation and application of KINEROS2 to conditions in Split Wash for summer and winter hydrologic conditions. Other software may be used when analysis of KINEROS2 performance, or other aspects of Split Wash area hydrology are under study

The KINEROS2 [K2] program is maintained and furnished via David Goodrich and Carl Unkrich at the Southwest Watershed Research Center, ARS USDA, Tucson AZ. (SWRC) I am responsible for several parts of the model, and have continued to maintain parts of the model (not at the expense of SWRC). The FORTRAN code is available through SWRC or through my copy.

Blank page

RES 9/13/01 K2 as designed up to now does not account separately for water (from rainfall) in upper and lower soil layers, nor does it correctly deal with filling the upper layer when a restrictive lower layer causes an infiltration limit and ultimately cause surface runoff due to saturation of the upper layer.

Modified code now contains:

a) logical variable to identify when lower layer is encountered by advancing wetting wave [lowet]. the wave may be saturated or unsaturated, depending on whether rain is greater than Ks1 or less.

b) logical variable [topfil] to note whether upper soil has been saturated.

c) variable (floating point) for storage of infiltrated water in upper layer [ff1(j,i)]. The storage can decrease during redistribution within a storm, as water seeps into the lower layer faster than rainfall.

d) another logical variable [subcon(i)] that denotes a case for which subsoil properties control the production of runoff

Modified K2 code now keeps account of depth and water content of pulse, and reflects that correctly in the value of ff1(j,i). This is implemented for any pattern of rain, which may fluctuate between r<Ks1 and r>Ks1 and r<Ks2 as well

* Ks1 is the saturated hydraulic conductivity of the upper layer. Ks2 is for the lower layer.

Several winter storms furnished by DAW, as well as parameter file for Upper Split Wash, used for testing revisions.

[redacted]

RES 11/12/01 Further tests of layer infiltration accounting. Sublayer infiltration should be difference between total infiltration, I, and storage in upper layer at end of storm, ff1. Call this I_2

Program can report I_2 at each node, but decided with D.A.W. to report I_2 at last node: I_{2n}

To find time when upper layer at node n fills, added this code at top of loop thru nodes:

RES 11/12/01

```
c
      if(subcon(i)) then   ! subsurface control condition:
        if(lowet(j,i)) then
          sku(j) = sk2(i)
          if((ff1(j,i) .ge. cumcm(i))
     &        .and. .not. topfil(j,i)) then
            topfil(j,i) = .true.
            tho(j,i) = ths1(i)
c
            if(j .eq. nk .and. .not. notify(i)) then
              dum = 1.0
              izr = 0                         ← entry type code
              call qwrt (id, izr, trace, cumcm(i), dum, dum,
     &       dum, id, 0, i, t)   ! pass cumcm and t to writer
c               if(diag .and. j .eq. jd) write(99,*) t(i)
              notify(i) = .true.
            end if
          end if
        else
c        if(fi(j,i) .gt. 0. .and. rfj .gt. 1.5*sk1(i)) then
          sku(j) = sk1(i)
        end if
      else
c             surface layer control conditions
```

- qwrt is call to transfer data to the <u>writer</u> module. (See below)

- to record total flow into lower layer, module is needed to provide information in subroutine <u>infil</u> so that last time step can be identified. Program modified with module <u>itpars</u> to furnish this data globally – includes # of steps called "itlim"

RES 11/12/01

- total value I_2 transmitted to writerg by addition of the following code at the end of the computation loop in infil:

```
c-----------------------------------------------------------------------------
      if(itm .ge. itlim .and. subcon(i)) then
        jmns = -1
        nkm = nk-1
        finlom = 0.
        finlot = 0
        if(lowet(1,i)) finlom = (fs(1,i) - ffl(1,i))

RES     do j = 2,~~nkm~~ nk
          finlo = 0.
          if(lowet(j,i)) finlo = (fs(j,i) - ffl(j,i) )
          finlot = 0.5*(finlom + finlo)  + finlot
          finlom = finlo
        end do

c RES    finlo~~t~~ = finlot/real(nkm,4)      <- this is I2
        if(finlo .gt. 1.e-6) then
c          write(77,'(" finlo ",3g13.4)') finlot, fs(2,1), ffl(2,1)
          call qwrt (id, jmns, trace, finlo, dum, dum,
     &       dum, id, 0, i, t)   ! pass cumcm and t to writer
        end if
      end if
c
```

(Handwritten annotations: "error found 3/15/02 RES" with arrows pointing to the `do j = 2, nk` line, the `+ finlot` addition, and the `finlo = finlot/real(nkm,4)` line; "entry type code" pointing to the argument `i` in the `qwrt` call.)

at the end of computations in each element, writerg is called to report in the output file. The data is entered via an entry qwrt, and that entry is used with new codes to provide new output data for element output. New code reported on next page. This now also distinguishes main channel from overbank cases.

[redacted]

[redacted]

RES 2/26/02

new writer entry code:

RES 2/26/02

```
      entry qwrt (id, k, idstr, qpk, tpk, vi, area,
     &             storq, typ, storr, qrmax)

      if(k .eq. 0) then
c
        if(storr .eq. 1) then
          fill2 = .true.
          filtv = qpk*conv
          filtm = qrmax/60.

        else if(storr .ge. 2) then
          fillov = .true.   ! overbank fill
          filtvo = qpk*conv
          filtmo = qrmax/60.
        end if
c
      else if(k .lt. 0) then  ! report loss to lower layer
         lowfin(storr) = .true.
         floss(storr) = qpk*conv
c
      else
```

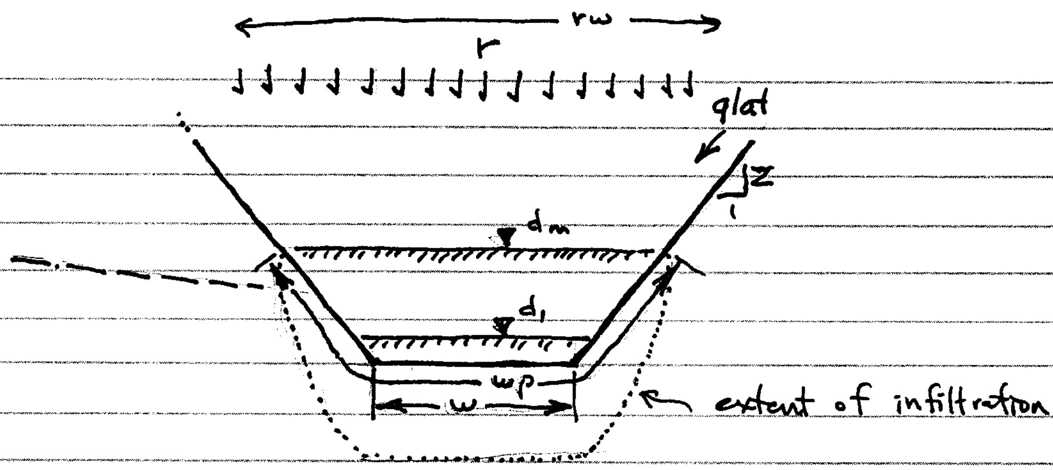
RES 3/15/02 The above code uses the integer k as a flag to indicate the entry of surface soil saturation data. ~~Also, if~~ k is normally positive, so 0 and negative values are used for these additional entry items. The integer <u>storr</u> is further used as a flag for indication of overbank cases, to be properly indicated in the output file.

The drawing on p.7 illustrates an idealized trapezoidal channel subject to infiltration during runoff. When bottom width w is large w/r mean flow depth ~~d~~ d_1, infiltration can be treated as approx. 1-dimensional as done on the plane. Actually in any case KINEROS2 (K2) treats infiltration by area (2-D) rather than depth (1-D) and the wetted perimeter, wp, changes during the flow event as depth does between d=0 and d_m (max depth)

* RES 3/5/02

In this case, special accounting is needed to track what water infiltrates the bed. This should be a minor improvement for the Split Wash channels where w >> d. For 2-layer soils, modifications are needed to account for the general case, especially

RES 3/15/02



- Code has been added to separately account for infiltration in the channel bottow thru $\underline{w}$ rather than the total thru $\underline{wp}$. This is reported for use by Dave W. in looking at the results of long storms on channel leakage.
- Code has been added to K2 for a tabular report. See attachment on next page. Upland areas are separated from channels in the report. Infiltration is divided into parts when 2-layer soils are considered

3/16/02 The accounting must consider that the users may specify a rain width for a channel that is greater or less than the width of flow, and the flow width may be less than the bottom width for the "Woolhiser effective width" option.

let the width on which only rain enters be called $\underline{w_r}$. The width specified by the user for rain-on-channels is $\underline{rw}$. This cannot be less than bottom width, $\underline{w}$. w_r will go to 0 whenever wetted perimeter $w_p \geq rw$. The infiltration routine calculates infiltration rate f based on rw for channels

total infiltration during a timestep is

$$\Delta Vf = r(rw - w_p) + f \cdot w_p \qquad (1)$$
$$= (vfr) + (vfq)$$

the time-weighted values are:

width under flow, $\bar{w}_q = \dfrac{\Sigma\, w_p\, \Delta t}{t}$ (2)

RES 3/15 Insert to p.8

8a

```
C - - - -- - - - - - - - - - - - - - -  optional table of element data
      if(tabl) then

        call blanks
        twol = .false.
        do je = 1,ltab
          if(sumtab(je)%twola) then
            twol = .true.
            exit
          end if
        end do
C
        if(twol) then
          m = 116
          write(colunits(99:114),'("    ",a2,"       ",a2,"    ")')
     &         qlab1,qlab1
          head1(99:115) = str2a !' TopLay  Subsoil'
          head2(99:115) = str2b !'  Infil.  Infil.'
        else
          m = 100
        end if
C
        aref = sumtab(ltab)%cumare
        if(sed) then
          adds1 = 'Sediment'
          adds2 = ' Yield  '
          ms = m+3
          me = ms+1
          colunits(ms:me) = wlab
        end if
C
        me = m+7
        write(head1(m:me),'(a8)') adds1
        write(head2(m:me),'(a8)') adds2
        write(file1,811) head1
        write(file1,811) head2
        write(file1,811) colunits
C                     list upland elements: plane, urban, injects
        do je = 1, ltab
          itu = sumtab(je)%itype
          if(itu .eq. 0 .or. itu .eq. 6 .or. itu .eq. 5) then
            write(idbuff(1:6),'(I5,1x)') sumtab(je)%idel
            idbuff(7:20) = typname(itu)(1:14)
            if(itu .eq. 5) then
              do j = 1,8
                tabstrng(j) = empty
              end do
            else
              call fmt10 (sumtab(je)%are, aref, tabstrng(1), j)
              call fmt10 (sumtab(je)%cumare, aref, tabstrng(2), j)
              write(tabstrng(3),'(1x,f9.4)') sumtab(je)%volin
              write(tabstrng(4),'(1x,f9.4)') sumtab(je)%volrn
            end if
            write(tabstrng(5),'(1x,f9.4)') sumtab(je)%volro
            write(tabstrng(6),'(2x,f8.3)') sumtab(je)%ropeak
            if (itu .ne. 3) then
              vinfl = sumtab(je)%ftot !/sumtab(je)%are
              write(tabstrng(7),'(1x,f9.4)') vinfl    ! infil in m^3
              write(tabstrng(8),'(4x,f6.4)') sumtab(je)%thst
            end if
            if(twol) then
              nx = 10
              js(9) = 2            ! first character of string to use
              js(10) = 2
              tabstrng(9) = empty  ! upper layer infil
              tabstrng(10) = empty  !lower layer infil
              if(sumtab(je)%twola) then
                flinmm = sumtab(je)%flowr * conv
                write(tabstrng(10),'(2x,f8.3)') flinmm
                upper = vinfl/sumtab(je)%are *conv - flinmm
                write(tabstrng(9),'(2x,f8.3)') upper
              end if
            else
              nx = 8
            end if
C            tabstrng(nx) = empty
            if(sed) then
              nx = nx+1
```

RES 3/15 insert to p. 8


```
C - - - - - - - - - - - - - - - - - - - optional table of element data
      if(tabl) then

        call blanks
        twol = .false.
        do je = 1,ltab
          if(sumtab(je)%twola) then
            twol = .true.
            exit
          end if
        end do
C
        if(twol) then
          m = 116
          write(colunits(99:114),'("     ",a2,"       ",a2,"    ")')
     &        qlab1,qlab1
          head1(99:115) = str2a !' TopLay  Subsoil'
          head2(99:115) = str2b !'  Infil.  Infil.'
        else
          m = 100
        end if
C
        aref = sumtab(ltab)%cumare
        if(sed) then
          adds1 = 'Sediment'
          adds2 = ' Yield  '
          ms = m+3
          me = ms+1
          colunits(ms:me) = wlab
        end if
C
        me = m+7
        write(head1(m:me),'(a8)') adds1
        write(head2(m:me),'(a8)') adds2
        write(file1,811) head1
        write(file1,811) head2
        write(file1,811) colunits
C                    list upland elements: plane, urban, injects
        do je = 1, ltab
          itu = sumtab(je)%itype
          if(itu .eq. 0 .or. itu .eq. 6 .or. itu .eq. 5) then
            write(idbuff(1:6),'(I5,1x)') sumtab(je)%idel
            idbuff(7:20) = typname(itu)(1:14)
            if(itu .eq. 5) then
              do j = 1,8
                tabstrng(j) = empty
              end do
            else
              call fmt10 (sumtab(je)%are, aref, tabstrng(1), j)
              call fmt10 (sumtab(je)%cumare, aref, tabstrng(2), j)
              write(tabstrng(3),'(1x,f9.4)') sumtab(je)%volin
              write(tabstrng(4),'(1x,f9.4)') sumtab(je)%volrn
            end if
            write(tabstrng(5),'(1x,f9.4)') sumtab(je)%volro
            write(tabstrng(6),'(2x,f8.3)') sumtab(je)%ropeak
            if (itu .ne. 3) then
              vinfl = sumtab(je)%ftot !/sumtab(je)%are
              write(tabstrng(7),'(1x,f9.4)') vinfl    ! infil in m^3
              write(tabstrng(8),'(4x,f6.4)') sumtab(je)%thst
            end if
            if(twol) then
              nx = 10
              js(9) = 2            ! first character of string to use
              js(10) = 2
              tabstrng(9) = empty  ! upper layer infil
              tabstrng(10) = empty  !lower layer infil
              if(sumtab(je)%twola) then
                flinmm = sumtab(je)%flowr * conv
                write(tabstrng(10),'(2x,f8.3)') flinmm
                upper = vinfl/sumtab(je)%are *conv - flinmm
                write(tabstrng(9),'(2x,f8.3)') upper
              end if
            else
              nx = 8
            end if
C            tabstrng(nx) = empty
            if(sed) then
              nx = nx+1
```

```
            js(nx) = 1
            write(tabstrng(nx),'(1x,f9.3)') sumtab(je)%sedout
          end if
          write(file1,810) idbuff,(tabstrng(j)(js(j):10),j=1,nx)
        end if
      end do
c                                       now do channels and pipes:
      do je = 1, ltab
        itu = sumtab(je)%itype
        if(itu .eq. 1 .or. itu .eq. 2 .or. itu .eq. 4) then
          write(idbuff(1:6),'(I5,1x)') sumtab(je)%idel
          idbuff(7:20) = typname(itu)(1:14)
          if(itu .ne. 2) then
            call fmt10 (sumtab(je)%are, aref, tabstrng(1), j)
          else
            tabstrng(1) = empty
          end if
          call fmt10 (sumtab(je)%cumare, aref, tabstrng(2), j)
          write(tabstrng(3),'(1x,f9.4)') sumtab(je)%volin
          write(tabstrng(4),'(1x,f9.4)') sumtab(je)%volrn
          write(tabstrng(5),'(1x,f9.4)') sumtab(je)%volro
          write(tabstrng(6),'(2x,f8.3)') sumtab(je)%ropeak
          if(itu .eq. 2) then    !pipe case
            nx = 8
            if(twol) nx = 10
            do j=7,nx
              tabstrng(j) = empty
            end do
c                           (infiltrating channels)
          else
            vinfl = sumtab(je)%ftot !/sumtab(je)%are
            write(tabstrng(7),'(1x,f9.4)') vinfl
c          end if
            write(tabstrng(8),'(4x,f6.4)') sumtab(je)%thst
            if(twol) then
              nx = 10
              js(9) = 2
              js(10) = 2
              tabstrng(9) = empty   ! upper soil infil
              tabstrng(10) = empty  ! lower layer infil
              if(sumtab(je)%twola) then
                flinmm = sumtab(je)%flowr * conv
                write(tabstrng(10),'(2x,f8.3)') flinmm
                upper = sumtab(je)%vbot/sumtab(je)%abot*conv - flinmm
                write(tabstrng(9),'(2x,f8.3)') upper !sumtab(je)%pored
              end if
            else
              nx = 8
            end if
          end if
c
          if(sed) then
            nx = nx+1
            js(nx) = 1
            write(tabstrng(nx),'(1x,f9.3)') sumtab(je)%sedout
          end if
          write(file1,810) idbuff,(tabstrng(j)(js(j):10),j=1,nx)
        end if
      end do
c                                       now do ponds
      do je = 1, ltab
        itu = sumtab(je)%itype
        if(itu .eq. 3 ) then
          write(idbuff(1:6),'(I5,1x)') sumtab(je)%idel
          idbuff(7:20) = typname(itu)(1:14)
          call fmt10 (sumtab(je)%are, aref, tabstrng(1), j)
          call fmt10 (sumtab(je)%cumare, aref, tabstrng(2), j)
          write(tabstrng(3),'(1x,f9.4)') sumtab(je)%volin
          write(tabstrng(4),'(1x,f9.4)') sumtab(je)%volrn
          write(tabstrng(5),'(1x,f9.4)') sumtab(je)%volro
          write(tabstrng(6),'(2x,f8.3)') sumtab(je)%ropeak
          vinfl = sumtab(je)%ftot !/sumtab(je)%are
          write(tabstrng(7),'(1x,f9.4)') vinfl
c          end if
          tabstrng(8) = empty
c          write(tabstrng(8),'(4x,f6.4)') sumtab(je)%thst
          if(twol) then
            nx = 10
```

```
            js(9) = 2
            js(10) = 2
            tabstrng(9) = empty    !  upper layer infil
            tabstrng(10) = empty   !  lower soil infil
          else
            nx = 8
          end if
c
          if(sed) then
            nx = nx+1
            js(nx) = 1
            write(tabstrng(nx),'(1x,f9.3)') sumtab(je)%sedout
          end if
          write(file1,810) idbuff,(tabstrng(j)(js(j):10),j=1,nx)
        end if
      end do
c
      end if
 810  format(a20,5a10,4a9,2a8,3a10)
 811  format(a123)
c
```

RES 3/15/02

width of rain only = $\bar{w}_r = \frac{\Sigma (rw - w_p)\Delta t}{t}$ (3)

note that $(rw - w_p)$ is 0 for all $w_p \geq rw$

In addition, the total mean width can be found by an infil. depth weighted total. Let (from [1])

$$TV_f = \Sigma \Delta vf = \Sigma \bar{V}_{fr} + \Sigma \bar{V}_{fq}$$

then

$$\bar{w} = \frac{\Sigma \bar{V}_{fr}}{TV_f}(\bar{w}_r) + \frac{\Sigma \bar{V}_{fq}}{TV_f}(\bar{w}_q) \quad (4)$$

where $\bar{w}$ is an average over both space and time.

Note: in the channels of Split Wash studied by Dave Woolhiser, the flows are generally shallow w/r to the width, and this computation is generally unnecessary.

On p.10 an example of the output table produced for element hydrologic analysis.

RL 4/14/2008

Tabular Summary of Element Hydrologic Components

ID	Element Type	Areas Element m^2	Areas Cumulated m^2	Inflow m^3	Rainfall m^3	Outflow m^3	Peak Flow mm/h	Total Infil m^3	Initial Water Content	Upper Layer: Max Stor mm	Upper Layer: Infil. mm	Subsoil Infil. mm
1	Plane	1787.40	1787.40	0.000	46.472	0.000	0.00	46.47	0.1356	187.43	26.00	0.00
3	Plane	840.95	2628.35	0.000	21.865	4.241	1.71	17.62	0.1375	92.73	17.28	3.67
5	Plane	1905.70	4534.05	4.247	49.548	13.690	2.59	40.11	0.1375	55.64	17.29	3.76
7	Plane	2592.60	7126.65	13.707	67.408	26.436	2.99	54.68	0.1375	148.37	17.37	3.72
9	Plane	3133.47	10260.12	26.462	81.470	17.564	1.63	90.40	0.1356	187.43	28.85	0.00
11	Plane	2398.82	12658.94	17.553	62.369	10.747	1.04	69.22	0.1356	281.14	28.86	0.00
2	Plane	2410.80	2410.80	0.000	62.681	0.000	0.00	62.68	0.1356	28.11	15.29	10.71
4	Plane	1026.38	3437.18	0.000	26.686	5.174	1.56	21.51	0.1375	46.37	17.27	3.69
6	Plane	1997.04	5434.22	5.181	51.923	15.069	2.36	42.04	0.1375	129.83	17.29	3.76
8	Plane	2092.72	7526.94	15.087	54.411	25.338	2.71	44.16	0.1375	185.46	17.39	3.71
10	Plane	1425.06	8952.00	25.364	37.052	19.787	2.00	42.63	0.1356	337.37	29.92	0.00
12	Plane	1435.20	10387.20	19.798	37.315	13.386	1.46	43.75	0.1356	337.37	30.48	0.00
13	Plane	1809.50	1809.50	0.000	47.047	0.000	0.00	47.05	0.1356	112.46	15.82	10.18
14	Plane	471.60	2281.10	0.000	12.262	0.000	0.00	12.26	0.1356	337.37	26.00	0.00
15	Plane	551.88	551.88	0.000	14.349	0.000	0.00	14.35	0.1356	28.11	15.29	10.71
16	Plane	271.56	823.44	0.000	7.061	1.368	1.71	5.69	0.1375	37.09	17.26	3.70
17	Plane	684.64	1508.08	1.370	17.801	4.739	2.53	14.43	0.1375	148.37	17.33	3.75
18	Plane	668.10	2176.18	4.745	17.371	7.957	2.97	14.16	0.1375	222.56	17.48	3.71
19	Plane	627.20	2803.38	7.963	16.307	3.213	0.93	21.07	0.1356	149.94	23.59	10.01
20	Plane	1168.86	3972.24	3.205	30.390	1.239	0.25	32.38	0.1356	112.46	17.02	10.69
21	Plane	455.00	4427.24	1.236	11.830	0.902	0.16	12.17	0.1356	937.14	26.74	0.00
22	Channel	30.00	29754.48	25.017	0.000	23.623	0.96	1.40	0.0900	86.99	43.43	3.22

RL 4/14/2008

RL 4/14/2008

3.27.02 RES

Comparing KINEROS2 Infiltration Approximations with Richards' Equation.

Objectives: Solve Richards's Equation for examples of the layered conditions found at Upper Split Wash in comparison with the approximations of KINEROS2. Determine if any significant bias can be found.

General Approach: Use the well-tested GNFLUX program to represent cases taken from Split Wash. Small modifications will be made as needed to obtain relevant data.

Richards' equation represents unsaturated flow by the Darcy equation:

$$q = -K(\psi)\frac{dH}{dz} \qquad (5)$$

where H is total potential (length), $K(\psi)$ is hydraulic conductivity [L/T], ψ is capillary head (negative, [L]), and z is depth measured downward. We assume 1-D flow.

RES 3/27 Total head $H = \psi - z$ (z opposite dir. to grav. potential)

so

$$q = -K(\psi)\frac{d\psi}{dz} + K(\psi) \qquad (6)$$

The numerical solution model GNFLUX has been applied in many published studies, i.e.:

Smith, R.E., C. Corradini, & F. Melone, "Modeling infiltration for multistorm runoff events," Water Resources Res., V.29(1):133-144, 1993.

3/27/02 RSS

Equation (6) is used, as well as total water balance, to evaluate infiltration rate at the surface ($z=0$). RSS 3/27 It is ~~also~~ also calculated throughout the wetting zone. We will also use it to calculate flux from the upper soil layer into the lower. In addition, we can use the model to study the filling of the upper layer when it is relatively shallow, and rain intensities are low. This case is treated with certain approximations in KINEROS2.

In numerical finite difference (FD) form, equation (6) becomes:

$$q_{i,i+1} = \alpha\left\{\hat{K}(\bar{\psi})\left[\psi_i - \psi_{i+1}\right]^j + \hat{K}_g(\psi_i)\right\}^j + (1-\alpha)\left\{\hat{K}(\bar{\psi})\left[\psi_i - \psi_{i+1}\right]^{j-1} + \hat{K}_g(\psi_i)\right\}^{j-1} \quad (7)$$

in which

- j is a time step index
- i is a space step (node) index
- α is a time weighting (~ 0.7)
- $\hat{K}$ is an effective K value for flow between soil locations with values ψ_i and ψ_{i+1}. GNFLUX takes an integral mean based on the $K(\psi)$ relation
- $\hat{K}_g$ is effective K for gravitational flux, based on the upper node value ψ_i

We can apply this eqn. (7) to flow across a soil interface by calculating an effective internodal K based on a linear gradient of ψ through each soil with equation of flux across the boundary:

$$q = K_1 \frac{d\psi}{dz_1} = K_2 \frac{d\psi}{dz_2}$$

3/27/02 RES

Since GNFLUX calculates an array of q_i during each timestep, the model only need be modified to provide relevant output. The auxiliary output file (AUX) contains flux vs time values, and columns are added to represent q at the soil boundary. Also, net addition of water for the total profile is calculated (CUMIN), and an additional value for the second layer (CUM2) is created to look at additions in the second layer.

New code for flux rate into the second layer is:

File: Edit2 3/27/2002, 8:58:18AM

```
c
        obflx = boflx
        If(ntsoil .gt. 1) then
          Do i = 2, ms
            if(ntype(i) .ne. ntype(i-1)) then  ! soil interface
              boflx = flowi(i)  ! to *60.cm/hr
              rdelf2 = abs(boflx-obflx)/rbase
              if(rdelf2 .gt. rdelf) rdelf = rdelf2
              exit
            end if
          End Do
        Else
          boflx = 0.
        End If
```

Horton Case: The disturbed section, element 3, has a relatively low upper soil K_s, and storm G3.87.1 provides an example of Horton (surface-control) runoff. Since the infiltration model in K2 is analytically derived for the Horton case, we expect it to be well represented by the Richard's solution.

The soil parameters for Plane 3 are:

Soil	K_s (mm/h)	G (mm)	λ	θ_s	θ_i	
1	1.9	80	.25	.323	.1375 ~~.1375~~ ~~.106~~	RES 3/27/02
2	.685	50	.25	.19		

3/27/02 RES

For GNFLUX, we must design a soil that manifests G with the given λ. Thus the retention relation is found based on a scaling value* to match a G of about 80 mm. Using ψ_b of 66 mm and d of 10 mm, (* ψ_b) G of 82 is found.

The retention and relative conductivity relation in GNFLUX is:

retention: $$\theta_e = \frac{\theta - \theta_r}{\theta_s - \theta_r} = \left[1 + \left(\frac{\psi + d}{\psi_b}\right)^c\right]^{-\frac{\lambda}{c}} \quad (8)$$

rel. K = $$k_r = \theta_e^{\epsilon} \qquad \left[1 + \left(\frac{\psi + d}{\psi_b}\right)^c\right]^{\frac{-2-3\lambda}{c}} \quad (9)$$

and G by definition is

$$G = \int_{-\infty}^{0} k_r(\psi)\, d\psi \quad (10)$$

for plane 3: results: (Note that the following parameters do not match: θ_i, G ~~especially~~) RES 3/27/02

model	rain	infiltration	θ_i	G
K2	26 mm	20.2	.1375	80
GNFLUX	26 mm	22.1	.108	82
"	"	21.6	.136	82

The difference in θ_i may be significant, as shown.
The results are quite acceptable, given the approximations in K2

[redacted]

5/27/02 Comparison of the results of K2 with the solution using a more sophisticated solution of Richards' equation [GNFLUX] is extended to several storms and several elements of the K2 model of Split Wash. ridge top elements are chosen since they are not subjected to runon. Elements 2, 23 and 170 represent a variety of upper soil thicknesses. Storms from the study of D. Woolhiser are chosen to represent the cases most extreme in givin saturation overland flow with large rainfall depths (winter storms - long & deep)

Values of G can be matched better than done previously. G values of 50 and 60 are attained by fitting ψ_B more closely prior to long runs. Errors within 1 or 2%:

K2 G	GNFLUX G	ψ_B	λ
50 mm	50.1	41.6	.25
60	60.5	48.8	.25

Results shown on table, p. 18

Graphs for various cases also attached.

GNFLUX program output (only) modified to account directly for flow into the second layer. The rainfall data is complex and long, but is handled well by GNFLUX.

Table of Results: GNFLUX vs K2

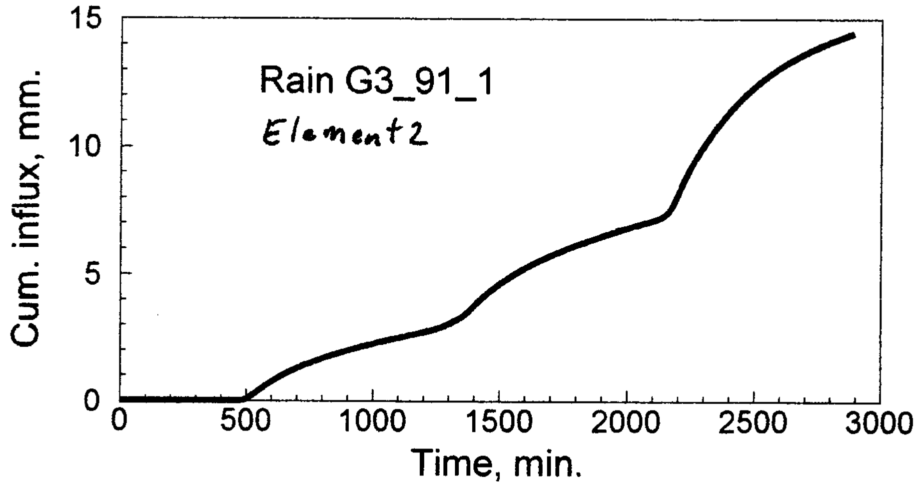
① CASE	Elem. No.	⑥ Z_1 mm	⑦ K_2 mm/h	rain ID	② t_e min	③ t_p min	rain mm	Res ~~with~~ GNFLUX ro	④ I_{2e}	⑤ I_{2p}	K2 ro	I_{2e}	θ_i
A	2	150	.68	G3_91_1	2192	2892	32	0	8.2	14.9	0	8.7	.137
B	2	150	.68	S18_95_2	7263	7982	86.7	12.4	45.9	54.4	22	36.9	.135
C	23	300	.68	S18_95_2	7263	8182	86.7	0	34	47.6	7.5	23.8	.135
D	170	120	.503	G3_91_1	2192	3092	32	0	10.5	17.2	0.9	10.0	.137
E	170	120	.503	S18_95_2	7263	8182	86.7	23	40.9	48	33	31.6	.135
F	2	150	.68	G3_92_7	2184	3104	59	10.9	18.1	28.3	16.7	14.1	.124

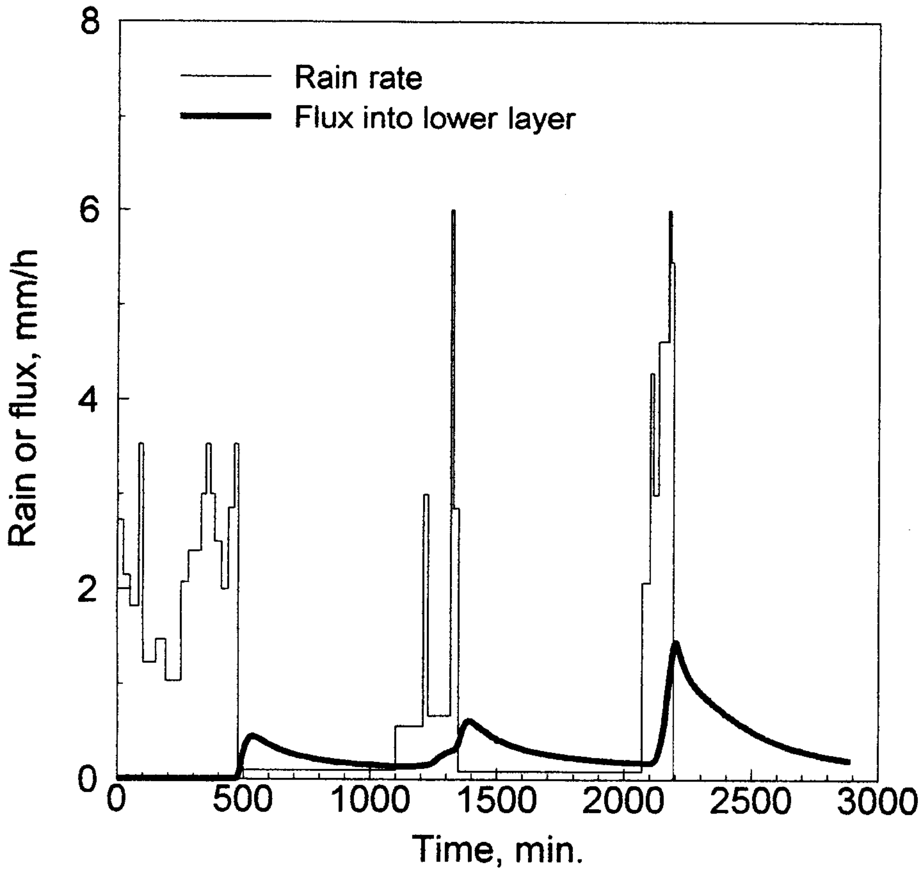
Notes:
1. Full names of input, output files are, e.g. sp12lay*A*.*
2. t_e is end of storm rain
3. t_p is end of extended simulation, illustrating post rain seepage - see graphs
4. I_{2e} is cum. inflow into lower layer at end of rain (t_e)
5. I_{2p} is cum. " at t_p, usually 1/2 to 2/3 day later
6. Z_1 is depth of upper soil layer
7. K_2 is saturated hydraulic conductivity of broken tuff sublayer

RES 5/27/02

Reg 5/27/02

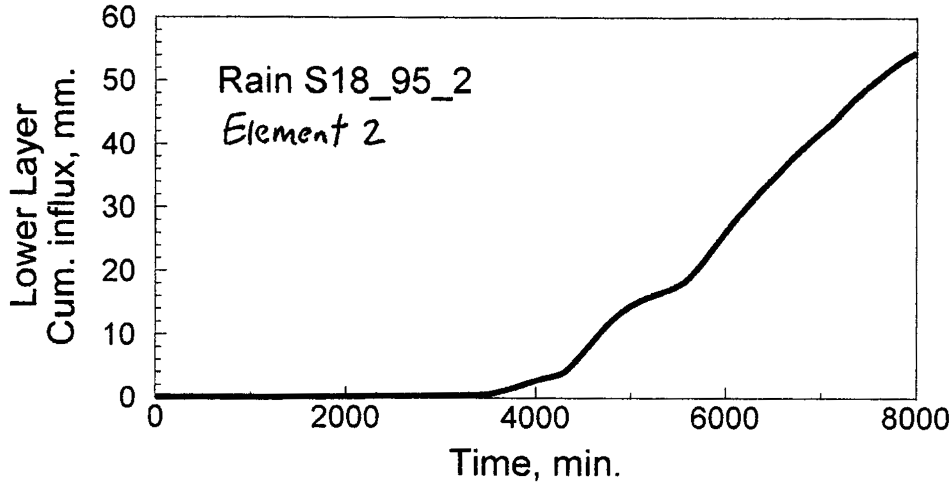
CASE A

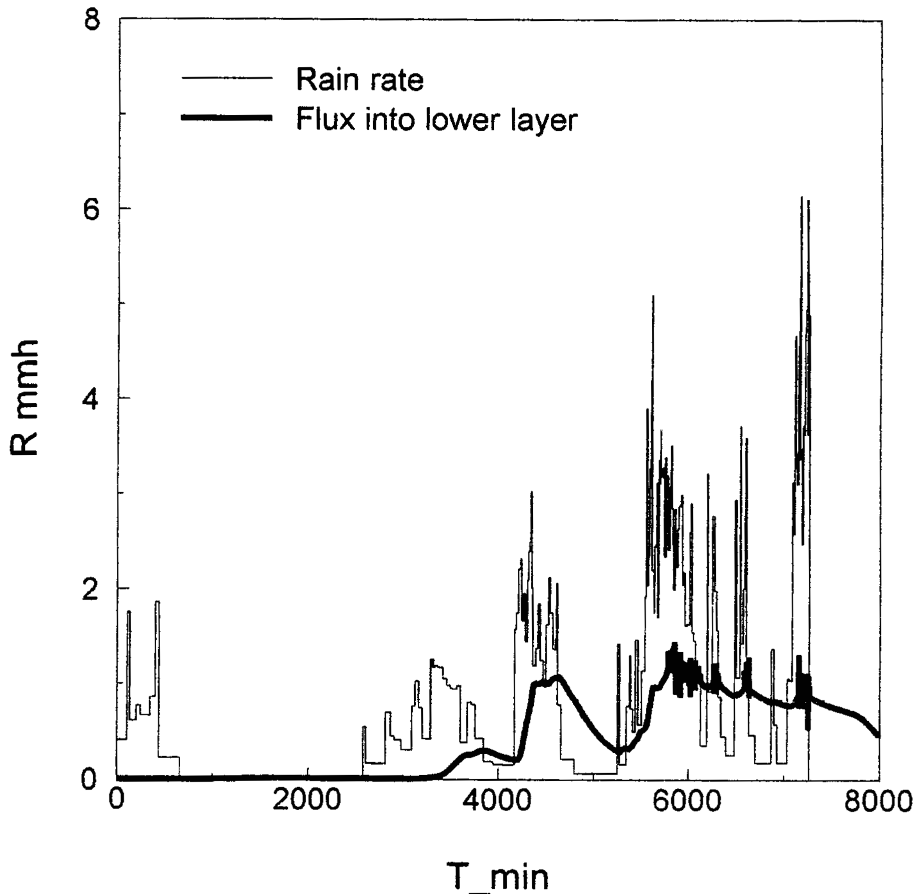




RES 5/27/02

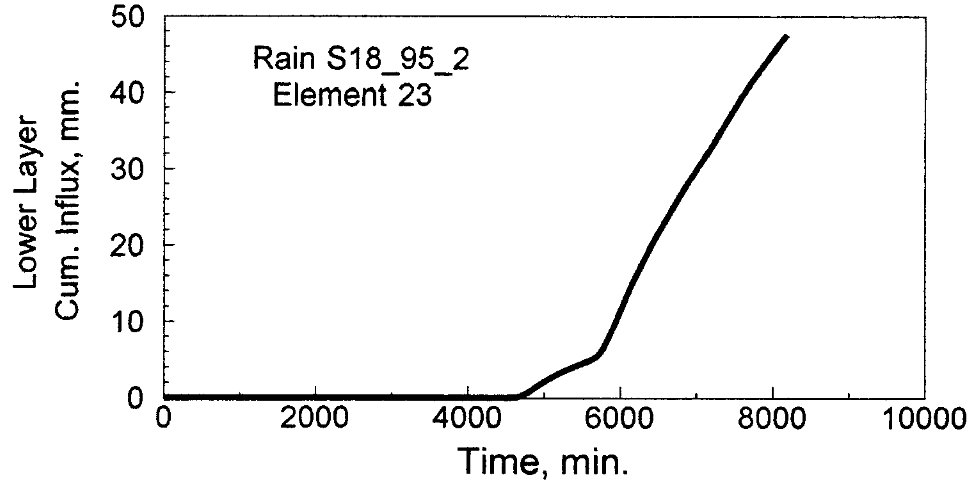
CASE B

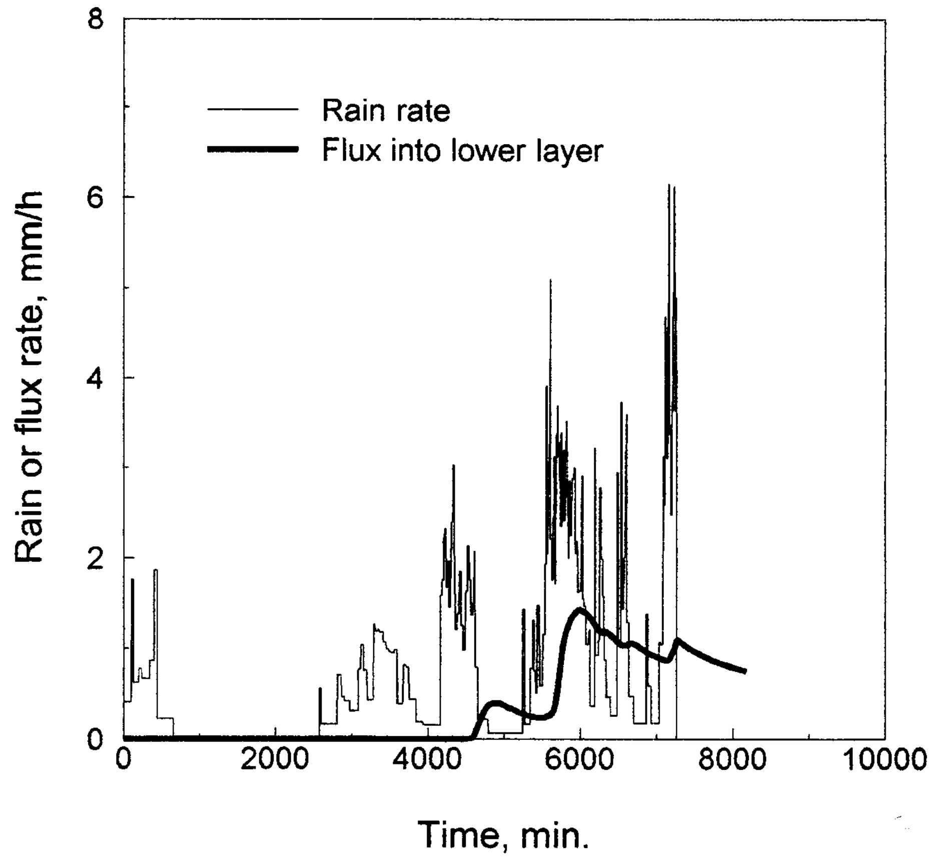




RES 6/02/02

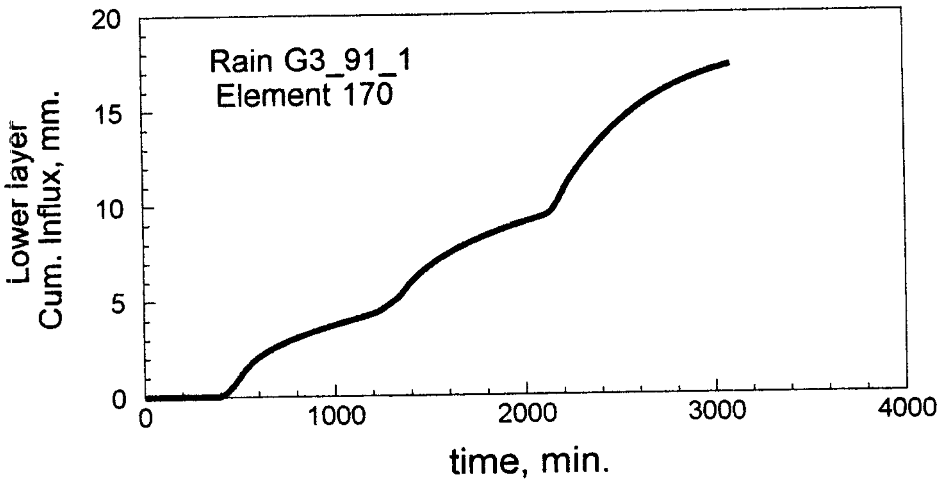
CASE C

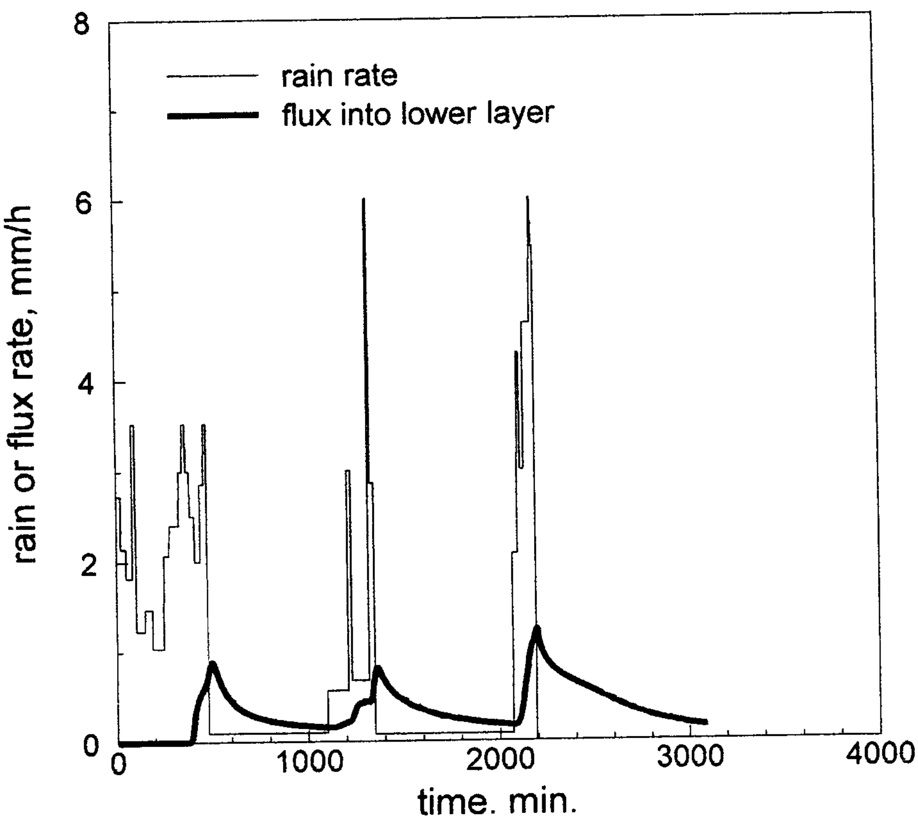




RES 6/2/02

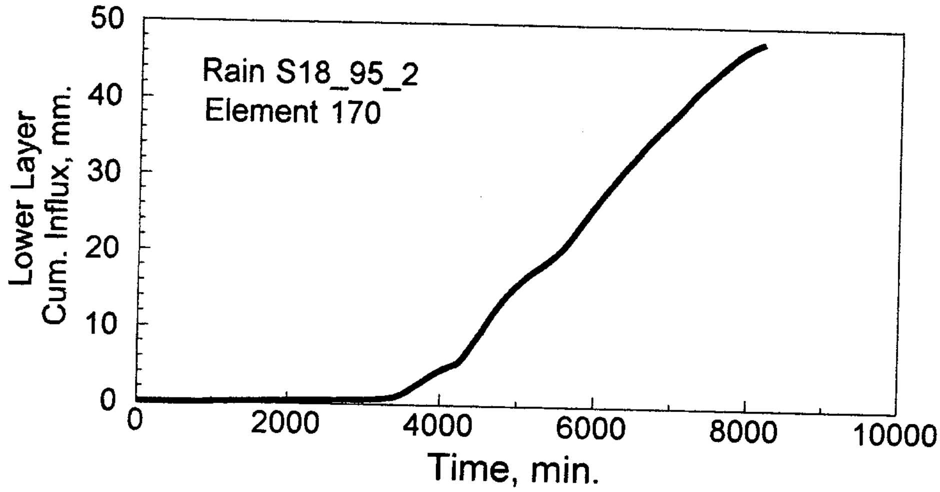
CASE D

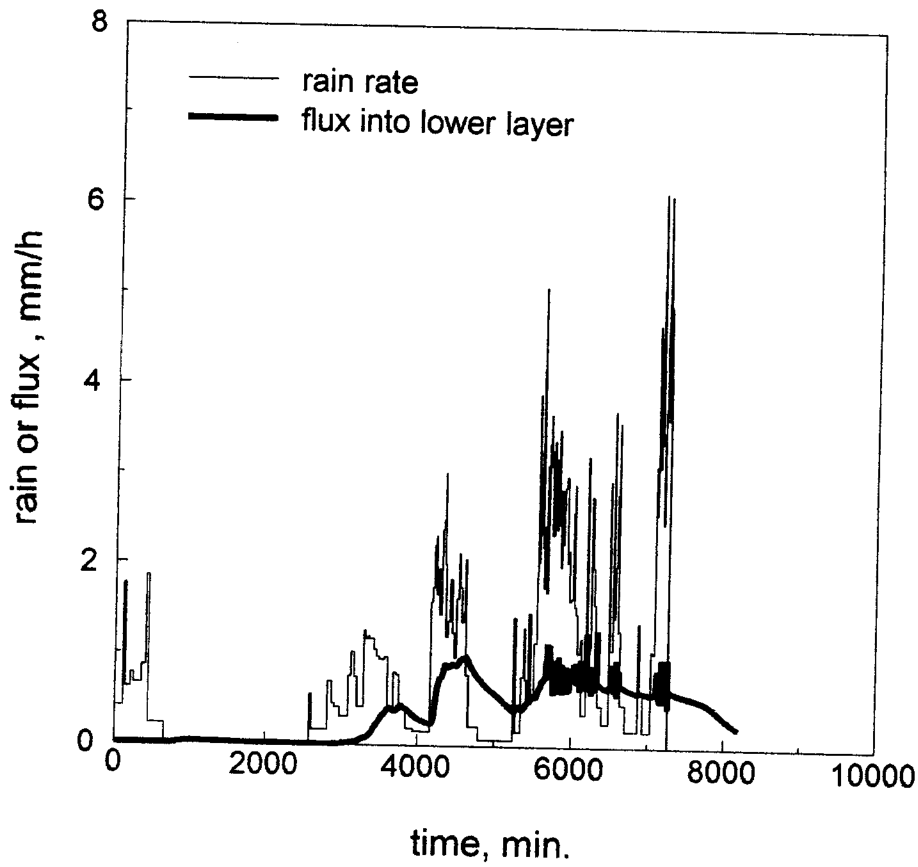




RSS 6/2/02

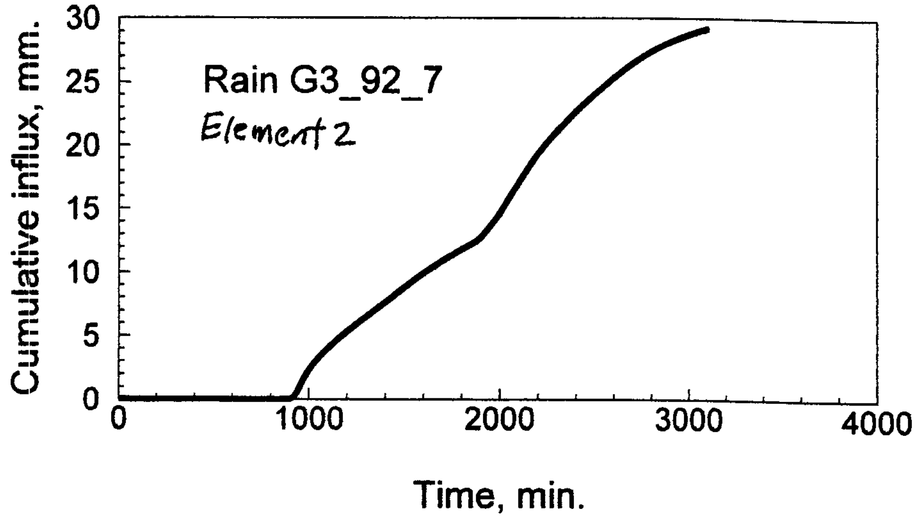
CASE E

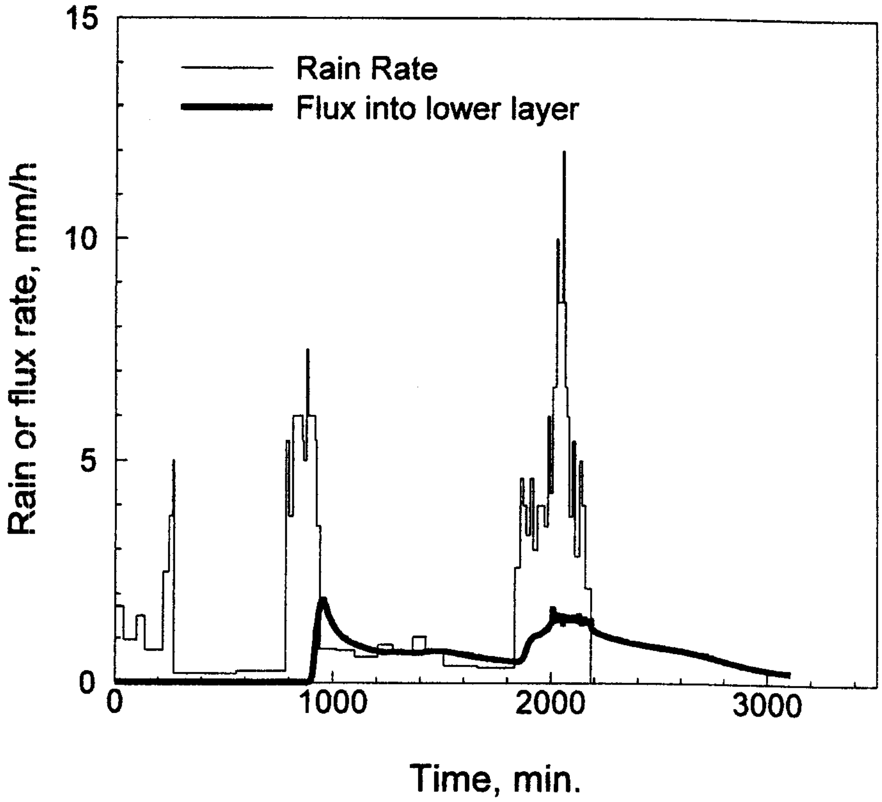




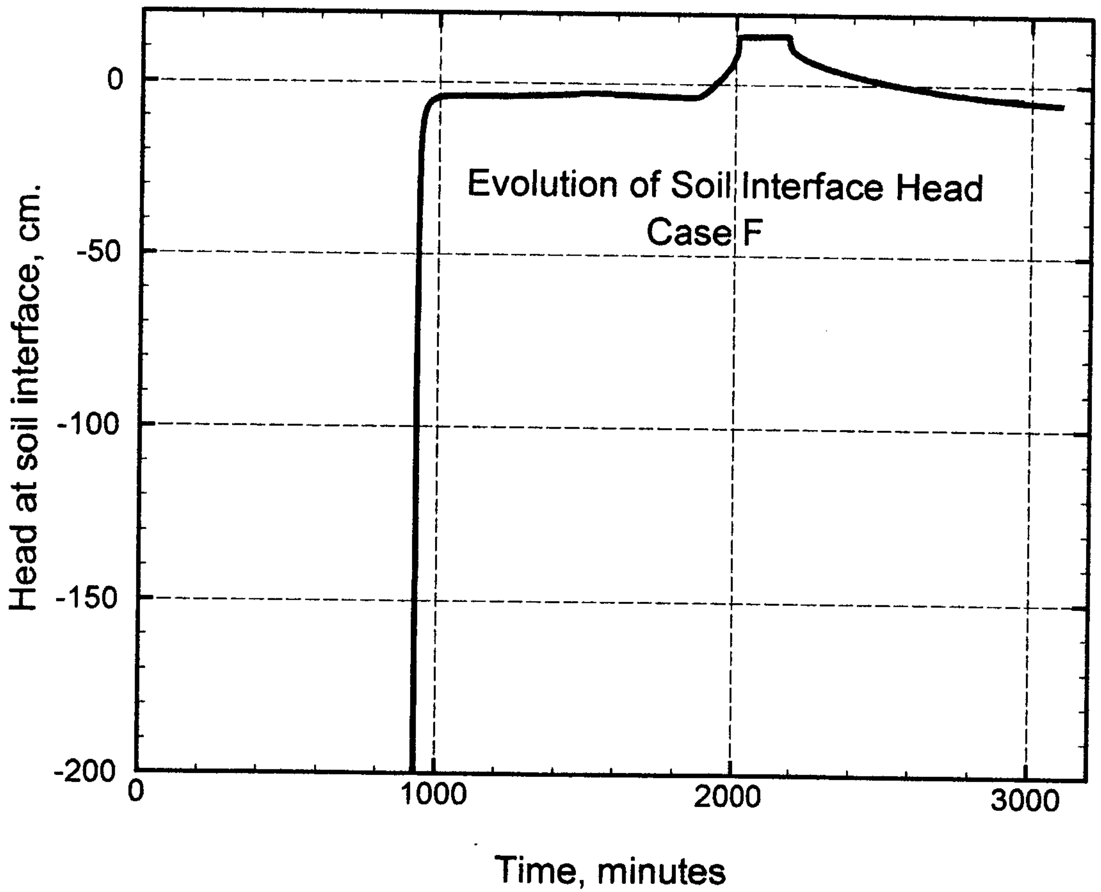
RES 6/2/02

CASE F





RES 6/2/02



Evolution of Soil Interface Head
Case F
Head at soil interface, cm.
0
-50
-100
-150
-200
0
1000
2000
3000
Time, minutes

RES 6/2/02 Analysis of Results. The table on p.18 shows that GNFLUX indicates somewhat higher rates of infiltration into the lower layer for these exceptional cases than does KINEROS2. Looking at the graphs of fluxes in these cases, the major differences are associated with those cases where significant positive heads are formed at the interface when subsurface saturation runoff occurs. This is accompanied by rapid small changes in interface flux (see graphs of cases B, E & F) as the saturated layer responds to rapidly changing rainfall rates.

Since the capillary drive value for the subsoil inflow is relatively small (60 mm) compared to the head on the interface (120 to 150 mm or more), depending on the upper layer depth, K2 could be probably dramatically improved for these cases if head at the interface could be estimated (when upper layer layer approaches saturation)

Thus $\widehat{G_2}$ should $= G_2 + h_c$

This would increase infiltration considerably and bring K2 results more in line with these.

Notice that the first runoff period for case F at around 1000 min corresponds to Horton runoff, just as simulated by K2, and only the latter 'pulse' at about 2000 min. is saturation runoff, somewhat under-predicted by K2.

Also, Case A in Table (p.18) shows excellent agreement comparing lower soil cumulative F at the end of rain. This is a case where there developed no significant positive interface head.

Seepage after the storm, from water in the upper soil layer, was significant in all cases, as shown in the figures on pp. 19-24

RL 4/14/2008

RGS 21/8/02 (Aug.) In order to further estimate the leachable water that can be expected to be available in the period immediately after a storm, DAW and I discussed modification of the output table (p. 10 this book) so that column 12 can show the amount of water at the end of simulation of KINEROS2 which is above field capacity.

Given parameters for Eq. (8) p. 16, field capacity (tho a crude concept) can be taken as 1/3 bar water content (~330 cm).

In K2, ψ_b is estimated, from published data, based on values of λ and G:

$$\psi_b \approx G(2+5\lambda)/(3.9+3\lambda) \qquad (11)$$

K2 uses a simpler version of Eq. (8), which is the Brooks-Corey relation for retention:

RGS 21/8/02

$$\theta_e = S ~~X~~ = (\psi/\psi_b)^{-\lambda} \qquad (12)$$

Thus, with $\psi_b = 330$ and ψ in cm., ~~wilting pt.~~ field capy. (RGS 21/8/02) : θ_f is

$$\theta_f = \theta_r + (\theta_s - \theta_r)\left(\frac{330}{\psi_b}\right)^{-\lambda} \qquad (13)$$

Likewise, wilting pt $[\theta_w]$ would be

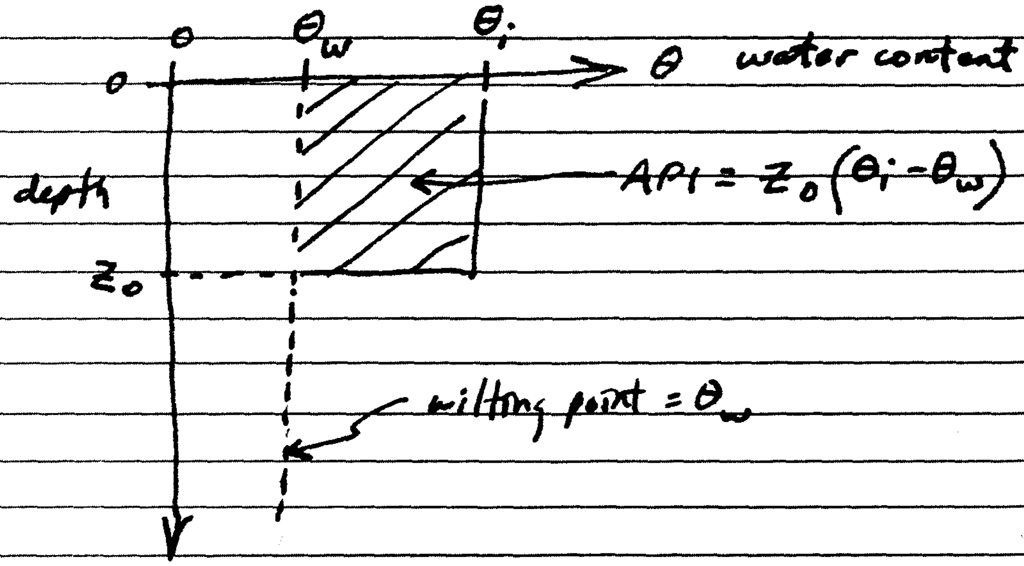
$$\theta_w = \theta_r + (\theta_s - \theta_r)\left(\frac{15300}{\psi_b}\right)^{-\lambda} \qquad (14)$$

which is at head of 15 bars.

This is added to K2, and printed for each element in the output table.

RGS 22/8/02: K2 table changes finished, tested etc.

RES 10 Sept. 02: In order to perform appropriate sensitivity tests of K2 results, (discussions with DW) we decided to make a version of K2 altering only the nature of the initial condition as an option. This would allow specification of an antecedent precip. index (API) such that initial conditions need not be uniform. The initial water saturation parameter, SAT, would in this option not apply to an indefinite depth, but would apply to a depth such that the water above ~~field capacity down to a~~ wilting point would extend to a depth defined by the value of API. Thus: RES 9/10/02



Thus for a given API value, reducing θ_i towards θ_w would result in a ~~lower~~ (RES 9/10/02) deeper value of Z_0.
K2S is a version of K2 with this specific initial condition option. It outputs calculated values of Z_0 and θ_w for each element.

RES 16 Sept. 02 DAW requests API value be read in run file, so K2S version is so altered. New options dialog in run window made

RL 4/14/2008

TL 4/14/2008

RL 4/14/2008

RES 22/8/02 In order to further evaluate the results of K2, especially the initial soil conditions for critical storms, and leaching amounts for various locations, the model Opus will be used, and data prepared.

USGS data from the 1987-94 record is first transformed into format for Opus. A FORTRAN program is written for this purpose, and the data edited to eliminate extremely long periods of apparent (but misleading) very low rates between tips when rainrate is in fact 0. Program written is listed below.

(NOV. 1) RES 1/11/02 Opus rain data file consists of a header line, giving date, no. of data points, and a code ("MIDN") indicating how many midnights the storm passes thru. Thus the time data can reset. This is followed by the data pairs (time, cumulative depth), as many as indicated in the header file.

The USGS data editing consisted of inserting a pair of data points between tips (one tip per mm. rain) to create a fixed end to the previous storm, and a reasonable beginning to the next. Example:

(USGS format):

YR	month	day	hr	cum depth	
1989	1	3	1520	2	
1989	1	3	1600	3	
1989	1	3	1630	3.5	inserted
1989	1	17	745	3.5	inserted
1989	1	17	801	4	
1989	1	17	817	5	

— etc. —

File: D:\OPus2\SplitW\dtransf.for 8/26/2002, 9:34:42AM

```
C --- program to read GS tip data from DAW and write Opus file
      program dtransf
C
      character(LEN=50) :: filein, outfile
      real,dimension(100) :: tw, dw
C     read input file name
      write(*,*)'  Enter name of raw data file: '
      read(*,'(a)') filein
      open(3,file=filein,status='old')
C     read output file name
      write(*,*)'  enter name of file to create: '
      read(*,'(a)') outfile
      open(6,file=outfile,status='unknown')
C
      i = 1
      read(3,101) iyrn, jdan, ihrn, iminn, dmmn
      call monday( jdan, mons, ids, iyrn)
      idp = ids
      jds = jdan
      dmms = dmmn
C
   10 continue
      call monday( jdan, mons, ids, iyrn)
      tw(i) = ihrn*60 + iminn + 1440*(jdan-jds)
      dw(i) = dmmn
      iyro = iyrn
      jdao = jdan
C      ihro = ihrn
C      imino = iminn
      dmmo = dmmn
C
      read(3,101,err=98, end=99) iyrn, jdan, ihrn, iminn, dmmn
C
      if((dmmn - dmmo) .lt. 0.0001 .or. iyrn .gt. iyro) then
C       write event
        midn = jdao - jds
        n = i
        tdep = dmmo - dmms
        write(6,201) iyro, mons, idp, n, midn, tdep
        write(6,202) (dw(i),tw(i),i=1,n)
C
        jds = jdan
        idp = ids
        dmms = dmmn
        i = 1
      else
        i = i + 1
      end if
      go to 10
C
  101 format(2x,i4,i4,2i3,f7.1)
```

RSS 22/8/02

File: D:\OPus2\SplitW\dtransf.for 8/26/2002, 9:34:42AM

```
 201  format(4x,I4,4x,I2,I2,I8,I8,f8.2)
  202 format(5(f8.2,f8.1))
   98 continue
      write(*,*) ' error end '
      go to 199
   99 write(*,*) ' finished input file'
  199 continue
      close (3)
      close (6)
      end
C-----------------------------------------------------------------------
       SUBROUTINE MONDAY(jDATE,MON,DAY,YEAR)
C Finds Month, Day of Month, and Year, given date in GGG format, where
C  GGG is gregorian day
      INTEGER :: jDATE, DAY, MON, MOM, M, YEAR
      integer,dimension(13) :: CAL = (/31,60,91,121,152,182,213,244,274,
     &          305,335,366,0/)
C* CAL(MON) IS THE JULIAN DAY OF LAST DAY OF MONTH ON LEAPYEARS
C
      IF(jDATE .LE. 0) Then
        MON     =13
        RETURN
      Else
C        JDAY   = MOD(jDATE,1000)
        jday = jdate
C        YEAR = jDATE/1000
C* ADD DAY TO GET IN CORRECT PHASE WITH CAL(M)  FOR NON-LEAP YEARS:
        IF(JDAY .GE. CAL(2) .AND. MOD(YEAR,4) .GT. 0) JDAY = JDAY+1
        If(JDAY .gt. 366) Then
          YEAR = YEAR + 1
          JDAY = JDAY-366
        End If
        M = 1
        Do While(JDAY .gt. CAL(M))
          M = M + 1
        End Do
C       M     =13
        MON = M
        MOM = MON - 1
        IF(MON .LE. 1) THEN
          DAY = JDAY
        ELSE
          DAY = JDAY-CAL(MOM)
        END IF
        RETURN
      End If
      END
```

RES Nov 1, 02 Opus requires data on daily max and min. temperatures, as well as radiation, for plant environmental ~~rsp~~ [RES 11/1/02] response in use of soil water. It is possible to estimate daily radiation using the span between daily max. & min. temperature, and elevation and relative humidity, according to results of Running,

** et. al, described below.

The D.O.E. data base web site location was furnished by Randy Fedors:

On this web site, the meteorological data for some or all of 9 met. sites, including hourly or daily data on Temp. and humidity is found in zipped form.

After ~~downloading~~ (RES 11/1/02) studying the available data, the following were downloaded (available by 3-month periods):

Data type	time interval	Period
Temperature	hourly	12/85 - 12/94
daily Max temp	daily	1/95 - 12/97
min. temp	"	"
Rel. humidity	hourly	12/85 - 12/97

Site 3 was chosen as being closest in elevation and location to split wash. However, site 3 proved to have several periods of missing records. In those intervals, substitutions are made using the nearest site. The substitution priorities depend on available sites: early records are for sites 2, 3, 4, 5 and sometimes 1. Site 4 is most desireable substitute followed by 5.

** 4/10/06 see pp. 82+

RES Nov 1, 02

Table

Substitutions used for Relative Humidity data when site 3 data are missing:

dates	subs. site
12/01/85	4
1/29/86 - 1/31/86	"
2/28/86 - 3/09/86	"
3/13 - 3/20/86	"
4/12 - 4/22/86	"
1/10 - 1/11/87	4, w. estimation
11/28 - 11/29/87	4
7/18/87	4
9/08 - 9/14/88	4
9/20/88	"
9/25 - 9/27/88	"
9/28 - 10/05/88	5
10/06 - 10/12/88	4
10/27 - 10/28/88	4
11/2 - 11/10/88	4
2/11 - 2/14/89	4
3/05/89	4
3/25 - 3/28/89	4
3/30 - 4/03/89	4
8/13/89	5
9/12/89	5
10/02 - 10/03/89	5
11/9/90 - 11/11/90	4
11/28 - 12/05/90	4
1/05 - 1/09/91	4
10/22 - 10/23/91	4
2/13 - 2/17/92	4
2/20/92	4
1/17 - 1/18/93	4
8/19 - 8/24/93	2
8/25 - 8/27/93	6
4/23 - 4/24/94	4

dates	subs. site
12/08/96	2

11/1/02 RES

Table

Daily Max and Min Temperature record substitutions (for missing data)

	dates	substitute site
	12/01/85	4
11/1/02 RES	1/31 & 2/~~88~~ 28/86	4 & 5
	3/05 & 3/11/86	4
	3/06 & 3/13/86	4
11/1/02 RES	3/~~15~~ 14/86 - 3/20/86	4
	4/2 - 4/5/86	"
	4/12 - 4/22/86	"
	4/26 - 4/27/86	4
	5/01 - 5/06/86	4
	10/20/86	5
	11/18 - 11/20/86	4
	12/16/86	4m
	1/10/87 - 1/11/87	4
	9/19 - 9/20/88	4
	11/28 - 11/29/87	4
	9/8 - 9/14/88	4
	9/24 - 9/28/88	4
	9/29 - 9/30/88	5
	10/1 - 10/12/88	4 ☆ NOTE: ↗
	11/2 - 11/8/88	4
	11/9/88 & 11/15/88	e4x
	2/11 - 2/14/89	4
	3/5/89	4
	3/25 - 4/3/89	4
	6/1 - 6/2/89	4
	6/14 - 7/19/89	4
	9/12/89	5
	10/2 - 10/3/89	5
	11/9 - 11/11/90	4
	11/28 - 12/5/90	4

dates	substitute site
1/5 - 1/9/91	4
10/22 - 10/23/91	4
2/13 - 2/17/92	4
2/20/92	4
1/17 - 1/18/93	4
8/18 - 8/27/93	1
5/13 - 5/14/96	4
5/20/97	4 x
RES 11/18/02:	
10/12 - 10/13/88	5
☆ 9/29 - 10/05/88	5

notes: e = partial record est.

x = max T

m = min T

11/1/02 RESmith

Since the temperature data came in two time types (hourly and daily min or max) for two parts of the record, two programs were written to extract the data and prepare files for Opus 2. One program reads the text files for a selected site (1 thru 5, usually available), in hourly values, and finds the max and min for a given day, writes both values and the date on the output file, and goes on. A list of sequential input files is provided as input, since each DOE file only covers 1/4 year. This program is shown following, pp. ~~40-42~~ 43-44 (RES 11/1/02). This applies to records up to end of 1994

A second program is for the records of daily max and min available starting in 1995. This program reads pairs of raw data files – one max and one min T, and finds both values for a given day, and writes this to the named output file using the same format that can be read by Opus 2.

A third program, like the first, reads the DOE relative humidity data and finds the daily average of the hourly values. Both the first and third programs also count the number of hours with missing data and report that, so that substitutions can be made where necessary.

The second program is on pp. 40-42,

The relative humidity data transform program is on pp 45-47

File: C:\Applications\OPus2\SplitW\transpdmx.for 10/18/2002, 11:10:14

```
C this program reads USGS quarterly daily tmax and tmin files
C and writes daily max and min temps in Opus format on a file
C whose name is specified.
C
      program ymdmxmn
C
      character(LEN=50) :: opusfil, qfilmin, qfilmax, blank50,
     & filist
      character(LEN=20) :: filmx, filmn
      integer :: nsite, lumax, lumin, lunout
C
      blank50 = '                                                  '
      write(*,'("  Enter name for OPus actdat file to create: ")')
      read(*,'(a)') opusfil
      lunout = 8
      open(8,file=opusfil,status='unknown',err=998)
      write(*,'("  Which Met site do you wish to compile?: ")')
      read(*,*) nsite
      write(8,92) nsite
C
      write(*,91)
   91 Format(' Enter filename for list of files to process:  ')
      read(*,'(a)') filist
      open(7,file=filist,status='old',err=998)
C   91 format('  Enter name of the next quarterly TdMX file to use:'
C     &/'   ')
C   93 format('  Enter name of the corresponding TdMN file to use:'
C     &/'   ')
   92 format("mm/dd/YY   met site ",i2)
    1 continue
       read(7,'(a)') qfilmax
       if(qfilmax .eq. blank50) go to 999
       open(4,file=qfilmax,status='old',err=998)
       filmx = trim(qfilmax)
       lumax = 4
C
       read(7,'(a)') qfilmin
       if(qfilmin .eq. blank50) go to 999
       open(3,file=qfilmin,status='old',err=998)
       filmn = trim(qfilmin)
       lumin = 3
       call reader(lumax, lumin, nsite, lunout, filmx, filmn)
       close (3)
       close (4)
       go to 1
C
  998 stop ' error opening input file '
  999 write (*,'(" end specified ")')
      close (7)
      end
C
C --------------------------------------------------
C
      subroutine reader(lmax, lmin, nsite, lout, filmx, filmn)
      character(len=1) :: first
```

File: C:\Applications\OPus2\SplitW\transpdmx.for 10/18/2002, 11:10:14A

```
      character(LEN=4) :: site = 'SITE', dsite
      character(LEN=10) :: chdate
      character(LEN=20) :: filmx, filmn
      character(LEN=5) :: chour
      integer :: nsite, jsite, nrow, ldate, jdate, jd, jm, jy, mdate,
     &           lmax, lmin
      real :: th, tmin, tmax
c
      linek = 0
      do while (.true.)
        read(lmax,'(a1)') first
        linek  = linek + 1
        if(linek .gt. 5. and. first .eq. '1') exit    ! found first line of
maxT data
      end do
      backspace (lmax)
c
      linek = 0
      do while (.true.)
        read(lmin,'(a1)') first
        linek  = linek + 1
        if(linek .gt. 5. and. first .eq. '1') exit    ! found first line of
minT data
      end do
      backspace (lmin)
      jsite = 0
c
      do while(jsite .ne. nsite)
        read(lmin,*,end=96) nrow, tmin, chdate, chour, dsite, jsite
c         write(lout,'(I5,1x,a10,1x,a5,1x,a4,i2)')
c     &   nrow, chdate, chour, dsite, jsite
      end do
c
      jsite = 0
      do while(jsite .ne. nsite)
        read(lmax,*,end=97) nrow, tmax, chdate, chour, dsite, jsite
c         write(lout,'(I5,1x,a10,1x,a5,1x,a4,i2)')
c     &   nrow, chdate, chour, dsite, jsite
      end do
c                                  found first line of site, each file
      read(chdate,49) jm, jd, jy
      mdate = 1000000*jm + 10000*jd + jy
      write(lout,48) mdate, tmin, tmax   !, miss
   49 format(i2,1x,i2,1x,i4)
c
    1 continue
c      jdate = jd + 100*jm + 10000*jy
c      ldate = jdate
c      do while (jsite .eq. nsite)
        read (lmin, *) nrow, tmin, chdate, chour, dsite, jsite
        read (lmax, *) nrow, tmax, chdate, chour, dsite, jsite
        if(jsite .eq. nsite) then
          read(chdate,49) jm, jd, jy
          mdate = 1000000*jm + 10000*jd + jy
          write(lout,48) mdate, tmin, tmax   !, miss
```

11/1/02 RES

File: C:\Applications\OPus2\SplitW\transpdmx.for 10/18/2002, 11:10:14AM

```
        go to 1
      end if
 48   format(BZ,I8,48x,2f8.1, 6x,i2)
 59   format(1x,a10)
      return
 96   continue
196   format(' site',i2,' data not found in file ',a20)
      write(lout,196) nsite, filmn
      return
 97   continue
      write(lout,196) nsite, filmx
      return
      end
```

TRL 4/14/2008

```
C this program reads USGS quarterly hourly temperature files
C and writes daily max and min temps in Opus format on a files
C whose name is specified.
C
      program yuccahr
C
      character(LEN=50) :: opusfil, qfilen, blank50, filist
      character(LEN=20) :: fname
      integer :: nsite, lunin, lunout
C
      blank50 = '                                                  '
      write(*,'("  Enter name for OPus actdat file to create: ")')
      read(*,'(a)') opusfil
      lunout = 8
      open(8,file=opusfil,status='unknown',err=998)
      write(*,'("  Which Met site do you wish to compile?: ")')
      read(*,*) nsite
      write(8,92) nsite
C
      write(*,91)
      read(*,'(a)') filist
      open(3,file=filist,status='old',err=998)
   91 format('  Enter name list of quarterly USGS data files to use:'
     &/'    ')
   92 format("mm/dd/YY   met site ",i2,t76,"missing")
    1 continue
C       write(*,91)
       read(3,'(a)') qfilen
       if(qfilen .eq. blank50) go to 999
       open(4,file=qfilen,status='old',err=998)
       lunin = 4
       call reader(lunin,nsite,lunout, fname)
       close (4)
       go to 1
C
  998 stop ' error opening input file '
  999 write (*,'(" end specified ")')
      end
C
C ---------------------------------------------------
C
      subroutine reader(lin, nsite, lout, fname)
      character(len=1) :: first
      character(LEN=4) :: site = 'SITE', dsite
      character(LEN=10) :: chdate
      character(LEN=20) :: fname
      character(LEN=5) :: chour
      integer :: nsite, jsite, nrow, ldate, jdate, jd, jm, jy, mdate
      real :: th, tmin, tmax
C
      do while (.true.)
        read(lin,'(a1)') first
        if(first .eq. '1') exit    ! found first line of data
      end do
      backspace (lin)
```

```
      jsite = 0
      do while(jsite .ne. nsite)
        read(lin,*,end=95) nrow, th, chdate, chour, dsite, jsite
C         write(lout,'(I5,1x,a10,1x,a5,1x,a4,i2)')
C     &   nrow, chdate, chour, dsite, jsite
      end do
C                                              found first line of site
      read(chdate,49) jm, jd, jy
   49 format(i2,1x, i2,1x,i4)
      jdate = jd + 100*jm + 10000*jy
C      backspace (lin)
    1 continue
      ldate = jdate
C      if(th .gt. 9998.) th = 999.
      tmin = max(th,999.)    ! initialize
      tmax = min(th,-999.)
      miss = 0
      do while (jsite .eq. nsite)
        read (lin, *, end=96) nrow, th, chdate, chour, dsite, jsite
        read(chdate,49) im, id, jy
        jdate = id + 100*im + 10000*jy
        if(jdate .eq. ldate ) then
          if(th .lt. 999.) then
            if(tmax .lt. th) tmax = th
            if(tmin .gt. th) tmin = th
          else
            miss = miss + 1
          end if
        else
          mdate = 1000000*jm + 10000*jd + jy
          write(lout,48) mdate, tmin, tmax, miss
          jm = im
          jd = id
          if(jsite .eq. nsite) go to 1
        end if
      end do
   48 format(BZ,I8,48x,2f8.1, 6x,i2)
      return
   95 continue
      write(lout,195)nsite, fname
  195 format(' site',i2,' data not found in file ',a20)
      return
   96 continue
      mdate = 1000000*jm + 10000*jd + jy
      write(lout,48) mdate, tmin, tmax, miss  ! in case site is last in list
      return
      end
```

File: C:\Applications\OPus2\SplitW\transpRH.for 10/11/2002, 7:32:22PM

```
C this program reads USGS quarterly hourly Rel Humidity files
C and writes daily ave RH on a file whose name is specified.
C
      program yuccaHUM
C
      character(LEN=50) :: opusfil, qfilen, filist, blank50
      character(LEN=256) :: message
      integer :: nsite, lunin, lunout, listu
C
      blank50 = '                                                  '
      write(*,'("  Enter name for file to create: ")')
      read(*,'(a)') opusfil
      lunout = 8
      open(8,file=opusfil,status='unknown',err=998)
      write(*,'("  Which Met site do you wish to compile?: ")')
      read(*,*) nsite
      write(8,92) nsite
C
      write(*,91)
      read(*,'(a)') filist
      write(*,'(2x,a50)') filist
      open(3,file=filist,status='old',iostat=ierr)
   91 format(' Enter name of sequential list of quarterly '
     &/'  USGS data filenames:  ')
   92 format(" mm/dd/YY   met site ",i2,t28,"missing")
    1 continue
       read(3,'(a)') qfilen
       if(qfilen .eq. blank50) go to 999
       open(4,file=qfilen,status='old',err=998)
       lunin = 4
       call reader(lunin,nsite,lunout,qfilen)
       close (4)
       go to 1
C
  998 stop ' error opening input file '
  999 write (*,'(" end specified ")')
      close (9)
      end
C
C --------------------------------------------------
C
      subroutine reader(lin, nsite, lout, fname)
      character(len=1) :: first
      character(LEN=4) :: site = 'SITE', dsite
      character(LEN=10) :: chdate
      character(LEN=50) :: fname
      character(LEN=20) :: tfname
      character(LEN=5) :: chour
      integer :: nsite, jsite, nrow, ldate, jdate, jd, jm, jy, mdate
      real :: th, tmin, tmax
C
      linek = 0
      do while (.true.)
        read(lin,'(a1)') first
        linek = linek + 1
```

```
        if(linek .gt. 6 .and. first .eq. '1') exit    ! found first line of
data
      end do
      backspace (lin)
C
      jsite = 0
      do while(jsite .ne. nsite)
        read(lin,*,end=97) nrow, th, chdate, chour, dsite, jsite
C         write(lout,'(I5,1x,a10,1x,a5,1x,a4,i2)')
C     &   nrow, chdate, chour, dsite, jsite
      end do
C                                       found first line of site
      read(chdate,49) jm, jd, jy
   49 format(i2,1x, i2,1x,i4)
      jdate = jd + 100*jm + 10000*jy
C
      if(th .lt. 999.) then
       nh = 1
       sumh = th
       miss = 0
      else
        nh = 0
        sumh = 0.
        miss = 1
      end if
    1 continue
      ldate = jdate
C      if(th .gt. 9998.) th = 999.
C      tmin = max(th,999.)   ! initialize
C      tmax = min(th,-999.)
      do while (jsite .eq. nsite)
        read (lin, *) nrow, th, chdate, chour, dsite, jsite
        read(chdate,49) im, id, jy
        jdate = id + 100*im + 10000*jy
        if(jdate .eq. ldate) then
          if(th .lt. 999.) then
            nh = nh + 1
            sumH = sumh + th
          else
            miss = miss + 1
          end if
        else
          if(nh .ge. 1) then
            avrh = sumH/real(nh,4)
          else
            avrh = 9999.
          end if
          mdate = 1000000*jm + 10000*jd + jy
          write(lout,48) mdate, avrh, miss
          jm = im
          jd = id
          if(th .lt. 999.) then
            nh = 1
            sumh = th
            miss = 0
```

11/1/02 RES

File: C:\Applications\OPus2\SplitW\transpRH.for 10/11/2002, 7:32:22PM

```
          else
            miss = 1
            nh = 0
            sumh = 0
          end if
          if(jsite .eq. nsite) go to 1
        end if
      end do
 48   format(BZ,I8,8x,f8.1,6x,i2)
      return
 97   continue
      tfname = trim(fname)
      write(lout,'(" site",i2," not found in file ",a20)')nsite, tfname
      return
      end
```

RL 4/14/2008

11/28/02 RES

Kineros2 validation

Possibilities for runs which can be compared with analytic solutions:

1. Steady runon case, flat surface. use steady infiltration so that vol. balance will easily predict the asymptotic length of wetting on downstream plane. depth profile also calculable.

2 steady runon case, no infiltration. Can predict rate of advance

3. steady runon, microtopography as in Split wash. This requires a complex o.d.e. solution. DAW will work on this. Can find profile and length of advance as in 1 (above).

4. infiltrability - do case with steady rainfall and compare f(I) with 3-parameter analytic solution. [f = infiltrability, I = infil. depth]. This requires auxiliary printout of diagnostic option.

TL 4/14/2003

2/14/2003 RCS

A test Opus2 file for Split Wash
- try runon of one plane onto another, using unique capability of Opus 2 (vs. Opus)
- upper plane with soil (upper) at 200 mm
lower plane " at 300 mm.
total area 0.5 ha. 100 m upper, 20 m lower
d use 10 % slope

Plant properties trial: sparse plant with low water use efficiency, deep roots, perennial, low leaf area index:
IPER = 5 continuous growth like tropical forest, temp. & H_2O limiting
PLAI = .3
PDRYM = 2000 kg/ha
RDEP = 1000 mm
POTHT = .6 m - relatively low growing shrubs
TGBM = 2., TGOP = 22 try this - could reduce TGBM to below 0°C ?
CONVF = 3. low - crops have 30 or more
PPCV = .15 - .17 they cover small % of surface

Upper soil uses 7 (200 mm) or 8 (300 mm) numerical layers. IPER ≅ 1200 to 1300 mm.

/03 RES Output is in Splitwtest*.out in C:\applications\Opus2\splitw
seepage ~50 mm in 1992
90 1993
-.3 1994 etc.

15/5/03 RES

For climate change scenarios, the plant mix at upper Split Wash can be expected to change. If a monsoon-type climate such as in S. Arizona develops, both cool & warm season grasses can be reasonably expected to move into the area. The rainfall should increase somewhat and have both a winter and late summer wet period. DAW is developing a climate data file for this scenario.

For Opus, the plant model uses a degree-day timing parameter for perennial growth start and for length of growth period. Also, the growth is regulated by optimum and min. daily temperature. These parameters are adjusted for development of a cool-season and a ~~wet-seas~~ warm season grass. Compared w. the desert brush plant now in Opus, grasses should: a) have a more limited growth period, with somewhat higher growth efficiency. b) have no permanent stalks (COVI) cover, c) have a higher concentration of shallower roots.

Parameters for these crops are shown below. For a 10-yr. test run, the cool-season grass peaked in about April-May, and the warm-season grass peaked in Aug-Sept.

```
File: Edit2  9/18/2003, 10:09:29PM

* ---------------------------------------
*   IDCR    IPER    PLAI    DDEM    DDMX   PDRYM    POTY    RDP    PLIG    RLIG
DesBrush     5     .30     14.5   2343.   2000.    10.  1000.0   0.15    0.10        D02
* POTHT    PPCV    TGBM    TGOP   CONVF   DEACT    COVI  DMINIT    PST     HPC
   .6    0.17    -1.0    22.0    1.0    0.01    0.1     0.0    0.      0.          D03
*  CONY    CFXN     PNO     PNF     DKC   PNRAT
  0.018     0.0    0.02   0.012   3.00    0.25                                     D04
*   IDCR    IPER    PLAI    DDEM    DDMX   PDRYM    POTY    RDP    PLIG    RLIG
CoolGrss     3    1.50     80.   1100.   2000.    10.   300.0   0.15    0.10        D02
* POTHT    PPCV    TGBM    TGOP   CONVF   DEACT    COVI  DMINIT    PST     HPC
   .3    0.30     0.5    16.0    8.0    0.02    0.0     0.0    0.      0.           D03
*  CONY    CFXN     PNO     PNF     DKC   PNRAT
  0.018     0.0    0.02   0.012   3.00    0.25                                     D04
*   IDCR    IPER    PLAI    DDEM    DDMX   PDRYM    POTY    RDP    PLIG    RLIG
WarmGrss     3    1.50    550.   1400.   2000.    10.   300.0   0.15    0.10        D02
* POTHT    PPCV    TGBM    TGOP   CONVF   DEACT    COVI  DMINIT    PST     HPC
   .3    0.30     8.0    21.0    8.0    0.02    0.0     0.0    0.      0.           D03
*  CONY    CFXN     PNO     PNF     DKC   PNRAT
  0.018     0.0    0.02   0.012   3.00    0.25                                     D04
```

Aug 22, '03 RES

Further on Validation of KINEROS2 (continued from p 48)

Validation type 1: Runon with steady f. This case has a theoretical solution discussed by Cunge & Woolhiser:

[Cunge, J.A., and D.A. Woolhiser, "Irrigation Systems", Ch. 13 in: Unsteady Flow in Open Channels (Vol. 2), pp. 522-533+, Water Resources Publications, Ft. Collins, CO, 1975

the time of arrival of a kinematic shock at a point X down the plane, with a suddenly started steady upslope inflow Q_0 and $X=0$ and $t=0$ is:

$$t_* = m\left(1-[1-X_*]^{\frac{1}{m}}\right)$$

in which $t_* = t/T_0$

$T_0 = H_0/i$

$H_0 = \left(\frac{Q_0}{\alpha}\right)^{1/m}$ $\qquad Q_0$ in $m^3/sec/m$

α = ~~roughess coef.~~ = $\frac{\sqrt{s}}{n}$

s = uniform slope

n = manning roughness

i = steady infil. rate, m/sec

$X_* = X/X_0$

$X_0 = Q_0/i$

for a test case, try a plane of 100 m., $s = .04$, $n = .05$ and $Q_0 = 0.0002\ m^2/s$, $i = 5\ mm/h = 1.3889\times10^{-6}\ m/s$

in K2, $m = 5/3$

RES Aug. 22, '03

thus $\alpha = .2/.05 = 4$

$$H_o = \left(\frac{.0002}{4}\right)^{.6} = 0.002627 \text{ m}$$

$$T_o = \frac{.002627}{1.3889 \times 10^{-6}} = 31.52 \text{ minutes}$$

$$X_o = \frac{.0002}{1.3889 \times 10^{-6}} = 144 \text{ m.}$$

so $X_* = 100/144 = 0.6944$

and $t_* = \frac{5}{3}(1 - .30556^{.6}) = .8484$

arrival @ t(100) = 26.74 minutes

K2 includes numerical dispersion so this should be an average. Results are plotted as shown at top of p.53

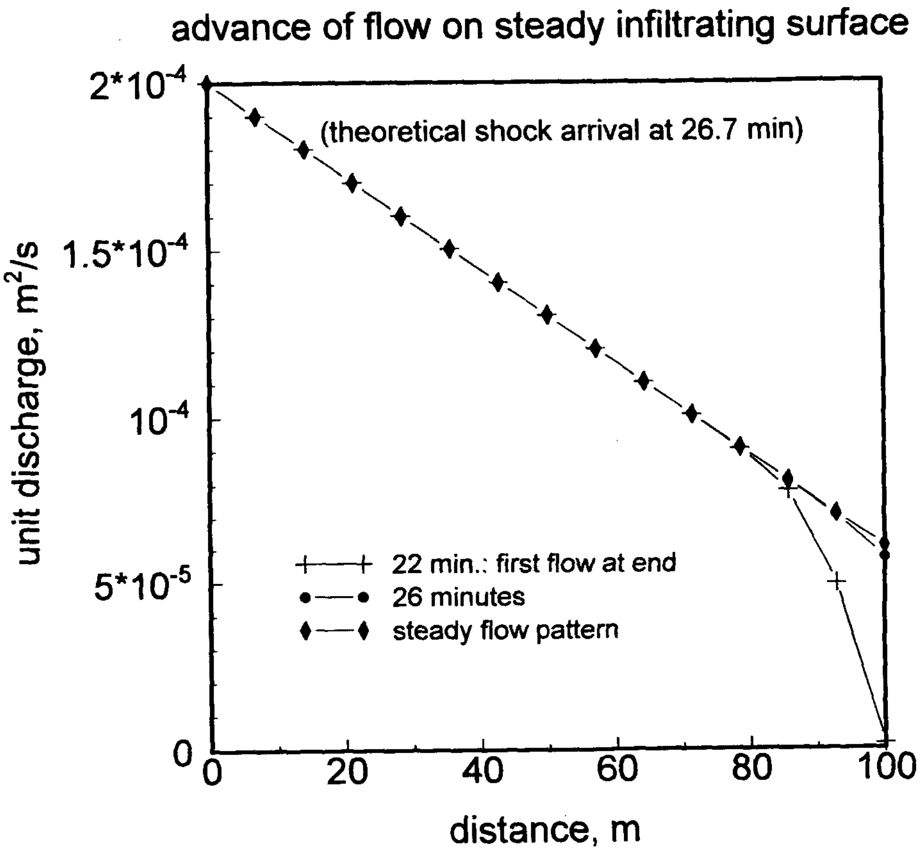
Validation type 2 (p.48) works similarly, with expected dispersion of front rather than a sharp shock front. The kinematic theory is simple, with velocity = Q_o/H_o, and advance distance = velocity x time.

Validation type 3 (p.48) is more complex; D.A. Woolhiser will develop an analytic solution. In this case, flow characteristics and geometry changes with flow depth, and even with steady i, loss rate changes with wetted width down the surface.

The K2 results for one case are graphed on P. 54.

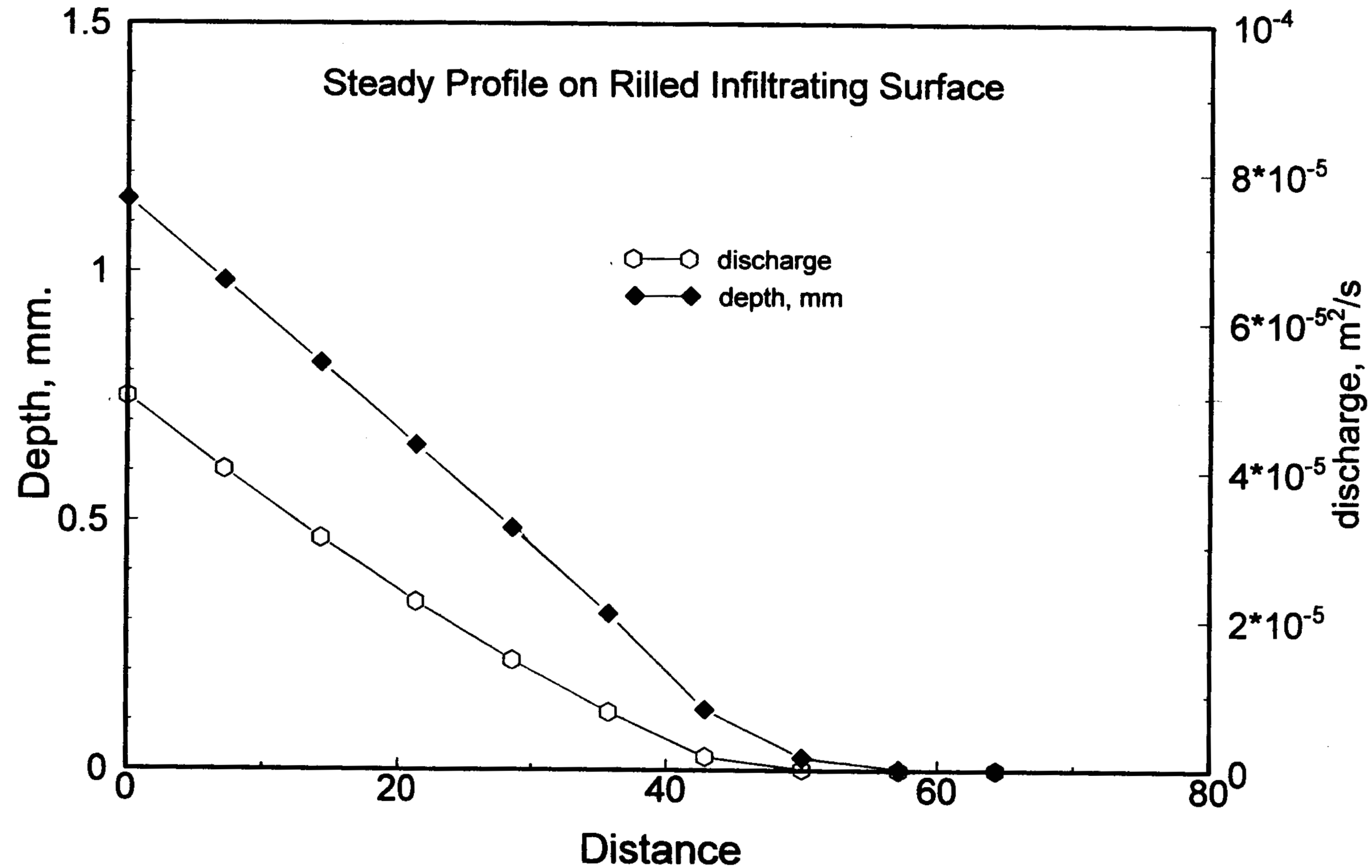
This case has rills (trapezoidal), 10 cm bottom width, 50 mm deep, and 1m wide at top. S = .05, n = .151, Q_o = .0001 m^2/s

RES Aug. 22, '03



RL 4/14/2008

Rss Aug 22, '03



Steady Profile on Rilled Infiltrating Surface
discharge
depth, mm
Depth, mm.
discharge, m^2/s
Distance
0
20
40
60
80
0.5
1
1.5
2*10^-5
4*10^-5
6*10^-5
8*10^-5
10^-4

RES: Aug. 22, '03

Validation test 4 is for the infiltration model in K2, which is the Parlange 3-parameter model:

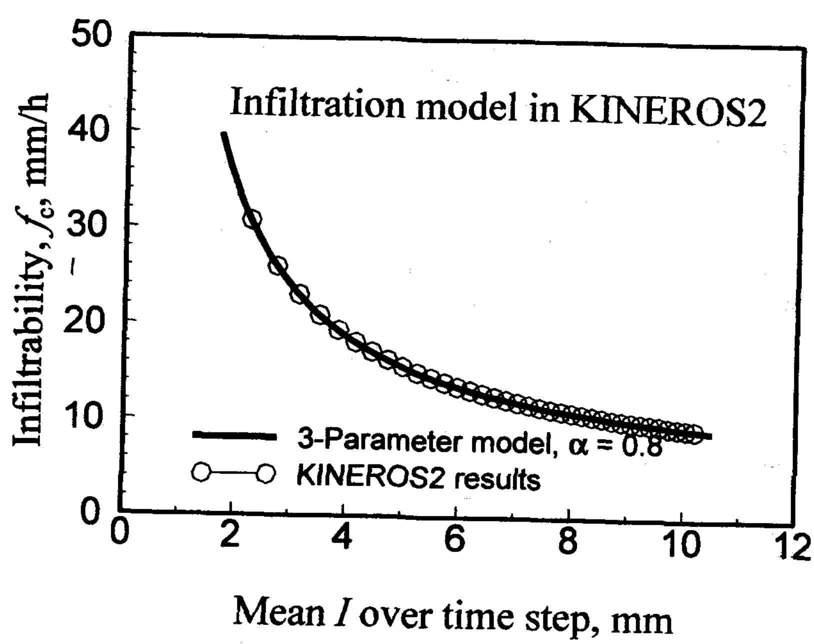
$$f_* = 1 + \frac{\gamma}{\exp(\gamma I_*) - 1}$$

with $\gamma = 0.8$, $I_* = I/G\Delta\theta$

$$f_* = f/K_s$$

in which G is soil capillary length scale, mm
$\Delta\theta$ is soil saturation deficit
K_s is " saturated conductivity, mm/hr.

Validation is straightforward, using aux. diagnostic printout giving f and I.



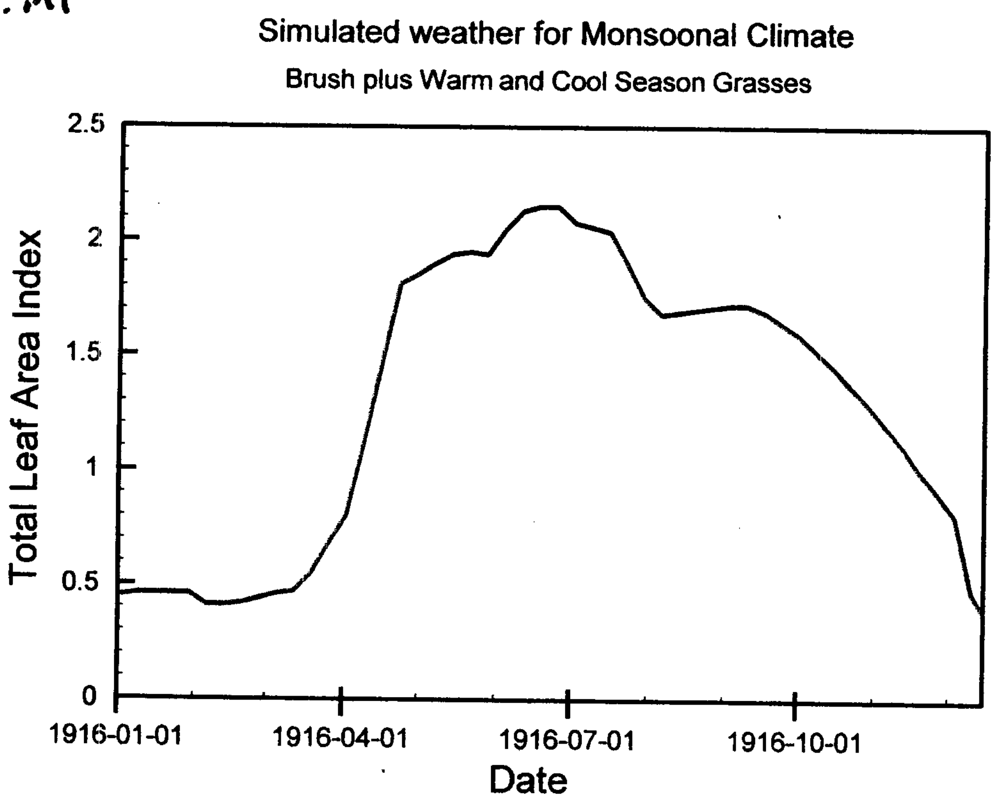
RGL Feb. [illegible] continued from p. 50

Trial of monsoon climate -

DAW has prepared 50 years of simulated monsoon climate, and we want to compare a year or 2 of data using KINEROS2 on selected events, and using OPUS2 on the complete period.

2/10/04 The plant mix includes both cool-season and warm season ~~pl~~ ~~reg~~ grasses and a sparse brush (see p. 50). The total Leaf area index for a year cycle is shown in fig. M1, below. The initial increase comes from spring grasses, which are replaced by warm season grasses in June and senesce thru the fall. Brush is active throughout the year, with deeper root extraction.

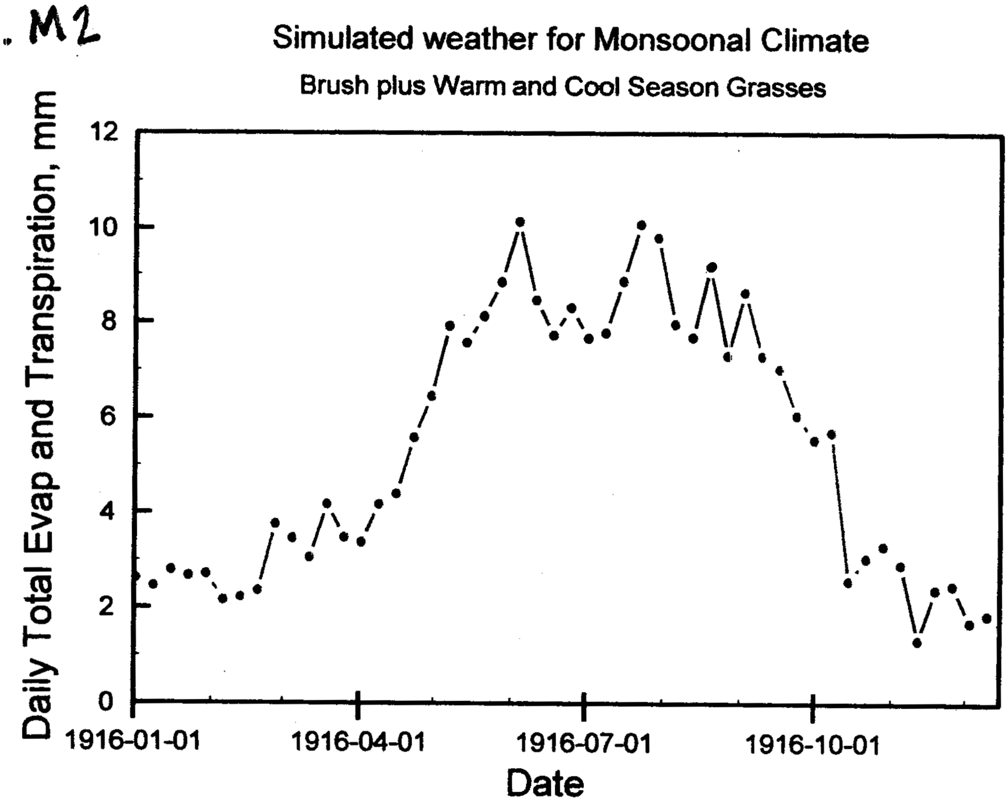
fig. M1



RES Feb. 10, '04

Fig. M2 illustrates the daily total ET which reflects the LAI pattern, as expected. These 2 graphs illustrate well how winter and early spring rains can be very effective in generating deep seepage, since plants are unable to utilize significant water during this period.

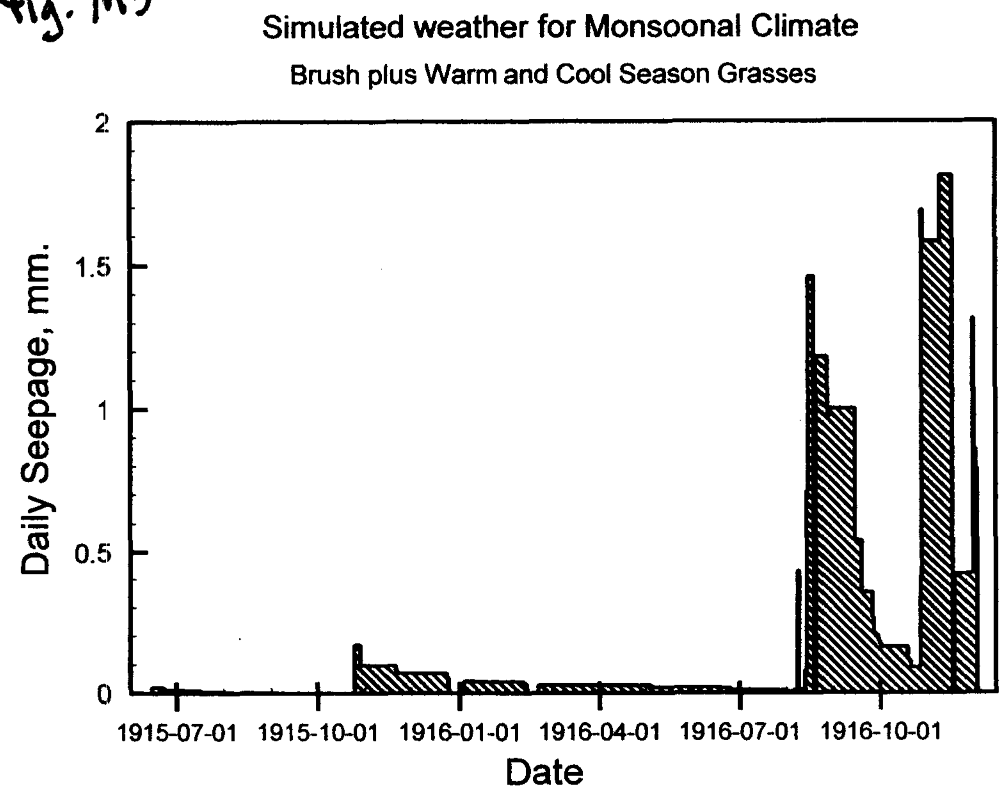
fig. M2



There are always variations in the water use pattern due to variations in temperature, clouds (net radiation) and rainfall. Figure M3 illustrates that rainfall has an overwhelming influence on seepage, interacting with the plant ET potential for the period of the year when rain occurs. For this year (or 2) the rain coming in the fall is far greater than the plant's ability to transpire.

RGS Feb. 10, '04

fig. M3



The above data comes from 2 runs, 1 with lower K_s to match the K2 value for summer, and a 2nd with $K_2 - K_s$ to match winter:

K_s used	output file	hydrology outfile
11.25 mm/h	~~split~~ spmo4yr16L.out (RGS)	splitw16L.hyd
22.5	spmon4yr16.out	splitw16.hyd

RES Apr 22, '04 Prepare data for simulating 10 years of Monsoon climate.

a. Dave Woolhiser has simulated an altered climate for split wash based on the assumption of a 'monsoon' climate such as occurs in southern AZ, where considerable data is available through the USDA ARS Walnut Gulch raingauge records

DAW simulated 50 years for Split Wash, and then disaggregated those storms which were likely to produce runoff, to produce a breakpoint rainfall record. These storms were used in KINEROS2 (K2) and the results analyzed statistically. K2 requires assumptions on initial soil water, and cannot simulated the amount of soil water that is used by plants. In order to see if the K2 results are reasonable, 10 years of the monsoonal rain (+Temperature and Radiation) record were used in Opus simulation

The days on which storms were simulated to occur are given below. These were all disaggregated to simulate a rain pattern, and separate *.pre files for each furnished by DAW Of the 50 year record, years 15 thru 24 were chosen (20th century arbitrarily used)

Year	Storm dates
1915	JUN14, JUN23, JUL14, JUL19, AUG01, SEP03
1916	JUL07, JUL08, AUG08, AUG19, ~~Feb~~ Pes OCT19, NOV17
1917	JUL29, AUG04, AUG16, AUG24, SEP30
1918	AUG08, AUG22, NOV03, NOV18
1919	JUL09, JUL30, AUG01, AUG04, AUG10, SEP24
1920	JUL25, AUG18, SEP15
1921	APR28, MAY25, JUL05, JUL07, AUG03, AUG05, AUG17, AUG28, SEP16
1922	JUL17, AUG21
1923	AUG31, NOV12
1924	AUG30[a], AUG30[b], SEP21, SEP30, OCT03, NOV18, NOV21, DEC24

underlined storms occur on days where more than one rain event occurs, according to the disaggregation simulation.

RES Apr 23, '04

Days with multiple storms will be treated below. K2 precip files are coordinated with the overall simulated record by taking cumulative depths from beginning of each calendar year for Opus. Opus *.met input file is first created from the DAW output with a simple transform program that makes each daily amount composed of 2 breakpoints written in Opus *.met format Then the K2 precip file for each storm day is transferred to Opus format, using the cumulative depth correct for the beginning of each storm. This is 5 (depth, time) pairs per line of file. Disaggregation usually produces 21 points or pairs. The rewritten storm record is then edited into the simple *.met file ("monsoon4.met) produced as described above. This gives a true breakpoint pattern only for K2-storm days.

For days with multiple storms, insignificant 2nd or 3rd pulses on that day must be located before or after the disaggregated pulse. Data from DAW for those days then gives the correct starting cumulative depth for each part.

RES 4/27/04 Calculated with the aid of a spreadsheet, we have:

Storm Day	Pulse" depth mm	[Starting] Cum. Depth mm	Length (Min)	Start Time assumed (min)
JUN14_15	16.61	50.08	90	840
	0.69	66.69	20	1130
		67.38		
JUN23_15	14.19	82.59	148	780
	0.27	96.78	20 }	1200
	0.91	97.05	30 }	
		97.96		
AUG19_16	3.91	225.78	60	720
	13.77	229.69	52	970
		243.46		

RES 4/27/04

day	part depth	cum depth	Length	start time of day (min)
OCT19_16	9.03	279.33	120	540
	13.51	288.36	204	850
		301.87		
NOV17_16	6.33	348.15	120	360
	20.19	354.48	720	710
		374.67		
AUG04_17	12.68	104.31	53	780
	10.20	116.99	120	1020
		127.19		
AUG16_17	5.71	141.66	60	520
	13.70	147.37	86	780
		161.07		
AUG10_19	10.93	185.05	120	280
	15.09	195.98	166	900
	2.84	211.07	60	1250
		213.91		
JUL25_20	14.97	37.49	136	780
	0.12	52.46	30	1120
		52.58		
SEP15_20	0.62	90.9	20	600
	35.89	91.52	133	840
	7.04	127.41	60	1200
		134.45		
MAY25_21	13.05	35.90	46	780
	4.93	48.95	60	1010
		53.88		

RSS 4/27/04

day	part depth	cum depth	Length min	start time of day (min)
JUL07_21	51.27	112.02	480	720
	1.79	163.29	30	1390
		165.08		
AUG03_21	8.18	217.29	60	480
	13.77	225.47	161	720
	2.38	239.24	40 }	1070
	1.32	241.62	30 }	
	0.47	242.94	15 }	
		243.41		
AUG05_21	18.28	246.27	156	720
	6.17	264.55	60 }	1050
	0.31	270.72	20 }	
		271.03		
AUG 28_21	14.75	321.62	150	720
	8.07	336.37	120	1060
		344.44		
AUG21_22	21.32	185.87	217	780
	0.59	207.19	20	1180
		207.78		
AUG31_23	14.61	167.93	128	780
	0.61	182.54	20	1100
		183.15		
NOV12_23	14.68	192.44	713	420
	4.53	207.12	60	1300
		211.65		

RES 4/27/04

day	part depth (mm)	cum depth	Length (min)	start time of day (min)
AUG30_24	20.48	91.99	91	750
	17.52	112.47	195	1080
		129.99		
SEP30_24	14.36	160.97	42	780
	7.55	183.33	120	920
		190.88		
OCT03_24	27.85	190.88	720	360
	3.16	218.73	120	1300
		221.89		
NOV21_24	31.50	281.03	720	330
	7.82	312.53	180 }	1240
	0.63	320.35	20 }	
		320.98		

Notes
1. underlined pulse is the disaggregated storm
2. start times are assumed. Early afternoon for main part during summer monsoon. Separations of 180 min make separate storms for Opus, but 10 minutes used in DAW generating (disagg.) program. Thus brackets indicate where small following pulses are described by one record for Opus *.met record

Lengths of disagregated storms taken from DAW data. Other lengths assumed to give flux values less than lower K_{sat}

RES 7/18/04 simulated (disaggregated) storms on p. 60-63 along with all other rains are input into Opus w. parameter files and simulated Tmin, Tmax and radiation to look at root zone & runoff response. Computer folder C:/applications/opus2/splitw contain the files: RES

Parameter files (K2 element 23 characteristics)

K_s condition	upper soil depth		
	120 mm	300 mm	500 mm
Low K_s	SplW12d10L.par	Splw30d10L.par	Splw50d10L.par
High K_s	SplW12d10H.par	Splw30d10H.par	Splw50d10H.par

Low k_s is 11.25 mm/hr
high is 22.25 " following the estimate of DAW
3 soil depths should give us an estimate of effect of overlying soil on hydrology. grass roots in each case are assumed to reach only to top of fractured tuff.
T&R data: Monsoon4.act preap data: Monsoon4.met

Output Files:

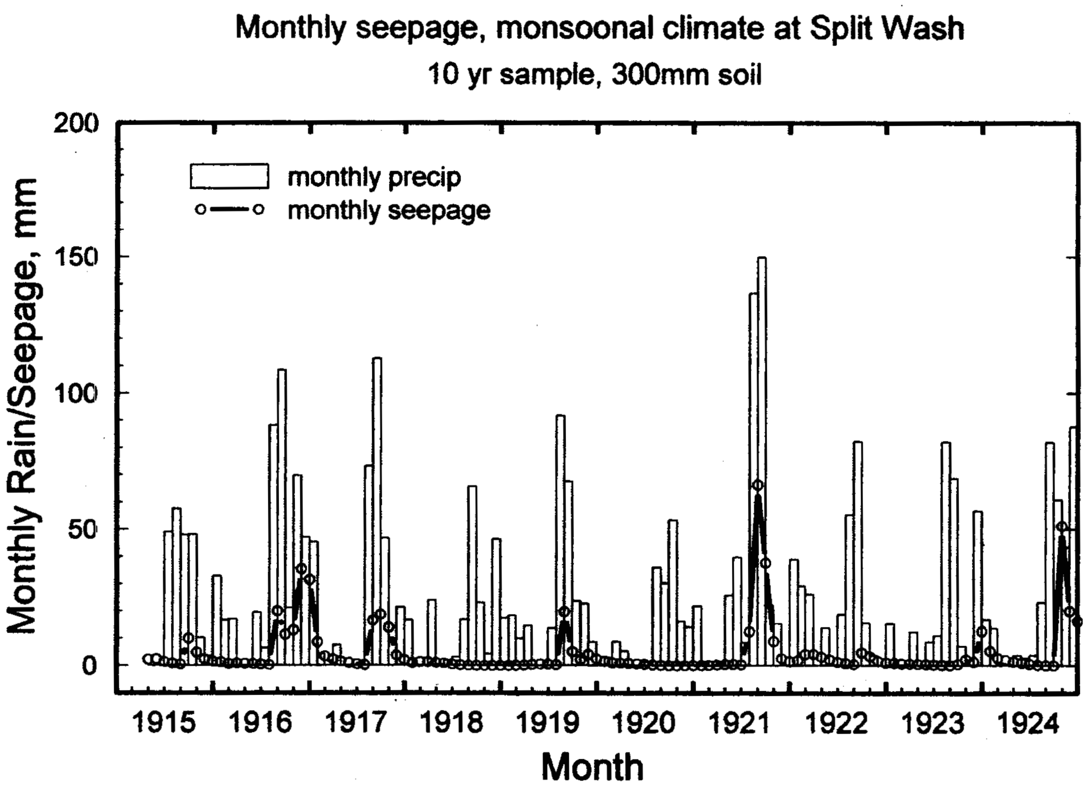
K_s used	Upper Soil Depth 120 mm	300 mm	500 mm
Low	Spl12d10yL.out	RES 7/18/04 ~~Spl12~~ Spl30d10yL.out	Spl50d10yL.out
High	Spl12d10yH.out	Spl30d10yH.out	Spl50d10yH.out

Subsurface output every week, with same file names but with extension *.sub rather than *.out

7/21/04 Test run data with initial θ (=θi) of 0.11 for 3+ years, indicates this is too low, since no seepage from any storm until after a 2 yr period. Changed to 0.21=θi, seems to start out well.

Some results plotted:

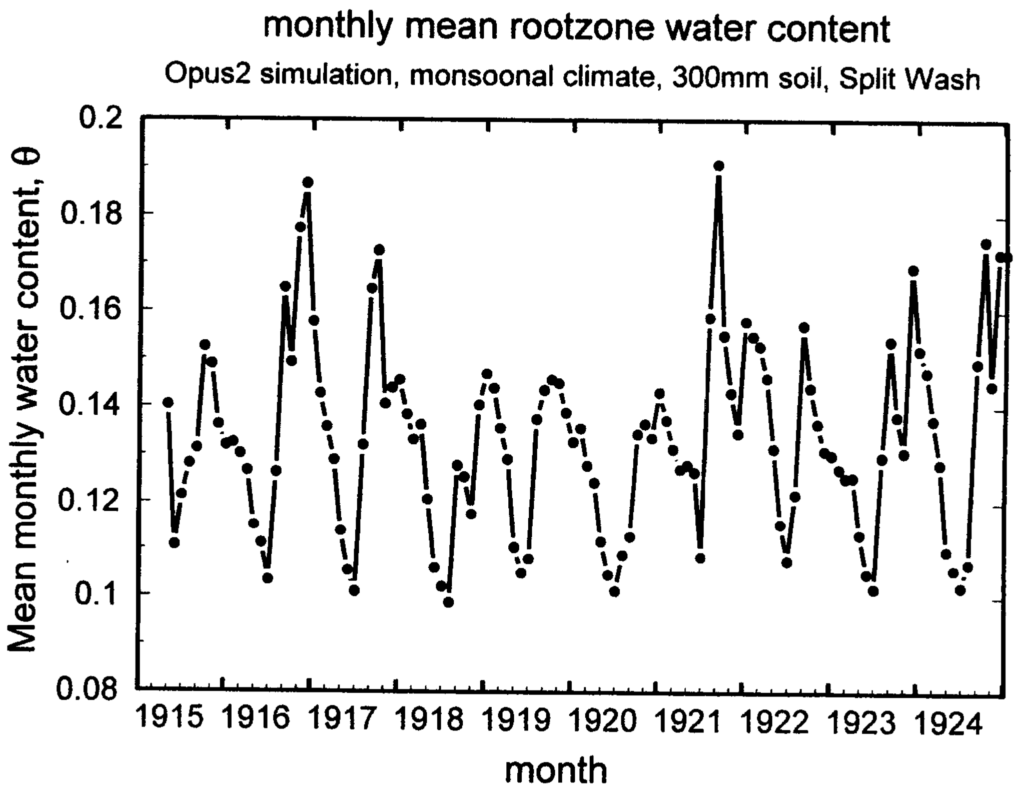
Fig. M4



C:\Applications\OPus2\SplitW\Spl30d10yLseep.draw

RES 7/21/04

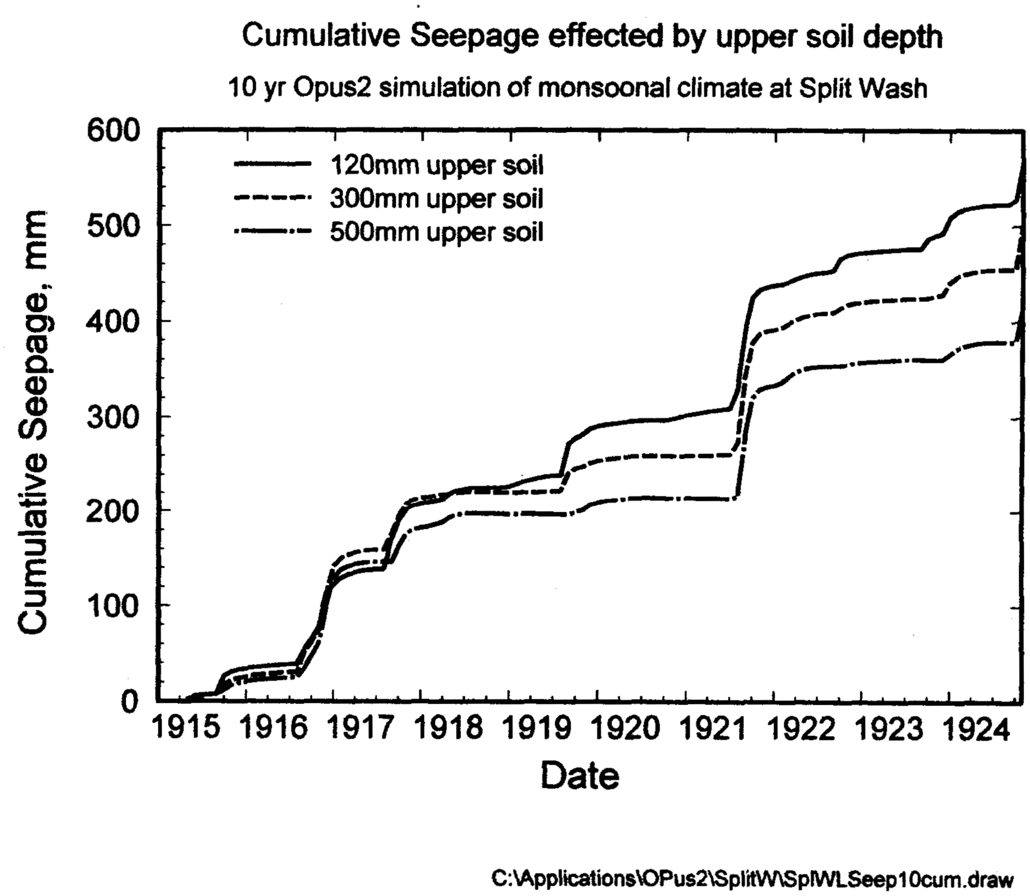
Figure M5



C:\Applications\OPus2\SplitW\Spl30d10yLtheta.draw

RSS 7/21/04

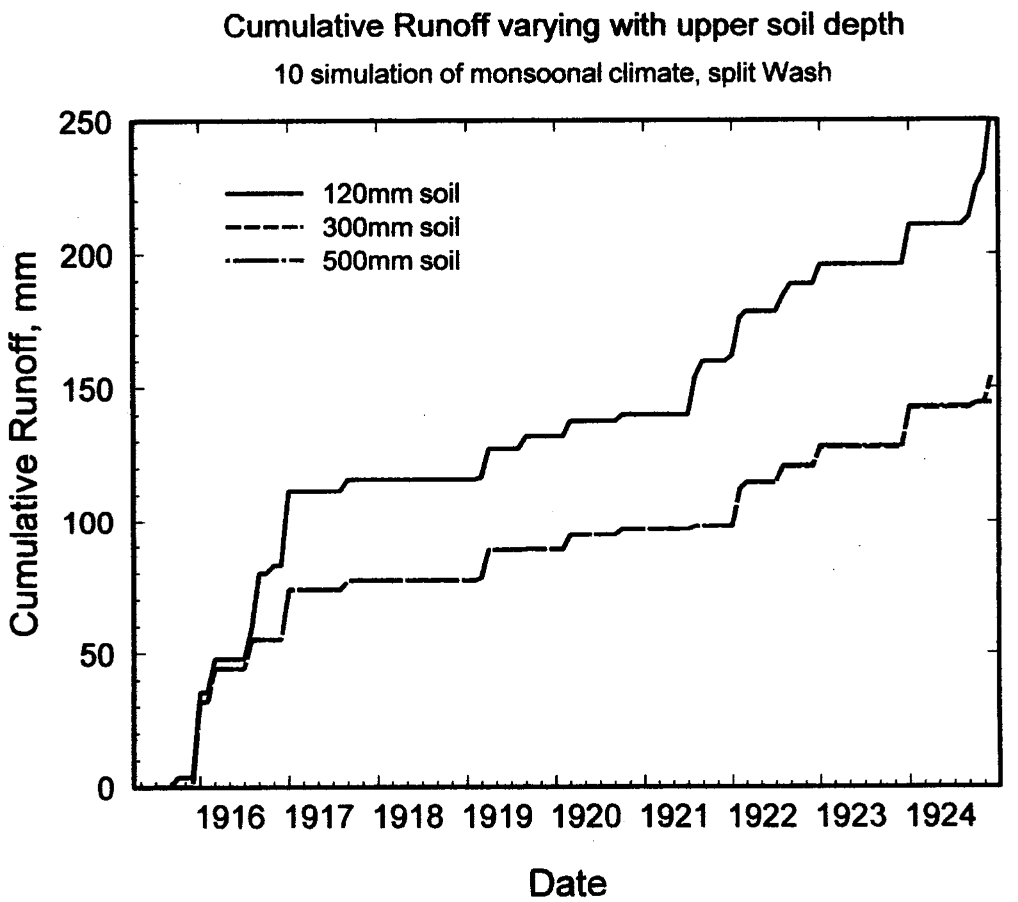
Figure M6



Note: both low and high K_{sat} assumptions for upper soil are simulated, and these plots are for low Ksat. Other Ksat has generally similar features.

RES 7/21/04

Figure M7



C:\Applications\OPus2\SplitW\SplWLROcum10y.draw

RES 7/21/04

Figure M4, p.65, shows the distribution of rainfall for this monsoon climate, with many large storms / wet months occurring in the summer, or some in fall. Some winter precip. in addition.

Seepage occurs of significant amounts only in conjunction with wettest periods, and persists for only 2-3 months. However, annual averages of 40-50 mm/yr occur. If the assumed plant characteristics are in error, so will be the seepage. Seepage is slightly negative for dry periods — seep is defined as the flux downward below the lowest root depth — at about 1100 mm.

Figure M5 shows the simulated mean monthly net rootzone water content, by volume. The variance for weekly and daily water contents would be larger. The net value includes both the upper soil, with max θ of 0.323, and fractured tuff (soil-filled fractures) with max θ of 0.18. Thus the net is a function of upper soil depth [— other cases not plotted here.]

The annual precip. cycle is evidently reflected in the θ pattern. Summer drying (early summer) is noted.

Figure M6 contrasts seepage patterns of 3 soil depths using cumulative seepage. Note that grasses (cool & warm season) have roots assumed down to the bottom of the upper soil, while the brush has a longer period of water use but with roots to 1000 mm in each case. As could be expected seepage is usually concentrated in the fall, as grass water use declines and rain may continue.

Figure M7 shows that the effect on runoff (of soil depth) is lost once the upper soil reaches 300 mm or more. The differences are seen in both summer and winter storms, but

RSS 7/21/04

in every event. Clearly more upper soil storage provides opportunity for plants to use water and reduce seepage dramatically for many storms

Higher Ksat results similar – most different for runoff, as expected.

RL 4/14/2008

9/06/04 RES

Existing recorded data, prepared for Opus2 use (see p. 33-47) is also treated in a 10 year simulation for present day climate. The last 2 1/2 years of tipping bucket data, from DAW, are from "INGRES REPORT", copyright "Computer Associates Int'l". DAW indicates site 18 is closest to Split Wash, and site 8 is best for period late 94 to 95 when there is a data hiatus.

This raw data is converted to Opus format with the F95 program on following pages, developed and debugged today.

With the precip data complete, the files used are

recorded Tmax, Tmin, humidity : splitwsh.act
meteorologic – breakpoint record : gs87-96.pre

Parameter files:

	soil depths 120 mm	300 mm	500 mm
Low K_s	SplW12d96L.par	SplW30d96L.par	SplW50d96L.Par
High K_s	SplW12d96H.par	SplW30d96H.par	SplW50d96H.Par

output files

Low K_s	Spl12d96L.out	Spl30d96L.OUT	Spl50d96L.out
High K_s	Spl12d96H.out	Spl30d96H.OUT	Spl50d96H.OUT

subsurface out same as *.par but with *.sub extension
hydrology outfile same as main outfile, with *.hyd "

9/06/04 RSS

File: C:\Applications\OPus2\SplitW\TippingData\tipover.for 9/7/2004, 2:35:42PM

```
C   program for converting tipping bucket data to Opus format
      program tipover
C
      real,dimension(900) :: cumt, cumrd
      integer :: im,id, iyr, nhr, nmin, nsec, nelaps, nzr = 0
      character(LEN=60) :: tipfile, outfile
      write(*,*) '   Transform tip data to cum. pairs:'
      write(*,101)
  101 format("  Enter filename of tip data:  ")
      read(*,'(a)') tipfile
      open(3,file=tipfile,status='old',err=991)
C
      write(*,102)
  102 format("  Enter filename for output:  ")
      read(*,'(a)') outfile
      open(7,file=outfile,status='unknown',err=992)
  103 format("  Enter averaging relative diff (0 to ~0.05):  ")
      call skipl(3)
C
      nwr = nzr
      nelaps = 0
      bkptot = 0.
      stmtot = 0.
      add = 0.
      rate = 0.
      do while (.true.)
        onlaps = nelaps
        lyr = iyr
        ratel = rate
        read(3,105,end=99) rate, im,id,iyr, nhr,nmin,nsec, nelaps
C        write(7,501) nhr,nmin,nsec
C  501 format(5x,i2,':',i2,':',i2)
C  nelaps is elapsed seconds since last tip.
C  rate is in mm/h
        if(nhr .eq. -9) cycle
        if(nelaps .eq. 0 .or. nelaps .gt. 18000) then
          if(ibp .ge. 2) then   ! write record
            write(7,204)iyr,jm,jd,ibp,nwr,stmtot
            write(7,205)(cumrd(i),cumt(i),i=1,ibp)
            bkptot =  cumrd(ibp)
            ibp = 0
          end if
          nwr = nzr
          if(iyr .gt. lyr) bkptot = 0.    !start cumulating at beg. of yr.
          btime = 0.
          add = 0.
          stmtot = 0.
          if(nelaps .eq. 0) cycle
        end if
        ptime = btime
        btime = ftime(nhr,nmin,nsec) + add
        if(btime .lt. ptime) then
          add = add + 1440.
          btime = btime + 1440.
          nwr = nwr + 1
        end if
        elapm = real(nelaps,4)/60.   ! elapsed time in min.
        if(nelaps .gt. 18000) then   ! dummy up a start time and rate
          elapm = 120.   ! = 2 hrs (in minutes) arbitrary
          deld = rate*2.   ! added last tip
          if(deld .le. .08) then
             deld = 0.105   ! small rates are reported as 0. and must be revised
             rate = deld/2.
          end if
          ot = btime - elapm
          jd = id
          jm = im
```

7/06/04 RES

```
          if(ot .lt. 0.) then    ! start prev. day
            jd = jd - 1
            add = 1440.
            ot = add + ot
            nwr = nwr + 1
            btime = btime + add
          end if
          ibp = 1  ! start a bkpt record
          cumt(1) = ot
          cumrd(1) = bkptot
c          stmtot = deld

        end if
c
        rmx = max(rate,ratel)
        if(rate .gt. 0. .and. ratel .gt. 0.) then
          dlrate = abs(rate - ratel)
          rdlrate = dlrate/rmx
          if(rdlrate .lt. 0.1 .or. dlrate .lt. 0.1) then      !.and. rmx .gt. 1.0
            cumrd(ibp) = cumrd(ibp) + rate/60.*elapm
            cumt(ibp) = btime   ! only in case of large interval next
            cycle         ! don't increment ibp
          end if
        end if
        ibn = ibp+1
        delr = rate/60.*elapm
        stmtot = stmtot + delr
        cumrd(ibn) = cumrd(ibp) + delr
        cumt(ibn) = btime
        ibp = ibn
c        write(7,203) nelaps, apcap
      end do
   99 continue
      stop ' end of input file found'
  991 continue
      stop ' unable to open input tipfile '
  992 continue
      stop ' unable to open Out file '
  993 continue
  105 format(t11,f6.0,t54,i2,1x,i2,1x,i4,t71,3(i2,1x),t88,i7)
  202 format(4x,"interval   apparent"/
     &     "         sec        vol(mm)")
  203 format(t5,I6,t20,f6.4)
  204 format(t5,I4,t13,2I2,2I8,T36,f5.2)      ! bkpt rain header line
  205 format(5(f8.2,f8.1))                    ! bkpt rain pair line
      end
C------------------------------------------------------------------------
      subroutine skipl(nu)  ! gets past heading info in file nu
      integer :: nu
      character(LEN=1) :: cha
      do while (.true.)
        read(nu,'(a)') cha
        if(cha .ne. '*') then
          backspace nu
          return
        end if
      end do
      stop ' end of record in skipl '
      end
C------------------------------------------------------------------------
      function ftime(nh,nm,ns)
      integer :: nh,nm,ns
      real :: ftime
      ftime = real(nh,4)*60. + real(nm,4) + real(ns/60.,4)
      return
      end
```

9/9/04 RES

The 10 year record from near Split Wash is dominated (in runoff) by 3 long wet sequences, plus 2 very dry seasons. The late winter 94-95 is especially wet. The tipping bucket record is long enough that OPUS2 dimensions for rain storage are increased to size 500.

p.75 Figure A1 shows the pattern of monthly precip. This contrasts with figure M4, with big storms only in winter, and other input in scattered sequences. Again, small negative seepage is simulated for dry periods, when roots create an upward head gradient and some water moves up from below.

p.76 Figure A2 demonstrates the dominance of late winter wet periods in soil moisture response, and the continued drying through years (dry) 1988 thru 1990

p.77 Fig. A3 shows an interesting process interaction, since the seepage for 300 mm is less than either 500 or 120 mm soil depth. This is believed to be due to a non-linear combination of the effects of net storage and net (sat.) conductivity for the root zone. Since seepage is measured at 1000 mm, the profile is made up of an upper part with higher storage and higher K_s, and a lower part of low storage and low K_s. Higher storage reduces seepage and higher K_s increases it, but the combination in different proportions is not a linear function. ~~(RES 9/9/04)~~ We also expect a role to be played by the relative ~~values of~~ patterns and amounts of plant water requirements, soil storages, and rain input sequences.

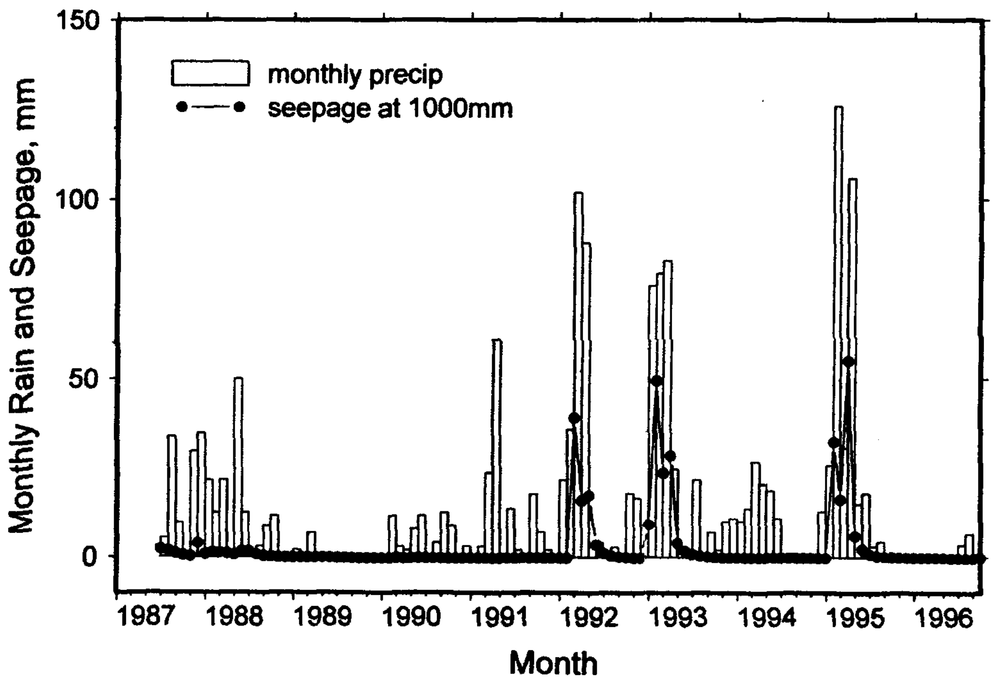
Fig. A4 is a good illustration of the saturation-runoff mechanism affected by surface soil depth, and the dominance of 1 or 2 storms on expected runoff during the period of record. (page 78)

9/7/04 RES

Fig. A1

Opus2 simulated seepage, Split Wash

10 yrs recorded data, 300mm soil w. desert brush

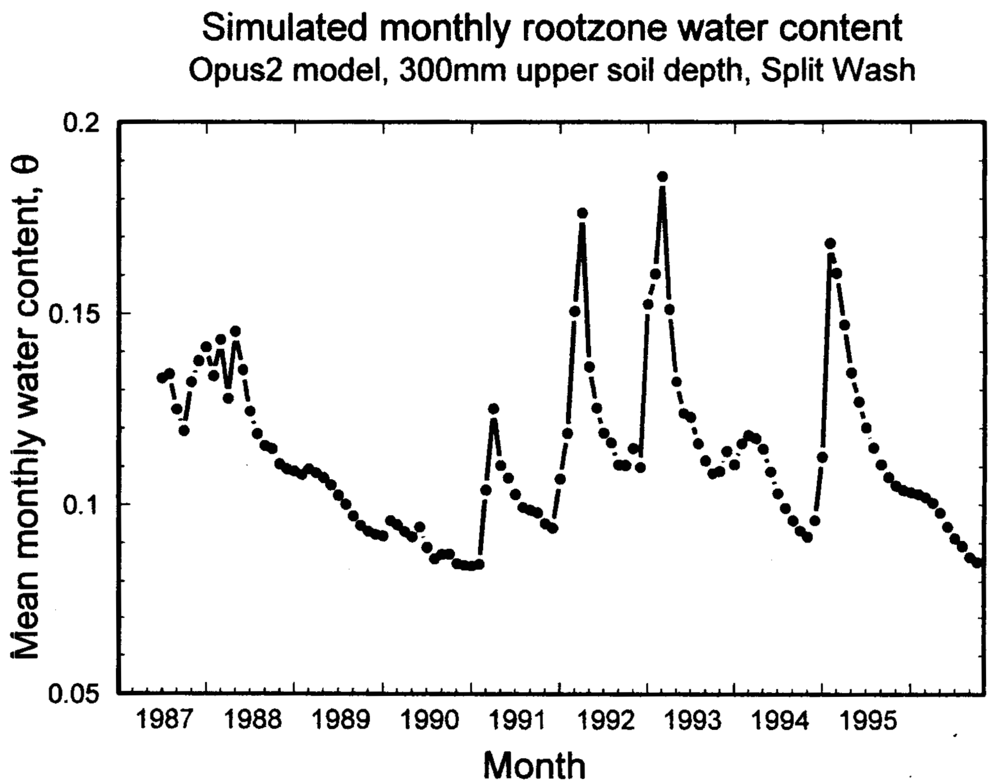


C:\Applications\OPus2\SplitW\SplW30d96Lsp.draw

[redacted]

9/07/04 RSS

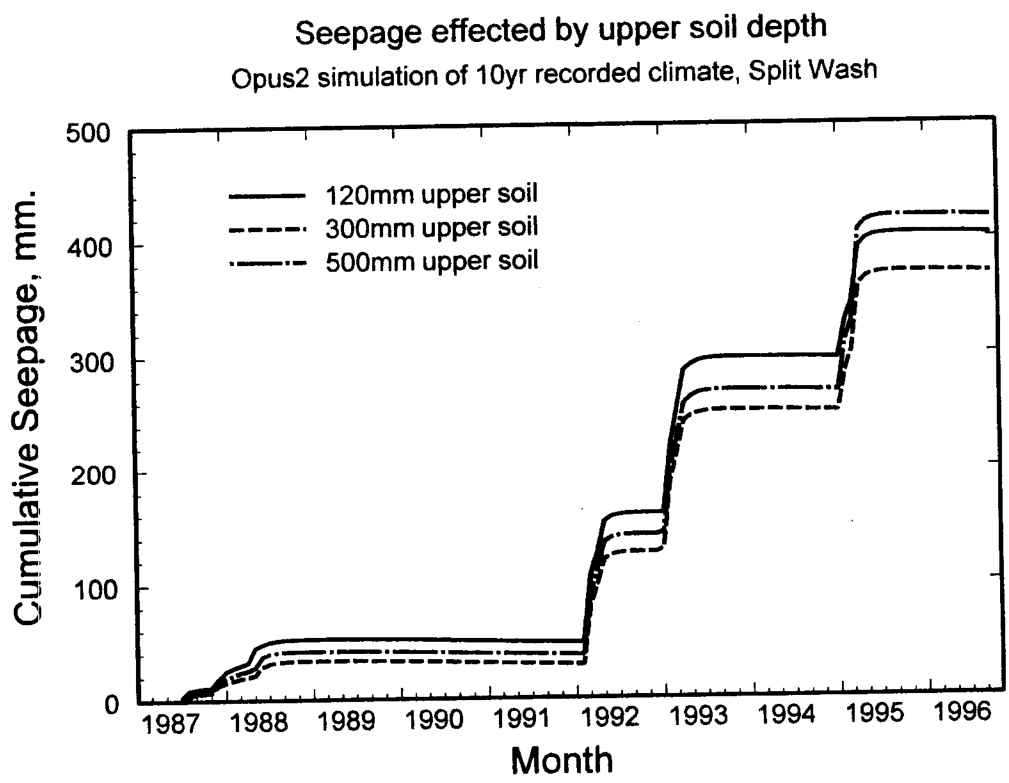
fig. A2



C:\Applications\OPus2\SplitW\SplW30d96LSW.draw

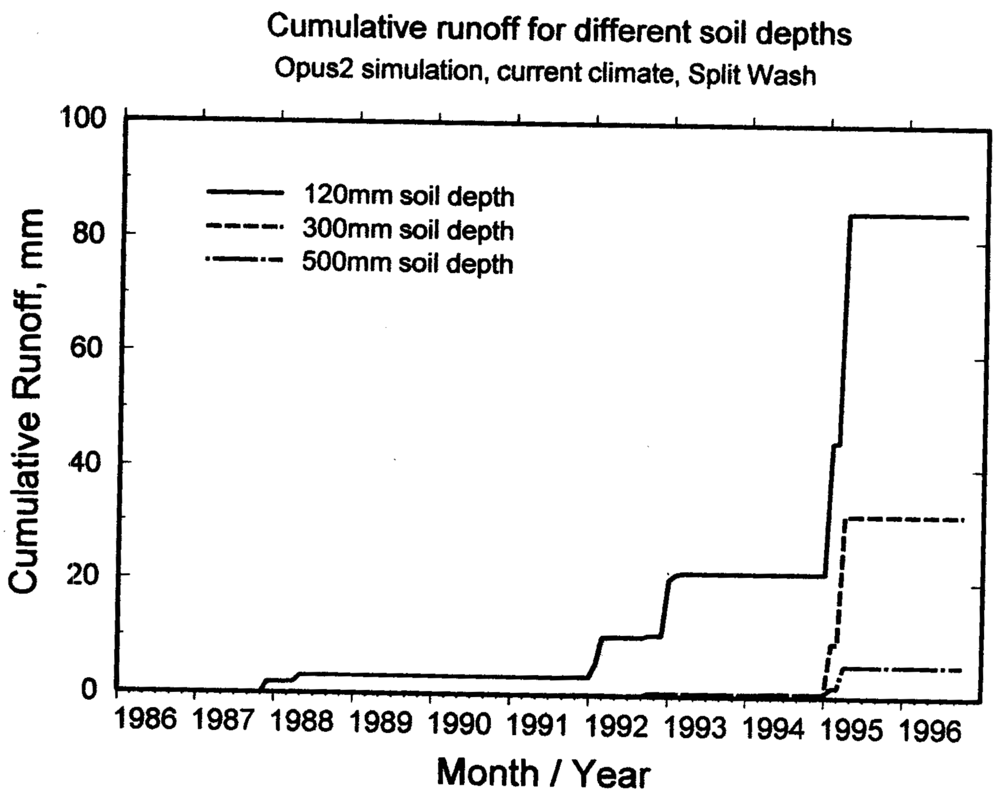
9/07/04 RSS

Fig. A3



C:\Applications\OPus2\SplitW\SplWAd96Lseep.draw

9/9/04 RSS



C:\Applications\OPus2\SplitW\SplWAd96LRO.draw

RES 9/16/04 After discussions with DAW, we decided to redo the simulations of Opus on Monsoonal climate record, with the following simplifications to better match the K2 results:

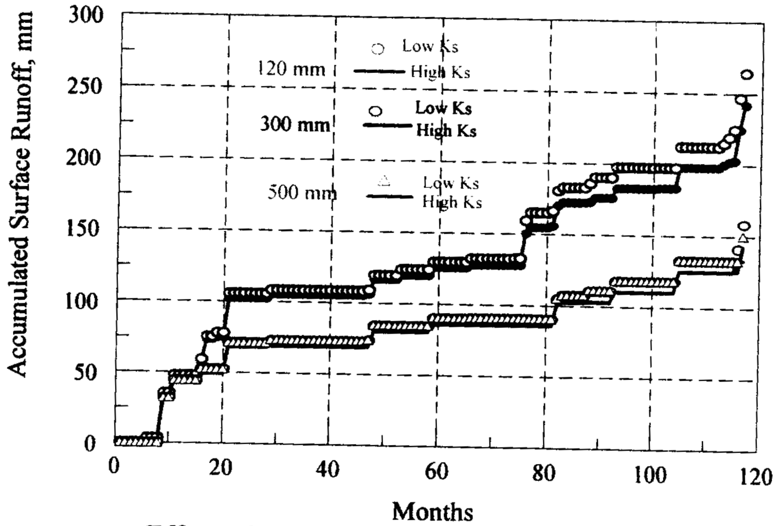
1. do not allow Opus to modify surface K_s due to accumulated rain energy.
2. make the low K_s series such that only the top 2 cm. of soil has a low K_s, and the remaining profile matches the high K_s (22.25 mm/h) (surface soil)

These runs use revised parameter files – same as shown on p. 64 except an "n" at the end. i.e., SplW12d10L.par becomes SplW12d10Ln.par. etc.
naming similar for output files as well.

Results are rather similar to previous, in general. 2 examples shown on next page. Some small differences in Horton runoff cases.

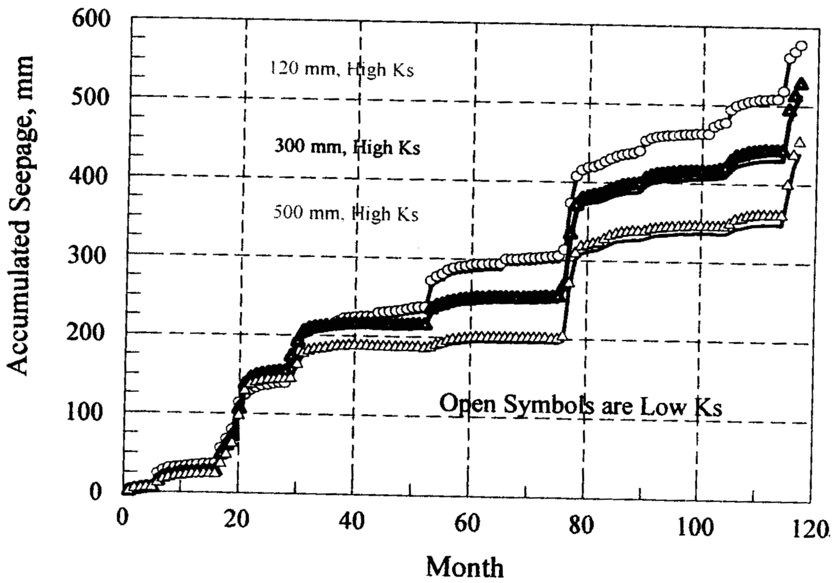
RL 4/14/2008

 RES ~~4/16/04~~ 9/30/04



Effect of Soil Depth on Surface Runoff, OPUS Element 23. Monsoon Climate. No Ks Variation

File: C: | USW_04 | OPUS_KINEROS | SMITH_OPUS3 | OPUSRO3.PGW
daw 9 - 30 - 04



Effect of Soil Depth and Soil Conductivity on Seepage Element 23, Monsoon Climate, No Seasonal Variation of Ks

File: C: | USW_04 | OPUS_KINEROS | SMITH_OPUS3 | OPUSSEEP3.PGW
daw 9 - 30 - 04

RL 4/14/2008

4/10/06 RES

K2O2 is a program in preparation by a team composed of Dave Goodrich and Carl Unkrich, ARS Tucson, and R.E. Smith consultant, Fort Collins; along with members of the U of Ariz. hydrology dept. This model is a continuous simulation extension of KINEROS2, restructured considerably, with additions of plant water use, soil ET, soil water redistribution and other processes which account for water movement and material transport.

The model is neither finished nor released, but the runoff/plant/soil water is working. Application to Split Wash data that was used in KINEROS2, and the 9-year record used for a single element, can be done with addition of appropriate modules to read Opus format data, and provide daily output.

Thornton and Running estimation. To use K2O2 with the Split Wash data, relative humidity needs to be used with T_{min} & T_{max} to estimate daily net radiation, R, as in Opus2. This method is from:

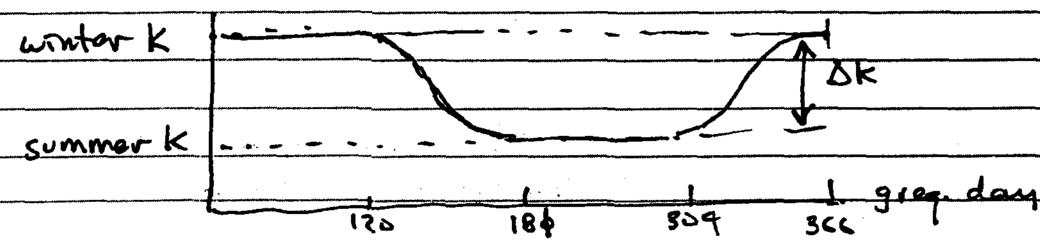
P.E. Thornton, & S.W. Running, 1999, "An improved algorithm for estimating incident solar radiation from measurements of temperature, humidity, and precipitation", Agri. and Forest Meteorology, v99: 211-228.

the method estimates vapor pressure from H (rel. humidity) and then estimates transmittance, using day of year to get mean sun angle. [It would not be very accurate, presumably, on dry but cloudy days.]

RES 9/20/06

Programming a preliminary version of K2O2 to deal with Opus2 input files over the last few months (occasionally) has resulted in several inserted functions to read and report results. This addition is encapsulated in an accompanying CD. Some of the read logic is from Opus 2, but can be modified since rainfall data is read continuously and not in units of blocks from storm start to next storm start. See Sci. Noteb. attachm. CD for files developed. K2O2 code not released yet.

10/30/06: in order to combine runs of K2 which used one K_s for winter and another for summer-fall, continuous runs w. seasonal shift is coded, as below.




```
!-----------------------------------------------------------------------
!
      subroutine soil_set_ksv ( iel, jday )
!
!  sets kbar for first layer and second based on time of year
!
      integer :: jday, dtd, iel
      real ::  fvm

      sdat => soilreads(iel)
 !
      fvm = 1.0 - sdat%kshift
      select case (jday )
        case (1:120)
          fvk = 1.0
        case (121:181)
          dtd = jday - 120
          fvk = 1.0 - real(dtd,4)/61.0 * sdat%kshift
        case (182:304)
          fvk = fvm
        case (305:366)
          dtd = jday - 304
          fvk = fvm + real(dtd,4)/62. * sdat%kshift
      end select
      end subroutine soil_set_ksv
 !
!-----------------------------------------------------------------------------
```

12/11/06 RES

Testing K202 indicates need for revision of desert veg. parameters to better match expected growth patterns. This model is somewhat simplified compared to opus. e.g., lignin content assumed. Changed parameters: (see p. 50)

Plant type	pot. dry matter Kg/ha	pot. Lai	DDEM	DDMX	Tgb °C	Tgmx	Cnv	root* mm
Desert Brush	2000	0.30	14.5	4500	-1	23	1.0	1000
Cool season grass	2000	1.5	180	800	0.5	16	6.0	240*
warm season grass	1500	1.5	500	1000	8.0	23	5.0	240*

* root depth depends on soil depth, except brush
many of these param's. are same as p.50

RL 4/14/2008

RL 4/14/2008

Roger E. Smith
PhD., P.E.
Engineering Consulting

819 Columbia Road
Fort Collins, CO 80525
(970) 493 2662

March 6, 2008

Stuart Stothoff
Center for Nuclear Regulatory Analysis
Southwest Research Institute
6220 Culebra Road
San Antonio, TX 78228

Subject: Watershed Modeling in the Yucca Mountain, NV Region

Dear Stuart:

Since our consulting activities have been suspended, I enclose herewith my CNWRA Scientific Notebook, #473. There are no entries since my last submission.

Sincerely,

Roger Smith

Roger E. Smith, P.E.

RL 4/14/2008

ADDITIONAL INFORMATION FOR SCIENTIFIC NOTEBOOK NO. 473

Document Date:	9/24/2001
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:	March 27, 2002
Operating System: (including version number)	Windows
Application Used: (including version number)	
Media Type: (CDs, 3 ½, 5 1/4 disks, etc.)	3 ½ disk
File Types: (.exe, .bat, .zip, etc.)	Files in ASCII format
Remarks: (computer runs, etc.)	KE documentation, GNFLUX output

ADDITIONAL INFORMATION FOR SCIENTIFIC NOTEBOOK NO. 473

Document Date:	9/24/2001
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:	March 2003
Operating System: (including version number)	Windows
Application Used: (including version number)	
Media Type: (CDs, 3 ½, 5 1/4 disks, etc.)	1 CD
File Types: (.exe, .bat, .zip, etc.)	
Remarks: (computer runs, etc.)	Files in ASCII format

ADDITIONAL INFORMATION FOR SCIENTIFIC NOTEBOOK NO. 473

Document Date:	9/24/2001
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:	3/13/07
Operating System: (including version number)	UNKNOWN
Application Used: (including version number)	OPUS
Media Type: (CDs, 3 ½, 5 1/4 disks, etc.)	1 CD
File Types: (.exe, .bat, .zip, etc.)	.f90, .out, .par
Remarks: (computer runs, etc.)	computer output and supporting routines

ADDITIONAL INFORMATION FOR SCIENTIFIC NOTEBOOK NO. 473

Document Date:	9/24/2001
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:	10/20/06
Operating System: (including version number)	UNKNOWN
Application Used: (including version number)	UNKNOWN
Media Type: (CDs, 3 ½, 5 1/4 disks, etc.)	3 ½ disk
File Types: (.exe, .bat, .zip, etc.)	.txt
Remarks: (computer runs, etc.)	FORTRAN 90 ROUTINES FOR OPUS