SCIENTIFIC NOTEBOOK

by

Scott Painter

Southwest Research Institute
Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

March 22, 1999

Table of Contents

INITIAL ENTRIES....3
Radionuclide Transport Analysis for Yucca Mountain....4
Objective:....4
3.22.99 Variogram analysis of sorption data for YM alluvium....5
4.8.99 Site-specific parameters for use in stochastic transport calculations....5
4.13.99 Integral Scale for W in RSF Model of Kd....6

INITIAL ENTRIES

Scientific notebook: #318 Vol. 1
Issued to: S. L. Painter
Original Issue Date: 22-March-1999

By agreement with CNWRA QA this notebook will be printed at approximate quarterly intervals. This computerized electronic notebook is intended to address the criteria of CNWRA QAP-001.

Radionuclide Transport Analysis for Yucca Mountain

Account number: 20-1402-871
Description: Radionuclide Transport Analysis.
Collaborators: Dr. D. Turner and Dr. V. Cvetkovic (consultant)

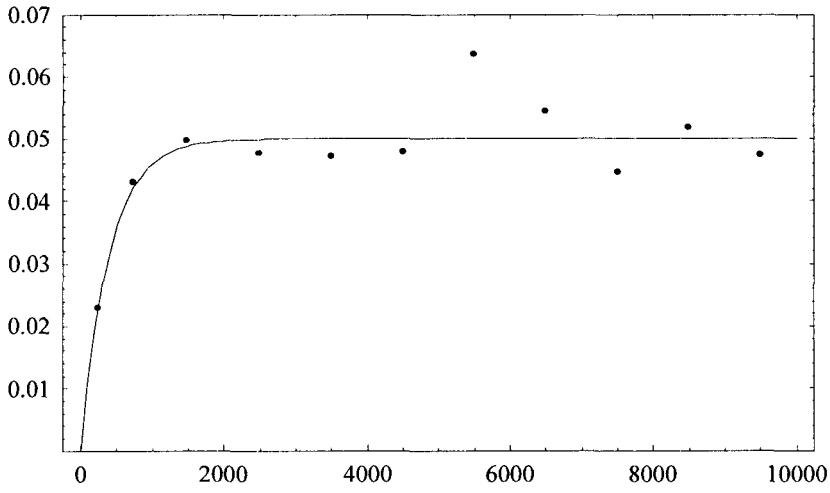
Objective:
Development and testing of transport analysis methodology for spatially variable systems including demonstrations and applications to the proposed repository at Yucca Mountain.

SCP 1/18/2010

3.22.99 Variogram analysis of sorption data for YM alluvium

Completed a preliminary look at spatial variability in sorption parameters in the alluvial aquifer near Yucca Mountain. The data provided by Dave Turner consists of Kd calculated from measured water chemistry, as described in IM 1402-871-810. The initial data were processed slightly to (arithmetically) average multiple values at a single location. The processed dataset can be found in ..\lsar\ymallv-av.xls. The analysis was performed using the Mathematica package ..\MathNB\tools\vgram2.nb. The detailed results for neptunium can be found in ..\lsar\np-vg.nb. The variogram analysis employed a maximum likelihood estimator (MLE), but was otherwise conventional. Because the dataset was rather limited, directional anisotropy was not tested. A logarithmic transform (base 10) was applied to the data before starting.

Results are shown below. This experimental variogram can be fitted well with a theoretical variogram without a nugget. The one shown has a sill of 0.05 and an integral scale of 400 meters. These were fit by eye in this preliminary analysis. (that is, no formal parameter fitting was applied) Other statistics are: Mean(log10(kd))=1.17 and Variance(log10(kd))=0.10. Here kd is is units of ml/g.



Semivariogram for Kd data (log10 transformed). X-axis is in meters.

4.8.99 Site-specific parameters for use in stochastic transport calculations

Developed two sets of hydrological parameters for YM alluvium. Parameter set 1 is based on the bimodal permeability distribution used in the TSPA-VA Section 8.5.2.2. Permeability = 10^{-13} m^2 = 289 m/y Hydraulic gradient from list of measured heads in Table 8.13. Head for J-12 = 727.3 m; head for NDOT = 705.8 m. deltax = 4067970-4055240. This implies J=0.00173 m/m. Using a porosity of 15% this implies a groundwater velocity of 3.3 m/yr. An integral scale of 2000 meters

and a ln-K variance of 3 are also consistent with this data. Note the ln-K variance is obtained in the above from the following calculation

$$\sigma_Y^2 = \left(P_1(13.1-14.1)^2 + P_2(13.1-12.5)^2\right)2.3^2 \cong 3$$

where P1=1/3 is the probability for a low permeability facies and P2=2/3 is the probability of a high permeability facies (coarse grain material). The factor 2.3^2 accounts for the change from base 10 logarithm to natural logarithm. The permeability values are from the above cited TSPA chapter.

Second set comes from USGS study SP23NM3 "Hydrogeology and Preliminary Three-Dimensional Finite-Element Groundwater Flow Model of the Site Saturated Zone, Yucca Mountain Nevada" and from DOE's "Regional Groundwater Flow and Tritium Transport Modeling and Risk Assessment of the Underground Test Area, Neveda Test Site, Nevada" October 1997. Here J=0.00145, K=527 m/yr and ln-K variance = 1.56.

Note also, the values given in the USFIC IRSR are consistent with the 289 m/yr estimate, and lower than the K=527 m/yr value.

Kd values are given in the 3-22-99 entry. Note that the values are based on a surface area of A=3 m^2/m^3 and need to be corrected for the additional variability in the surface area. Dave Turner estimates a mean of 3 and a range of 1-10 for the minerals at Yucca Mountain. Assuming a log-normal distribution and that Dave's range corresponds to the quantile range Q_{99}-Q_1, a ln-A variance of 0.25 is obtained. Thus the ln-Kd variance is enhanced from 0.57 to 0.82.

4.13.99 Integral Scale for W in RSF Model of Kd

Using the following model for K_d, we need to know the integral scale and variance for W given the same for Y and Z:

$$\frac{K_d}{K_{dG}} = Exp[\beta Y + W] \equiv Exp[Z].$$

This implies that $\sigma_Z^2 = \beta^2\sigma_Y^2 + \sigma_W^2$. The integral scale for Kd is

$$I_{K_d} = \int_0^\infty C_{kd}(\zeta)\,d\zeta = \frac{1}{\exp(\sigma_Z^2)-1}\int_0^\infty \exp[C_Z(\zeta)]-1\,d\zeta$$

but

$$I_{K_d} = \frac{1}{\exp\left[\beta^2\sigma_Y^2+\sigma_W^2\right]-1}\int_0^\infty \exp\left[\beta^2 C_Y(\zeta)+C_W(\zeta)\right]-1\,d\zeta$$

Equating the two and solving for I_w for fixed β and integral scale and variance for Y and Z gives the following table:

Table 1. Integral scale and variance for uncorrelated part of RSF model for Log Kd.
The Log K variance is $\sigma_Y^2 = 1.56$ for these.

β	Integral scale I_W	σ_W^2
0	400 m	0.820
-0.2	288 m	0.758
-0.3	127 m	0.680

Table 2. Integral scale and variance for uncorrelated part of RSF model for Log Kd.
The Log K variance is $\sigma_Y^2 = 3$ for these.

β	Integral scale I_W	σ_W^2
0	400 m	0.820
-0.1	347 m	0.790
-0.2	172 m	0.700

Other parameters appearing in the above are as follows: $\sigma_z^2 = 0.82$ I_Z=400 m, I_Y=2000 m.
Details in the above calculations can be found in ..\lsar\IntegralScale.nb

SCP 1/18/2010

Entries into Scientific Notebook No. 318 for the period 3.22.99 to 4.13.99 were made by Scott Painter.

No original text entered into this scientific notebook has been removed.

Scott Painter
1-18-10

ECS 1/18/2010

SCIENTIFIC NOTEBOOK

by

Scott Painter

Southwest Research Institute
Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

July 1, 1999

Table of Contents

INITIAL ENTRIES....3
Radionuclide Transport Analysis for Yucca Mountain....4
Objective:....4
7.6.99 Correction to entry of 4.8.99....5

INITIAL ENTRIES

Scientific notebook: #318 Vol. 2
Issued to: S. L. Painter
Original Issue Date: 22-March-1999

By agreement with CNWRA QA this notebook will be printed at approximate quarterly intervals. This computerized electronic notebook is intended to address the criteria of CNWRA QAP-001.

Radionuclide Transport Analysis for Yucca Mountain

Account number: 20-1402-871
Description: Radionuclide Transport Analysis.
Collaborators: Dr. D. Turner and Dr. V. Cvetkovic (consultant)

Objective:
Development and testing of transport analysis methodology for spatially variable systems including demonstrations and applications to the proposed repository at Yucca Mountain.

ECP 1/18/2010

7.6.99 Correction to entry of 4.8.99

Entry made on 4.8.99 contains an error. Last sentence should read: "Thus the ln-Kd variance is enhanced from 0.52 to 0.77.'' Mistake was probably due to mistranscribing "0.52" as "0.58".

RCS 1/18/2010

Entries into Scientific Notebook No. 318 for the period 7.6.99 to 7.6.99 were made by Scott Painter.

No original text entered into this scientific notebook has been removed.

Scott Painter
1-18-10

1/18/2010

SCIENTIFIC NOTEBOOK

by

Scott Painter

Southwest Research Institute
Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

August 1, 1999

Table of Contents

INITIAL ENTRIES....3
Radionuclide Transport Analysis for Yucca Mountain....4
Objective:....4
12.30.99 Input from Dr. V. Cvetkovic....5

INITIAL ENTRIES

Scientific notebook: #318 Vol. 3
Issued to: S. L. Painter
Original Issue Date: 22-March-1999

By agreement with CNWRA QA this notebook will be printed at approximate quarterly intervals. This computerized electronic notebook is intended to address the criteria of CNWRA QAP-001.

Radionuclide Transport Analysis for Yucca Mountain

Account number: 20-1402-871
Description: Radionuclide Transport Analysis.
Collaborators: Dr. D. Turner and Dr. V. Cvetkovic (consultant)

Objective:
Development and testing of transport analysis methodology for spatially variable systems including demonstrations and applications to the proposed repository at Yucca Mountain.

1/18/2010

12.30.99 Input from Dr. V. Cvetkovic.

Notes and a listing of a computer code used in a one-off calculation are included as an appendix. Also here is a manuscript delivered by Dr. Cvetkovic.

SP 1/18/2010

Entries into Scientific Notebook No. 318 for the period 12.30.99 to 12.30.99 were made by Scott Painter, as provided by V. Cvetkovic.

No original text entered into this scientific notebook has been removed.

Appendices referenced in text cannot be located.

Scott Painter
1-18-10

EC 1/18/2010

SCIENTIFIC NOTEBOOK

by

Scott Painter

Southwest Research Institute
Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

January 1, 2001

Table of Contents

INITIAL ENTRIES....3
Radionuclide Transport Analysis for Yucca Mountain....4
Objective:....4
1.25.01 Input from Dr. V. Cvetkovic on Colloidal Assisted Transport....5

INITIAL ENTRIES

Scientific notebook: #318 Vol. 4
Issued to: S. L. Painter
Original Issue Date: 22-March-1999

By agreement with CNWRA QA this notebook will be printed at approximate quarterly intervals. This computerized electronic notebook is intended to address the criteria of CNWRA QAP-001.

Radionuclide Transport Analysis for Yucca Mountain

Account number: 20-1402-871
Description: Radionuclide Transport Analysis.
Collaborators: Dr. D. Turner and Dr. V. Cvetkovic (consultant)

Objective:
Development and testing of transport analysis methodology for spatially variable systems including demonstrations and applications to the proposed repository at Yucca Mountain.

SP 1/18/2010

1.25.01 Input from Dr. V. Cvetkovic on Colloidal Assisted Transport

Notes on colloidal assisted transport are included as appendix. Listing of computer code used in a one-off calculation is included here.

```
*=====================================================================

*       COMPUTATION OF RADIONUCLIDE DISCHARGE IN A 3-PHASE SYSTEM

*               (groundwater-colloids-solid phase)


* Basic conditions/assumptions:

* FLOW/ADVECTION
*      - Hydraulic conductivity in aquifer is heterogeneous
*      - Groundwater flow is steady-state

* RADIONUCLIDE
*      - RN injected as pulse into groundwater
*      - RN sorbs to the solid-phase at equilibrium
*      - RN is irreversibly and kinetically transferred to colloids

* COLLOIDS
*      - Colloids subject to first-order removal/filtration
*      - Colloids are generated as a zero-order process
*      - Colloid concentration at steady-state
*      - Colloid concentration can vary spatially


*       Comment:  basic solution given in Cvetkovic (2000, PF)

*                       VERSION 2000-12-01

*=====================================================================

        real tlog(1000),fX(1000),fY(1000),f0(1000)

        real k2,KdG,Kc,Kprim,lambda

        implicit double precision (a-h)
        implicit double precision (p-z)

        common /param/k2,KdG,Kc,Kprim
        common /param2/alfa,eps,chi,cc0

*       character filin*20
*       character filout*20

*     write(*,*)'read filename input file'
*     read(*,'(a)')filin
```

```
*     write(*,*)'read filename output file'
*     read(*,'(a)')filout

*     open(3,file=filin)

      open(2,file='f.dat')

        pi=3.1415927


**************************************************
**** B E G I N   I N P U T   D A T A   ***********
**************************************************

*-------------------------------------------------
* GROUNDWATER RESIDENCE TIME - tau
*-------------------------------------------------
*       taum is mean [yr], tauvar is variance [yr^2]

        taum=8./.002
        tauvar=2.*1.56*8.*2./(.002)**2.

*-------------------------------------------------
* RADIONUCLIDE-RELATED PARAMETER
*-------------------------------------------------

* KdG: geometric mean of partitioning coefficient
*      (assumed constant, effective value)

* KdG=1800. for Plutonium ---- KdG=108 for Neptunium [-]
        KdG=37500.
*       KdG=108.

* Kc:   equilibrium partitioning coefficient for radionuclide
*       between aqueous phase and colloids - NEGLECTED!
        Kc=.0

* lambda: decay rate [1/T]
        lambda=2.9*10.**(-5.)
*       lambda=3.24*10.**(-7.)
*       lambda=.0


*--------------------------------------------------
* COLLOID-RELATED PARAMETERS:
*--------------------------------------------------

* alfa: transfer rate from aqueous to colloids defined per
*       unit colloid concentration [L^3/MT]
        alfa=1.e-6

* chi:  colloid generation rate, defined as colloid concentration
*       per unit time [ML^3/T]
        chi=.0

* eps:  colloid removal rate (first-order linear process) [1/T]
*       (input given below!)
*        eps=.001
```

```
* cc0:  colloid concentration at injection point [ML^3]
        cc0=1.

***********************************************
****    E N D   I N P U T   D A T A    ***********
***********************************************


*-------------------------------------------------------------------
* P r e p a r e   p a r a m e t e r s   f o r   c o m p u t a t i o n
*-------------------------------------------------------------------

        Kprim=KdG/(1.+Kc)

        taumR=taum*(1.+Kprim)
        timelast=8.*taumR

        tloglast=log10(timelast)

        nt1=200

        nt2=nt1
        nt=nt1+nt2

*     ntt=200
*     ttl=taum+10.*dsqrt(tauvar)
*     dtt=(ttl-t)/ntt

        tRlog=log10(taumR)

        tlog1=1.


        dtlog1=(tRlog-tlog1)/nt1
        dtlog2=(tloglast-tRlog)/nt2


        tlog(0)=tlog1

        do 55 m=0,nt1-1
 55     tlog(m+1)=tlog(m)+dtlog1

        do 66 m=nt1,nt-1
 66     tlog(m+1)=tlog(m)+dtlog2


*        write(*,*)'<tau>R=',taumR

*        do 77 k=0,nt
* 77     write(*,*)k,tlog(k),10.**tlog(k)


*-----------------------------------------------------------
*     S t a r t   t h e   c o m p u t a t i o n   l o o p s
*-----------------------------------------------------------


* Parameter set loop (alfa and eps)
        do 7000 nset=1,3
*        nset=4
```

```
*        if(nset.eq.1)alfa=10.**(-9.00)
*        if(nset.eq.2)alfa=10.**(-8.00)
*        if(nset.eq.3)alfa=10.**(-7.00)
*        if(nset.eq.4)alfa=10.**(-6.00)
*        if(nset.eq.5)alfa=10.**(-5.00)
*        if(nset.eq.6)alfa=10.**(-4.00)
*        if(nset.eq.7)alfa=10.**(-3.00)

        if(nset.eq.1)eps=1.e-2
        if(nset.eq.2)eps=1.e-3
        if(nset.eq.3)eps=1.e-4
*        if(nset.eq.4)alfa=10.**(-6.00)
*        if(nset.eq.5)alfa=10.**(-5.00)
*        if(nset.eq.6)alfa=10.**(-4.00)
*        if(nset.eq.7)alfa=10.**(-3.00)

        qnset=nset

        write(2,*)
        write(2,105)qnset
        write(2,*)

* Time loop
        do 1000 j=1,nt

           time=10.**tlog(j)

        g10=.0
        cumI0=.0

        g1=.0
        cumI=.0

        do 100 k=1,j

           t1=10.**tlog(k-1)
           t2=10.**tlog(k)

           if(k.eq.1)t1=10.**tlog1

           g10=dexp(-alfa*(t1-time)/Kprim)*ftau(t1)
           g20=dexp(-alfa*(t2-time)/Kprim)*ftau(t2)

           g1=dexp(-eps*t1)*zf(-(t1-time)/Kprim)*ftau(t1)
           g2=dexp(-eps*t2)*zf(-(t2-time)/Kprim)*ftau(t2)

           cumI0=cumI0+(g10+g20)*(t2-t1)/2.
           cumI=cumI+(g1+g2)*(t2-t1)/2.

*           write(2,101)j,k,time,t2
*           write(*,101)j,k,time,t1,t2

 100    continue

        g10=.0
        cumII0=.0

        g1=.0
        cumII=.0
```

```
        sum=.0
        do 200 k=1,j

            t1=10.**tlog(k-1)/(1.+Kprim)
            t2=10.**tlog(k)/(1.+Kprim)

           if(k.eq.1)t1=10.**tlog1/(1.+Kprim)

           g10=dexp(alfa*(t1-time)/Kprim)*ftau(t1)
           g20=dexp(alfa*(t2-time)/Kprim)*ftau(t2)

           g1=dexp(-eps*t1)*zf(-(t1-time)/Kprim)*ftau(t1)
           g2=dexp(-eps*t2)*zf(-(t2-time)/Kprim)*ftau(t2)

           cumII0=cumII0+(g10+g20)*(t2-t1)/2.
           cumII=cumII+(g1+g2)*(t2-t1)/2.

           q1=dexp(-eps*t1)*ftau(t1)
           q2=dexp(-eps*t2)*ftau(t2)

 200    continue


       gammaY=alfa*dexp(-lambda*time)*(cumI-cumII)/Kprim
       gammaX=dexp(-lambda*time-alfa*time/(1.+Kprim))*
     %           ftau(time/(1.+Kprim))/(1.+Kprim)


       gammaY0=alfa*dexp(-lambda*time)*(cumI0-cumII0)/Kprim
       gammaX0=dexp(-lambda*time-alfa*time/(1.+Kprim))*
     %           ftau(time/(1.+Kprim))/(1.+Kprim)


       dtime=10.**tlog(j)-10.**tlog(j-1)

       flux0=dexp(-lambda*time)*ftau(time/(1.+KdG))/(1.+KdG)

       fX(j)=gammaX
       fY(j)=gammaY
       f0(j)=flux0


*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
*** SIMPLIFIED CASE WITH TAU=TAUM
*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++


           if(time.ge.taum)g10=dexp(-alfa*(time-taum)/Kprim)
           if(time.ge.taum)g1=dexp(-eps*taum)*zf((time-taum)/Kprim)

*       gammaY=alfa*dexp(-lambda*time)*(cumI-cumII)/Kprim
*           g1=dexp(-eps*t1)*zf(-(t1-time)/Kprim)*ftau(t1)
*           g1=dexp(-eps*t1)*zf((t1-time)/Kprim)*ftau(t1)
*           g1=dexp(-eps*t1)*zf(-(t1-time)/Kprim)*ftau(t1)

           if(time.gt.taumR)g10=.0
           if(time.gt.taumR)g1=.0
```

```
          gamY_c=alfa*dexp(-lambda*time)*g1/Kprim
*          gammaY0=alfa*dexp(-lambda*time)*g10/Kprim


*+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

*           write(*,101)time,sum

       write(2,101)time,gamY_c,gammaY
       write(*,101)time,gamY_c,gammaY

*       write(*,101)time,flux0,gammaY0,gammaY

*         write(*,101)time,ftau(time/(1.+Kprim))/(1.+Kprim),gammaX,gammaY
*             write(2,101)t/4000.,cum*
*         write(2,101)time,ftau(time/(1.+Kprim))/(1.+Kprim),gammaX,gammaY


 1000 continue


* Computation of cumulative discharge

        cumX=.0
        cumY=.0
        cum0=.0

        goto 2001

        write(*,*)

      do 2000 m=1,nt-1

         t1=10.**tlog(m)
         t2=10.**tlog(m+1)

         cumX=cumX+(fX(m+1)+fX(m))*(t2-t1)/2.
         cumY=cumY+(fY(m+1)+fY(m))*(t2-t1)/2.
         cum0=cum0+(f0(m+1)+f0(m))*(t2-t1)/2.

*         write(*,*)t2,cum0,cumX,cumY,cumX+cumY

 2000    continue
 2001    continue

*         write(*,101)alfa,cum0,cumX,cumY,cumX+cumY
         write(*,*)dexp(-eps*taum)

 7000    continue


 101  format(7e12.5)
 105  format('zone t="nset=',f4.0,'"')

      stop
      end



        function ftau(x)
```

```
*  subroutine for evaluating the pdf for travel time;

      implicit double precision (a-h)
      implicit double precision (p-z)

*      common /com1/varx,xm,pi

      real mux,Iy

                U=.002
                sigm2Y=1.56
                Iy=2.
                xx=8.

                xm=xx/U
                varx=2.*sigm2Y*xx*IY/U**2.

                pi=3.141592

        Sigx=(log(varx/(xm*xm)+1.))

            mux=log(xm)-Sigx/2.

        qx=log(x)-mux

        ww=-.5*qx*qx/Sigx

      ftau=exp(ww)/(x*sqrt(2.*pi*Sigx))

      return
      end


      function zf(q)

* subroutine for computing the auxilary function z

      implicit double precision (a-h)
      implicit double precision (p-z)

      common /param2/alfa,eps,chi,cc0

      if(eps.lt.1.e-15) then

      cc=chi*q+cc0
      w=cc0*q

      else

      cc=chi*(1.-dexp(-eps*q))/eps+cc0*dexp(-eps*q)
      w=chi*(q-(1.-dexp(-eps*q))/eps)/eps+cc0*(1.-dexp(-eps*q))/eps

      endif

      zf=cc*dexp(-eps*q-alfa*w)

      return
      end
```

RCS 1/18/2010

SN 318 Vol. 4. Pg. 12, Scott Painter

Entries into Scientific Notebook No. 318 for the period 1.25.01 to 1.25.01 were made by Scott Painter, as provided by V. Cvetkovic.
No original text entered into this scientific notebook has been removed.

Scott Painter
1-18-10

1/18/2010

SCIENTIFIC NOTEBOOK

by

Scott Painter

Southwest Research Institute
Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

September 4, 2001

Table of Contents

INITIAL ENTRIES.....3
Radionuclide Transport Analysis for Yucca Mountain.....4
Objective:.....4
9.4.01 Input from Dr. V. Cvetkovic on Kinetic Sorption on Colloids.....5

INITIAL ENTRIES

Scientific notebook: #318 Vol. 5
Issued to: S. L. Painter
Original Issue Date: 22-March-1999

By agreement with CNWRA QA this notebook will be printed at approximate quarterly intervals. This computerized electronic notebook is intended to address the criteria of CNWRA QAP-001.

Radionuclide Transport Analysis for Yucca Mountain

Account number: 20-1402-871
Description: Radionuclide Transport Analysis.
Collaborators: Dr. D. Turner and Dr. V. Cvetkovic (consultant)

Objective:
Development and testing of transport analysis methodology for spatially variable systems including demonstrations and applications to the proposed repository at Yucca Mountain.

ECS 1/18/2010

9.4.01 Input from Dr. V. Cvetkovic on Kinetic Sorption on Colloids

Input from V. Cvetkovic on re-evaluation of the published kinetic sorption data of N. Lu is attached as appendix.

10.11.01 Input from Dr. V. Cvetkovic on Kinetic Sorption on Colloids

Additional details on kinetic model used in re-evaluation of N. Lu data are attached as appendix. Also there is FORTRAN codes used to generate figures for journal manuscript "Significance of kinetics for sorption on colloids: Modeling and experimental interpretation considerations" which will be submitted to Environmental Science and Technology.

EVC 1/18/2010

Entries into Scientific Notebook No. 318 for the period 9.4.01 to 10.11.01 were made by Scott Painter, as provided by V. Cvetkovic.
No original text entered into this scientific notebook has been removed.

Scott Painter
1-18-10

1/18/2010

SCIENTIFIC NOTEBOOK

by

Scott Painter

Southwest Research Institute
Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

February 11, 2002

Table of Contents

INITIAL ENTRIES....3
Radionuclide Transport Analysis for Yucca Mountain....4
Objective:....4
2.11.02 Correction to entry from V. Cvetkovic....5
2.14.02 Check of equation 5 in 9.4.01 entry....5

INITIAL ENTRIES

Scientific notebook: #318E Vol. 6
Issued to: S. L. Painter
Original Issue Date: 22-March-1999

By agreement with CNWRA QA this notebook will be printed at approximate quarterly intervals. This computerized electronic notebook is intended to address the criteria of CNWRA QAP-001.

Radionuclide Transport Analysis for Yucca Mountain

Account number: 20-1402-871
Description: Radionuclide Transport Analysis.
Collaborators: Drs. D. Turner, D. Pickett, and V. Cvetkovic (consultant)

Objective:
Development and testing of transport analysis methodology for spatially variable systems including demonstrations and applications to the proposed repository at Yucca Mountain.

1/18/2010

SN 318E Vol. 6. Pg. 4, Scott Painter

2.11.02 Correction to entry from V. Cvetkovic

Entries dated 9.4.01 and 10.11.01 contained errors. In Equation 4 of the 9.4.01 entry (in Vol 5 as appendix) the term in the dominator should read $s' + \omega + k_r$ instead of $s' + k_f$. Further, all occurrences of $s' + k_r$ should read $s' + \omega + k_r$. The typographical error does not propagate further. Correction should also be made to Equation 4 of 10.11.01 entry (Vol 5, Appendix B).

2.14.02 Check of equation 5 in 9.4.01 entry

The following is a check on equation 5 of 9.4.01 entry. The procedure in going from equation 4 to equation 5 is to divide numerator and denominator by kr (fast reverse rate) replace kf/kr by the value Cc Kc (same notation as in the 9.4.01 entry), let kr go to infinity, replace 1 + Cc Kc with Rc, replace R + Cc Kc with Rt. The following is from a Mathematica calculation. For simplicity, I use the term kc to denote the produce Cc Kc in the following:

For the numerator:

Expand [redacted] **ga + kr** [redacted] **ega + beta** [redacted]

$$\text{beta} + \text{omega} + \frac{\text{beta omega}}{\text{kr}} + \frac{\text{omega}^2}{\text{kr}} + \text{s} + \frac{\text{beta s}}{\text{kr}} + \frac{2\ \text{omega s}}{\text{kr}} + \frac{\text{s}^2}{\text{kr}}$$

% [redacted] kr ® Infinity

beta + omega + s

entries here are obscured by font compatibility issue.
Scott Painter
1-19-10

For the denominator:

Ex [redacted] **a + kf +** [redacted] **ga + beta** [redacted] **kr** [redacted]
- [redacted] lph [redacted] beta [redacted] **ega + kr** [redacted] **f k** [redacted] **ega + beta** [redacted]

$$\text{alpha omega} + \text{beta omega} + \frac{\text{beta kf omega}}{\text{kr}} + \text{omega}^2 + \frac{\text{alpha omega}^2}{\text{kr}} + \frac{\text{beta omega}^2}{\text{kr}} + \frac{\text{kf omega}^2}{\text{kr}} + \frac{\text{omega}^3}{\text{kr}} + \text{alpha s} + \frac{\text{beta kf s}}{\text{kr}} + \text{omega s} + \frac{2\ \text{alpha omega s}}{\text{kr}} + \frac{\text{beta omega s}}{\text{kr}} + \frac{2\ \text{kf omega s}}{\text{kr}} + \frac{2\ \text{omega}^2\ \text{s}}{\text{kr}} + \text{beta R s} + \text{omega R s} + \frac{\text{beta omega R s}}{\text{kr}} + \frac{\text{omega}^2\ \text{R s}}{\text{kr}} + \frac{\text{alpha s}^2}{\text{kr}} + \frac{\text{kf s}^2}{\text{kr}} + \frac{\text{omega s}^2}{\text{kr}} + \text{R s}^2 + \frac{\text{beta R s}^2}{\text{kr}} + \frac{2\ \text{omega R s}^2}{\text{kr}} + \frac{\text{R s}^3}{\text{kr}}$$

% [redacted] kf [redacted] kr ® [redacted] kc;

% [redacted] kr ® Infinity

$$\text{alpha omega} + \text{beta omega} + \text{beta kc omega} + \text{omega}^2 + \text{kc omega}^2 + \text{alpha s} + \text{beta kc s} + \text{omega s} + 2\ \text{kc omega s} + \text{beta R s} + \text{omega R s} + \text{kc s}^2 + \text{R s}^2$$

Collect [redacted]

al [redacted] ega + beta omega + beta kc omega + omega^2 + kc om [redacted] ega^2 [redacted]
[redacted] lph [redacted] + beta kc + omega + 2 kc omega + beta R + omega R [redacted] kc + [redacted] 2

Simplify [redacted]

alpha [redacted] omeg [redacted] + s [redacted] eta [redacted] omega + s [redacted] + [redacted] ga + [redacted] kc + [redacted]

Entries here are obscured by font issue.
Scott Painter
1-19-10

% kc + R ® Rt

alpha omega + s beta omega + s + ga + Rt s

% kc + 1 ® Rc

alpha omega + s beta omega + s eg Rc + Rt s

Simplify omega Rc + alpha ga + beta pha beta

alpha omega + s beta omega + s eg Rc + Rt s

The last expression above is same as denominator of Eq 5.

ECP 1/18/2010

Entries into Scientific Notebook No. 318E for the period 2.11.02 to 2.14.02 were made by Scott Painter, as provided by V. Cvetkovic.
No original text entered into this scientific notebook has been removed.

Scott Painter
1-18-10

ECP 1/18/2010

SCIENTIFIC NOTEBOOK

by

Scott Painter

Southwest Research Institute
Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

August 24, 2002

Table of Contents

INITIAL ENTRIES....3
Radionuclide Transport Analysis for Yucca Mountain....4
Objective:....4
8.24.02 Initial Estimate of Potential Colloid Facilitated Transport of Plutonium....5

INITIAL ENTRIES

Scientific notebook: #318E Vol. 7
Issued to: S. L. Painter
Original Issue Date: 22-March-1999

By agreement with CNWRA QA this notebook will be printed at approximate quarterly intervals. This computerized electronic notebook is intended to address the criteria of CNWRA QAP-001.

Radionuclide Transport Analysis for Yucca Mountain

Account number: 20-1402-871
Description: Radionuclide Transport Analysis.
Collaborators: Drs. D. Turner, D. Pickett, and V. Cvetkovic (consultant)

Objective:
Development and testing of transport analysis methodology for spatially variable systems including demonstrations and applications to the proposed repository at Yucca Mountain.

ECP 1/18/2010

SN 318E Vol. 7 Pg. 4, Scott Painter

8.24.02 Initial Estimate of Potential Colloid Facilitated Transport of Plutonium

An extended abstract on potential effects of colloid facilitated transport of plutonium is provided in Appendix A. This is to be submitted to a workshop on Colloids in the Environment. More detailed write-up will be provided in a future journal article. This entry provides supporting material for the abstract. Note that the Equation 1 and Figure 1 are from a previously reviewed journal article and are not addressed in this entry.

To get to Equation 2, we consider a streamtube carrying the constant volumetric flow q and neglect diffusion. The normalized solute concentration X is given by

$$R\frac{\partial X}{\partial t} + v\frac{\partial X}{\partial x} = -\alpha X(Y_m - Y) + \alpha_r Y - \lambda RX$$

and the normalized concentration sorbed onto colloids Y is

$$\frac{\partial Y}{\partial t} + v\frac{\partial Y}{\partial x} = \alpha X(Y_m - Y) - \alpha_r Y - \lambda Y$$

where v is velocity and the other constants are defined in Appendix A. The first two terms on the right side form a "bi-linear" model for kinetically controlled sorption analogous to equation 13.31 of Domenico and Schwartz (Physical and Chemical Hydrogeology, 1990 Wiley and Sons, New York.) Restricting attention to steady state, and switching to groundwater residence time $\tau = x/v$ as the independent variable, results in the Equation 2 in appendix A.

Equation 2 of Appendix A is solved by a Runge-Kutta method to produce $Z(\tau) = X(\tau) + Y(\tau)$. Given a model for the groundwater residence time distribution, the distribution of Z is easily obtained since
$\Pr(Z > \zeta) = \Pr(\tau > t^*)$ where $Z(t^*) = \zeta$. A previously published log-normal model was used for the residence time distribution (Effect of Heterogeneity on Radionuclide Retardation in the Alluvial Aquifer Near Yucca Mountain, Nevada, Ground Water 39(3), May-June 2001, Pages 326—338).

The calculations producing the results in Figure 2 are straightforward and are contained in the following fortran.

```
**  Site-specific calculations for Figure 2 of extended
*     abstract for Colloid workshop in Denmark, sept 19-20, 2002

!    (ym is a function of tau and alfa_f=0.1 1/yr is fixed)

!     2002-08-23  Stockholm

*-------------------------------------------------------------

*    Define and initialize variables

      real X(20001),Y(20001),Z(20001),tt(20001)

      real kappa,Kd,eps
      real alfa_f,lambda,sm,ej0,a0,en,xx
      real X0,fac_ar,alfa_f,eps,C00,lambda,kappa,Kd,R

*    Link variables to routines

      Common /runge/dtau,nn
      Common /site/sm,ej0,a0,en,xx,ym0
      Common /moments/taum,vartau
      Common /linear/X0,fac_ar,alfa_f,eps,C00,lambda,kappa,Kd,R


* read output index
* 6 gives Z_1%, 11 gives Z_50% (median), and 25 gives Z_95%
       write(*,*)'read index (6 - 500yrs, 11 - median, 25 - 10000yrs)'
       read(*,*)index

* read reversibility factor fac_ar=alpha_r/alpha _f; in manus
*      we use a=1/fac_ar
       write(*,*)'read fac_ar'
       read(*,*)fac_ar


***    I N P U T   P A R A M E T E R S --------------------
! forward sorption rate on colloids for Figs 3 and 4
! (fixed as 0.1 1/yr, from Painter et al., EST) [1/yr]
        alfa_f=.1
! dimensionless sorption coefficient [-]
        Kd=1.8e+3
        R=1+Kd
! decay rate [1/yr]
        lambda=1.e-5
! injection rate for discharge output (output normalized)
        X0=1.
! estimated cross-sectional area for injection [m^2]
        a0=0.01
! assumed Pu injection rate [g/yr]
        ej0=1.e-4
! aquifer porosity
        en=0.25
! neglect equilibrium sorption on colloids
        kappa=.0
! neglect colloidal removal
        eps=.0
*-----------------------------------------------
! the range of sm given by D.Turner is:
```

```
*      sm=2.4e-4
*      sm=6.e-7
* we use the range 1.e-7  -- .5e-4 for sensitivity
*-----------------------------------------------


*-------------------------------------------------------------
! hydraulic/flow/advective properties
*-------------------------------------------------------------

! gw mean velocity
                U=.002
! logK variance
                sigm2Y=1.56
! logK integral scale
                Iy=2.
! trasport distance in alluvial aquifer, to compliance boundary
                xx=8.
! mean and variance of gw residence time tau
                taum=xx/U
                vartau=2.*sigm2Y*xx*IY/U**2.
*-------------------------------------------------------------
! initialize gw residence time [yr]
        tau=.0
        dtau=1.

! number of steps for tau
      nn=10000

! open output files
      open(3, file='f.dat')
      open(4, file='ffa.dat')
      open(5, file='ffb.dat')


!     do-loop for parameter sm
! sm is the maximum site-specific colloid-bounded Pu concentration
! for a given colloidal concentration; here a sensitivity parameter
! in the range specified by D.Turner
      do 2000 k=1,4

            if(k.eq.1)sm=1.e-7
            if(k.eq.2)sm=1.e-6
            if(k.eq.3)sm=1.e-5
            if(k.eq.4)sm=1.e-4

! call the main routine for calculations
! solves two non-linear ODE's (eqs(2)) using Runge-Kutta
       call xytau(X,Y,tt)

! compute total steady-state tracer discharge at compliance boundary
      do j=0,nn,1
         Z(j)=X(j)+Y(j)
      enddo

!     do-loop for output at specified gw residence times tau
      do i=1,25
         if(i.eq.1)m=i
         if(i.ge.2.and.i.le.5)m=(i-1)*50
         if(i.gt.5)m=(i-5)*500
```

```
        ttt=tt(m)

! compute CCDF of total discharge
        ccdf_z=1.-ccdf_ln(ttt)

! compute total discharge using the linear models
! (reversible or irreversible) and write final results on output files
! z_reversible is based on analytical solutions of linearized eq(2)
! z_irreversible is based on analytical solutions of linearized eq(2)
!   with alpha_r=0

            if(fac_ar.ne..0) then
            z_rev=z_reversible(ttt)

! output for Tables
            if(i.eq.index)write(3,101)sm,z(m),Z_rev

            else
            z_irr=z_irreversible(ttt)
! output for Tables
            if(i.eq.index)write(3,101)sm,z(m),Z_irr
            endif

! output for Fig 2
        if(k.eq.1)write(4,101)z(m),ccdf_z
        if(k.eq.4)write(5,101)z(m),ccdf_z
*-------------------------------------------------------------

! compute tracer discharge without colloids
*           if(k.eq.1.and.mm.eq.1)XXX=X0*exp(-R*lambda*ttt)

      enddo

 2000 continue
! close do-loop for parameter fac_ar

 1000 continue
! close do-loop for parameter sm

 101  Format(5e12.5)
 102  Format('zone t="',e7.1,',',e7.1'"')
 103  Format('zone t="',e7.1,'"')

      Stop
      end


      Subroutine rkutta(i,tau,X,Y,stepX,stepY)
* routine for Runge-Kutta computations

*     Define variables

      real X(20001),Y(20001)

      real tau,stepX,stepY,dtau
      real k1,k2,k3,k4,l1,l2,l3,l4
      real C00,kappa,Kd,eps,R
      real lambda
```

```
      Integer i

      Common /runge/dtau,nn
      Common /site/sm,ej0,a0,en,xx,ym0
      Common /linear/X0,fac_ar,alfa_f,eps,C00,lambda,kappa,Kd,R
*
* Calculate Runge k's, l's and m's
*
      qX=X(i)
      qY=Y(i)

      k1=dtau*functX(tau,qX,qY)
      l1=dtau*functY(tau,qX,qY)

      k2=dtau*functX(tau+dtau/2.,qX+k1/2.,qY+l1/2.)
      l2=dtau*functY(tau+dtau/2.,qX+k1/2.,qY+l1/2.)

      k3=dtau*functX(tau+dtau/2.,qX+k2/2.,qY+l2/2.)
      l3=dtau*functY(tau+dtau/2.,qX+k2/2.,qY+l2/2.)

      k4=dtau*functX(tau+dtau,qX+k3,qY+l3)
      l4=dtau*functY(tau+dtau,qX+k3,qY+l3)

      stepX=(k1+2.*k2+2.*k3+k4)/6.
      stepY=(l1+2.*l2+2.*l3+l4)/6.

      Return
      End


      Real Function functX(tau,xdum,ydum)
* routine defining the discretized function X

* Define variables

      Real C00,kappa,Kd,eps,R,Rc0,xdum,ydum
      real Ym,alfa_f,alfa_r,lambda
      real sm,ej0,a0,en,xx

      Common /site/sm,ej0,a0,en,xx,ym0
      Common /linear/X0,fac_ar,alfa_f,eps,C00,lambda,kappa,Kd,R


! compute volumetric flow rate qqq for streamtube [L/yr]
!  (assumed perfectly correlated to gw residence time tau)
      qqq=xx*1000.*a0*en*1000./tau

      ym=sm*qqq/ej0

!-------------------------------------------------
             alfa_r=fac_ar*alfa_f
             Re=1+alfa_f/alfa_r
!-------------------------------------------------

      C0=C00*exp(-eps*tau)
      R=1.+kappa*C0+Kd
      Rc0=1.+kappa*C0

      A=-alfa_f/Rc0 -R*lambda
      B=alfa_r/Rc0
```

```
      C=alfa_f/(Rc0*ym)

      functX= A*xdum +B*ydum + C*xdum*ydum
      Return
      End


      Real Function functY(tau,xdum,ydum)
* routine defining the discretized function Y

* Define variables
      Real C0,kappa,Kd,eps,Rc,xdum,ydum
      real Ym,alfa_f,alfa_r,lambda,R
      real sm,ej0,a0,en,xx

      Common /site/sm,ej0,a0,en,xx,ym0
      Common /linear/X0,fac_ar,alfa_f,eps,C00,lambda,kappa,Kd,R

! compute volumetric flow rate qqq for streamtube [L/yr]
!  (assumed perfectly correlated to gw residence time tau)
      qqq=xx*1000.*a0*en*1000./tau
      ym=sm*qqq/ej0

!-----------------------------------------------
             alfa_r=fac_ar*alfa_f
             Re=1+alfa_f/alfa_r
!-----------------------------------------------

      C0=C00*exp(-eps*tau)
      Rc=1.+kappa*C0

      A=alfa_f
      B=-alfa_r-lambda-eps
      C=-alfa_f/ym

      functY= A*xdum +B*ydum + C*xdum*ydum

      Return
      End


         subroutine xytau(X,Y,tt)
* main routine for computations

         real X(20001),Y(20001),tt(20001)

      real stepX,stepY
      real kappa,Kd,eps,X0,Y0,C00,R
      real lambda

*    Link variables to routines

      Common /runge/dtau,nn
      Common /site/sm,ej0,a0,en,xx,ym0
      Common /linear/X0,fac_ar,alfa_f,eps,C00,lambda,kappa,Kd,R

      X0=1.
      Y0=.0

! set condition at injection point
```

```
      X(0)=X0
      Y(0)=Y0

! compute volumetric flow rate qqq for streamtube [L/yr]
!  (assumed perfectly correlated to gw residence time tau)
      qqq=xx*1000.*a0*en*1000./tau
      ym=sm*qqq/ej0

!-------------------------------------------------
             alfa_r=fac_ar*alfa_f
             Re=1+alfa_f/alfa_r
!-------------------------------------------------

         do l=1,nn
            tt(l)=.0
            enddo
            tau=.0

       Do i=0,nn

         tau=tau+dtau

         Call rkutta(i,tau,X,Y,stepX,stepY)

         X(i+1)=X(i)+stepX
         Y(i+1)=Y(i)+stepY

         tt(i+1)=tau

         enddo

         return
         end



        function erfc(x)
      implicit double precision (a-h,o-z)
      sign=1.
      if(x.lt.0.)sign=-1.
      xx=abs(x)
      if(xx.lt.1.9)q=sign*erf1(xx)
      if(xx.ge.1.9)q=sign*erf2(xx)
      erfc=1.-q
      return
      end



      function erf1(x)
      implicit double precision (a-h,o-z)

      sum=x
      fac=1.
      sig=1.

      do 10 i=1,10
      k=2*i+1
      fac=fac*i
      sig=sig*(-1)
```

```
      term=sig*x**k/(k*fac)
      sum=sum+term
10    continue

      erf1=sum*2./sqrt(3.141592)
      return
      end


      function erf2(x)
      implicit double precision (a-h,o-z)

      sum=1.
      fact=1.
      sig=1.
      do 10 j=1,10
      k=2.*j+1
      sig=sig*(-1.)
      term=sig*fact/(2.*x*x)**j
      sum=sum+term
      fact=fact*k
10    continue

      erf2=1.-sum*dexp(-x*x)/(x*sqrt(3.141592))

      return
      end



      function ccdf_ln(x)
* routine for computing ccdf of gw residence time tau

      double precision erfc,arg
      Common /moments/taum,vartau

      xm=taum
      varx=vartau

      xsig=log(varx/(xm*xm)+1.)
      xG=xm*exp(-xsig/2.)
      arg=log(x/xG)/sqrt(2.*xsig)

      ccdf_ln=.5*erfc(arg)

      return
      end


         function z_reversible(ttt)
* routine for computing the linear reversible Z

         real X0,fac_ar,alfa_f,eps,C00,lambda,kappa,Kd
         real Re,alfa_r,R
         Common /linear/X0,fac_ar,alfa_f,eps,C00,lambda,kappa,Kd,R

!-----------------------------------------------
             alfa_r=fac_ar*alfa_f
             Re=1+alfa_f/alfa_r
!-----------------------------------------------
```

```
      aaa=alfa_r*Re
      qqR=exp(-aaa*ttt)
      qq=exp(-alfa_r*ttt)
      qqw=1.-exp(-(alfa_f+lambda*R)*ttt)

        X_rev=((1./Re)*(1.-qqR)+qqR)*exp(-lambda*R*ttt)

        Y_rev=alfa_f*exp(-alfa_r*ttt)*
     &   (-qqw/(Re*(lambda*R+alfa_f))+qqw/(lambda*R+alfa_f))+
     &   (exp(-alfa_r*ttt)-exp(-alfa_r*ttt-(lambda*R-alfa_r)*ttt))*
     &       alfa_f/(Re*(lambda*R-alfa_r))

        Z_reversible=X_rev+Y_rev
        return
        end



      function z_irreversible(ttt)
* routine for computing the linear irreversible Z

      real X0,fac_ar,alfa_f,eps,C00,lambda,kappa,Kd,R
      Common /linear/X0,fac_ar,alfa_f,eps,C00,lambda,kappa,Kd,R

      C0=C00*exp(-eps*ttt)
      Rc=1.+kappa*C0

      aa=alfa_f/Rc+R*lambda
      bb=lambda+eps
      cc=bb-aa

      X_irr=X0*exp(-aa*ttt)
      Y_irr=alfa_f*X0*exp(-bb*ttt)*(exp(cc*ttt)-1.)/cc
      z_irreversible=x_irr+y_irr

      return
      end
```

ECS 1/18/2010

Entries into Scientific Notebook No. 318E for the period 8.24.02 to 8.24.02 were made by Scott Painter, as provided by V. Cvetkovic.
No original text entered into this scientific notebook has been removed.

Scott Painter
1-18-09 (SP)
10

1/18/2010

SP

SCIENTIFIC NOTEBOOK

by

Scott Painter

Southwest Research Institute
Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

May 12, 2003

Table of Contents

INITIAL ENTRIES....3
Radionuclide Transport Analysis for Yucca Mountain....4
Objective:....4
5.12.03 Model/data sensitivities on colloid facilitated transport....5
5.22.03 Model/data sensitivities on colloid facilitated transport (revised)....5

INITIAL ENTRIES

Scientific notebook: #318E Vol. 8
Issued to: S. L. Painter
Original Issue Date: 22-March-1999

By agreement with CNWRA QA this notebook will be printed at approximate quarterly intervals. This computerized electronic notebook is intended to address the criteria of CNWRA QAP-001.

Radionuclide Transport Analysis for Yucca Mountain

Account number: 20-1402-871
Description: Radionuclide Transport Analysis.
Collaborators: Drs. D. Turner, D. Pickett, and V. Cvetkovic (consultant)

Objective:
Development and testing of transport analysis methodology for spatially variable systems including demonstrations and applications to the proposed repository at Yucca Mountain.

ECS 1/18/2010

5.12.03 Model/data sensitivities on colloid facilitated transport

Draft manuscript on colloid facilitated transport was received from V. Cvetkovic. The work looks at model and data sensitivities for colloid facilitated transport. The analysis considers a constant plutonium release and the effect of interest in this semi-generic sensitivity study is normalized discharged in steady state. This manuscript and his notebook entries to support this are listed as appendices to this notebook. The model used includes plutonium mass exchanges between solution, porous matrix, mobile colloids, colloids that are irreversibly removed from water, and colloids that are reversibly attached to the porous matrix. See appendices for details.

5.22.03 Model/data sensitivities on colloid facilitated transport (revised)

Appendix C contains a significantly revised manuscript, which supersedes the entry on 5.12.03. Numerical calculation for figures 1-4 are the same as before and are documented in the entry of 5.12.03. Mathematica notebooks for the new figure 5-7 are attached (*Bilinear2.nb* for Figure 5). *Linear.nb* for Figures 6 and 7.

6.02.03 Revised Figures 2 and 3

Figures 2 and 3 of the manuscript "Parameter and model sensitivities for colloid facilitated radionuclide transport on the field scale" were revised. Calculations are in the mathematica notebook *Generic2.nb*, which is attached.

1/18/2010

Entries into Scientific Notebook No. 318E for the period 5.12.03 to 6.02.03 were made by Scott Painter, as provided by V. Cvetkovic.
No original text entered into this scientific notebook has been removed.

Scott Painter
1-18-10

RCP 1/18/2010

Colloid Facilitated Transport Results with Bi-Linear Sorption Model

No Colloid Sorption or Filtration

Scott Painter
5.22.03

Preliminaries

```
<< Graphics`Graphics`
```

■ Show the system to be solved

```
alphar = alphaf / a;
a2 = alphaf j0 / (L porosity Sm);
TraditionalForm[ x'[t] == -(lambda * R + alphaf ) x[t] + a2 t x[t] y[t] + (alphar ) y[t] ]
TraditionalForm[
 y'[t] == alphaf x[t] - x[t] y[t] a2 t - (alphar + epsilon + lambda Rc) y[t] ]
```

$$x'(t) == (-\text{alphaf} - \text{lambda}\, R)\, x(t) + \frac{\text{alphaf j0}\, t\, y(t)\, x(t)}{L\, \text{porosity Sm}} + \frac{\text{alphaf}\, y(t)}{a}$$

$$y'(t) == \text{alphaf}\, x(t) - \frac{\text{alphaf j0}\, t\, y(t)\, x(t)}{L\, \text{porosity Sm}} - \left(\frac{\text{alphaf}}{a} + \text{epsilon} + \text{lambda Rc}\right) y(t)$$

■ Fixed Parameters

j0 is j_0/A_0 in draft manuscript. Other symbols are self evident.

```
lambda = 2.88811 10^-5;
j0 = 0.01;
R = 3401;
porosity = 0.15;
alphaf = 0.1;
L = 5000;
```

■ Define the Solution

```
BiLinear[tau_, Sm_, Rc_, epsilon_, a_ ] :=
 Module[ {solution, alphar = alphaf / a, a2 = alphaf j0 / (L porosity Sm)},
solution = NDSolve[ {x'[t] == -(lambda * R + alphaf ) x[t] + a2 t x[t] y[t] + (alphar ) y[t] ,
     y'[t] == alphaf x[t] - x[t] y[t] a2 t - (alphar + epsilon + lambda Rc) y[t] ,
     y[0] == 0, x[0] == 1}, {x, y}, {t, 0, 2702}, AccuracyGoal → 100,
    MaxSteps → 5000 ];
  Extract[(x[tau] + y[tau]) /. solution, 1]]

BiLinear[ Sm_, Rc_, epsilon_, a_] := {BiLinear[40, Sm, Rc, epsilon, a],
  BiLinear[334, Sm, Rc, epsilon, a], BiLinear[2702, Sm, Rc, epsilon, a]}

SetAttributes[BiLinear, Listable]
```

■ test

```
BiLinear[ 10^-1, 1, 0, 1000]
```

{0.503338, 0.491649, 0.406093}

Table 1 from Appendix A

```
smlist = Flatten[ Table[{10^-4, 10^-1, Infinity}, {4}]]
```

$\left\{\frac{1}{10000}, \frac{1}{10}, \infty, \frac{1}{10000}, \frac{1}{10}, \infty, \frac{1}{10000}, \frac{1}{10}, \infty, \frac{1}{10000}, \frac{1}{10}, \infty\right\}$

```
alist = Flatten[ Transpose[ Table[ {Infinity, 1000, 100, 10}, {3}]]]
```

{∞, ∞, ∞, 1000, 1000, 1000, 100, 100, 100, 10, 10, 10}

```
TableForm[ Partition[BiLinear[ 40, smlist, 1, 0, alist], 3] ]
```

0.401443	0.504085	0.504148
0.400427	0.503338	0.503401
0.391578	0.496683	0.496749
0.323711	0.436239	0.436324

```
TableForm[ Partition[BiLinear[ 334, smlist, 1, 0, alist], 3] ]
```

0.0416983	0.499645	0.499708
0.0417861	0.491649	0.491755
0.0383063	0.425393	0.425792
0.0126463	0.104231	0.104585

```
TableForm[ Partition[BiLinear[ 2702, smlist, 1, 0, alist], 3] ]
```

0.00559113	0.466616	0.466675
0.00480695	0.406093	0.40842
0.00180069	0.120271	0.123387
8.45868×10^{-8}	1.04972×10^{-6}	1.05711×10^{-6}

Figures 5a and 5b (Appendix C)

```
result1 =
 Table[sm = 10^im; {sm / 1000., BiLinear[ 334, sm , 1, 0, 1000 ]}, {im, -4, -1, 0.1}]
```

$\{\{1.\times10^{-7}, 0.0417861\}, \{1.25893\times10^{-7}, 0.054397\},$
$\{1.58489\times10^{-7}, 0.0740361\}, \{1.99526\times10^{-7}, 0.0978708\},$
$\{2.51189\times10^{-7}, 0.119392\}, \{3.16228\times10^{-7}, 0.140282\}, \{3.98107\times10^{-7}, 0.162779\},$
$\{5.01187\times10^{-7}, 0.190771\}, \{6.30957\times10^{-7}, 0.249859\}, \{7.94328\times10^{-7}, 0.403487\},$
$\{1.\times10^{-6}, 0.466365\}, \{1.25893\times10^{-6}, 0.478076\}, \{1.58489\times10^{-6}, 0.482603\},$
$\{1.99526\times10^{-6}, 0.485146\}, \{2.51189\times10^{-6}, 0.486811\}, \{3.16228\times10^{-6}, 0.487984\},$
$\{3.98107\times10^{-6}, 0.488844\}, \{5.01187\times10^{-6}, 0.489491\}, \{6.30957\times10^{-6}, 0.489984\},$
$\{7.94328\times10^{-6}, 0.490365\}, \{0.00001, 0.490661\}, \{0.0000125893, 0.490892\},$
$\{0.0000158489, 0.491073\}, \{0.0000199526, 0.491216\}, \{0.0000251189, 0.491328\},$
$\{0.0000316228, 0.491417\}, \{0.0000398107, 0.491487\}, \{0.0000501187, 0.491542\},$
$\{0.0000630957, 0.491586\}, \{0.0000794328, 0.491621\}, \{0.0001, 0.491649\}\}$

```
result2 = Table[sm = 10^im; {sm / 1000, BiLinear[ 334, sm , 1, 0, 100]}, {im, -4, -1, 0.1}]
```

$\{\{\frac{1}{10000000}, 0.0383063\}, \{1.25893\times10^{-7}, 0.0484721\},$
$\{1.58489\times10^{-7}, 0.0612859\}, \{1.99526\times10^{-7}, 0.077183\},$
$\{2.51189\times10^{-7}, 0.0966272\}, \{3.16228\times10^{-7}, 0.120316\}, \{3.98107\times10^{-7}, 0.149888\},$
$\{5.01187\times10^{-7}, 0.189537\}, \{6.30957\times10^{-7}, 0.244858\}, \{7.94328\times10^{-7}, 0.30536\},$
$\{1.\times10^{-6}, 0.348687\}, \{1.25893\times10^{-6}, 0.374256\}, \{1.58489\times10^{-6}, 0.389642\},$
$\{1.99526\times10^{-6}, 0.399527\}, \{2.51189\times10^{-5}, 0.406243\}, \{3.16228\times10^{-6}, 0.410996\},$
$\{3.98107\times10^{-6}, 0.414459\}, \{5.01187\times10^{-5}, 0.417037\}, \{6.30957\times10^{-6}, 0.418985\},$
$\{7.94328\times10^{-6}, 0.420474\}, \{0.00001, 0.421622\}, \{0.0000125893, 0.422513\},$
$\{0.0000158489, 0.423208\}, \{0.0000199526, 0.423752\}, \{0.0000251189, 0.42418\},$
$\{0.0000316228, 0.424516\}, \{0.0000398107, 0.424782\}, \{0.0000501187, 0.424991\},$
$\{0.0000630957, 0.425157\}, \{0.0000794328, 0.425289\}, \{0.0001, 0.425393\}\}$

```
result3 = Table[sm = 10^im; {sm / 1000, BiLinear[ 334, sm , 1, 0, 10]}, {im, -4, -1, 0.1}]
```

$\{\{\frac{1}{10000000}, 0.0126463\}, \{1.25893\times10^{-7}, 0.0159537\},$
$\{1.58489\times10^{-7}, 0.0200803\}, \{1.99526\times10^{-7}, 0.0251334\},$
$\{2.51189\times10^{-7}, 0.0311404\}, \{3.16228\times10^{-7}, 0.0379967\}, \{3.98107\times10^{-7}, 0.0454465\},$
$\{5.01187\times10^{-7}, 0.0531214\}, \{6.30957\times10^{-7}, 0.0606275\}, \{7.94328\times10^{-7}, 0.0676348\},$
$\{1.\times10^{-6}, 0.0739276\}, \{1.25893\times10^{-6}, 0.0794086\}, \{1.58489\times10^{-6}, 0.0840716\},$
$\{1.99526\times10^{-6}, 0.0879712\}, \{2.51189\times10^{-6}, 0.0911903\}, \{3.16228\times10^{-6}, 0.0938228\},$
$\{3.98107\times10^{-6}, 0.0959599\}, \{5.01187\times10^{-6}, 0.0976867\}, \{6.30957\times10^{-6}, 0.0990761\},$
$\{7.94328\times10^{-6}, 0.100191\}, \{0.00001, 0.101084\}, \{0.0000125893, 0.101797\},$
$\{0.0000158489, 0.102366\}, \{0.0000199526, 0.10282\}, \{0.0000251189, 0.103181\},$
$\{0.0000316228, 0.103469\}, \{0.0000398107, 0.103698\}, \{0.0000501187, 0.10388\},$
$\{0.0000630957, 0.104025\}, \{0.0000794328, 0.10414\}, \{0.0001, 0.104231\}\}$

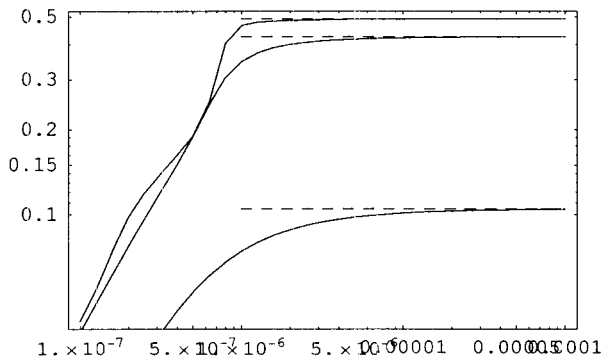
```
result4 = Table[sm = 10^im; {sm / 1000, BiLinear[ 40, sm , 1, 0, 100]}, {im, -5, -1, 0.1}]
```

$\{\{\frac{1}{100000000}, 0.052189\}, \{1.25893\times10^{-8}, 0.0613662\},$
$\{1.58489\times10^{-8}, 0.0734498\}, \{1.99526\times10^{-8}, 0.0896155\}, \{2.51189\times10^{-8}, 0.111623\},$
$\{3.16228\times10^{-8}, 0.141876\}, \{3.98107\times10^{-8}, 0.182905\}, \{5.01187\times10^{-8}, 0.235322\},$
$\{6.30957\times10^{-8}, 0.294206\}, \{7.94328\times10^{-8}, 0.349003\}, \{1.\times10^{-7}, 0.391578\},$
$\{1.25893\times10^{-7}, 0.421391\}, \{1.58489\times10^{-7}, 0.441733\}, \{1.99526\times10^{-7}, 0.455816\},$
$\{2.51189\times10^{-7}, 0.465817\}, \{3.16228\times10^{-7}, 0.473095\}, \{3.98107\times10^{-7}, 0.478499\},$
$\{5.01187\times10^{-7}, 0.482574\}, \{6.30957\times10^{-7}, 0.485683\}, \{7.94328\times10^{-7}, 0.488077\},$
$\{1.\times10^{-6}, 0.489933\}, \{1.25893\times10^{-6}, 0.491379\}, \{1.58489\times10^{-6}, 0.492511\},$
$\{1.99526\times10^{-6}, 0.4934\}, \{2.51189\times10^{-6}, 0.4941\}, \{3.16228\times10^{-6}, 0.494651\},$
$\{3.98107\times10^{-6}, 0.495087\}, \{5.01187\times10^{-6}, 0.495431\}, \{6.30957\times10^{-6}, 0.495704\},$
$\{7.94328\times10^{-6}, 0.49592\}, \{0.00001, 0.496091\}, \{0.0000125893, 0.496227\},$
$\{0.0000158489, 0.496334\}, \{0.0000199526, 0.49642\}, \{0.0000251189, 0.496488\},$
$\{0.0000316228, 0.496541\}, \{0.0000398107, 0.496584\}, \{0.0000501187, 0.496618\},$
$\{0.0000630957, 0.496645\}, \{0.0000794328, 0.496666\}, \{0.0001, 0.496683\}\}$

```
result5 =
 Table[sm = 10^im; {sm / 1000, BiLinear[ 2702, sm , 1, 0, 100]}, {im, -4, -1, 0.1}]
```

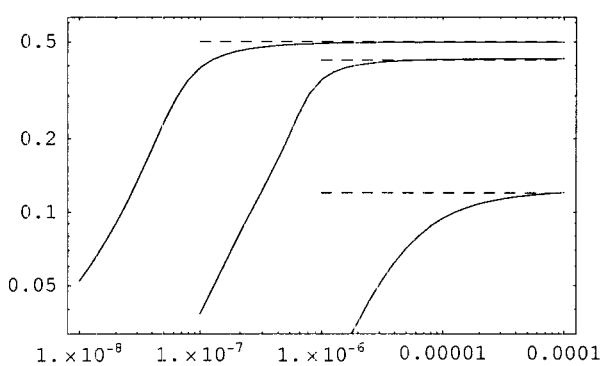
$\{\{\frac{1}{10000000}, 0.00180069\}, \{1.25893\times10^{-7}, 0.00226695\},$
$\{1.58489\times10^{-7}, 0.00285393\}, \{1.99526\times10^{-7}, 0.0035929\}, \{2.51189\times10^{-7}, 0.00452322\},$
$\{3.16228\times10^{-7}, 0.00569444\}, \{3.98107\times10^{-7}, 0.00716877\},$
$\{5.01187\times10^{-7}, 0.00902482\}, \{6.30957\times10^{-7}, 0.0113616\}, \{7.94328\times10^{-7}, 0.0143038\},$
$\{1.\times10^{-6}, 0.0180062\}, \{1.25893\times10^{-6}, 0.0226464\}, \{1.58489\times10^{-6}, 0.0283979\},$
$\{1.99526\times10^{-6}, 0.0353645\}, \{2.51189\times10^{-6}, 0.0434931\}, \{3.16228\times10^{-6}, 0.0525075\},$
$\{3.98107\times10^{-6}, 0.0619289\}, \{5.01187\times10^{-6}, 0.0712007\}, \{6.30957\times10^{-6}, 0.0798418\},$
$\{7.94328\times10^{-6}, 0.0875438\}, \{0.00001, 0.0941805\}, \{0.0000125893, 0.0997608\},$
$\{0.0000158489, 0.104373\}, \{0.0000199526, 0.108139\}, \{0.0000251189, 0.111192\},$
$\{0.0000316228, 0.11365\}, \{0.0000398107, 0.115623\}, \{0.0000501187, 0.117202\},$
$\{0.0000630957, 0.118464\}, \{0.0000794328, 0.119469\}, \{0.0001, 0.120271\}\}$

```
DisplayTogether[
 LogLogListPlot[ result1, PlotJoined → True],
 LogLogListPlot[ result3, PlotJoined → True],
 LogLogListPlot[ result2, PlotJoined → True],
 LogLogListPlot[ {{10^-6, 0.4258}, { 10^-4, 0.4258}},
  PlotJoined → True, PlotStyle → Dashing[{0.02, 0.02}] ] ,
 LogLogListPlot[ {{10^-6, 0.1046}, { 10^-4, 0.1046}},
  PlotJoined → True,  PlotStyle → Dashing[{0.02, 0.02}]] ,
 LogLogListPlot[ {{10^-6, 0.492}, { 10^-4, 0.492}}, PlotJoined → True,
  PlotStyle → Dashing[{0.02, 0.02}] ] , Frame → True, Axes → False ]
```



- Graphics -

```
DisplayTogether[
 LogLogListPlot[ result4, PlotJoined → True],
 LogLogListPlot[ result5, PlotJoined → True],
 LogLogListPlot[ result2, PlotJoined → True],
 LogLogListPlot[ {{10^-7, 0.50}, { 10^-4, 0.50}},
  PlotJoined → True, PlotStyle → Dashing[{0.02, 0.02}] ],
 LogLogListPlot[ {{10^-6, 0.42}, { 10^-4, 0.42}},
  PlotJoined → True, PlotStyle → Dashing[{0.02, 0.02}] ],
 LogLogListPlot[ {{10^-6, 0.12}, { 10^-4, 0.12}},
  PlotJoined → True, PlotStyle → Dashing[{0.02, 0.02}] ],
 Frame → True , PlotRange → {Automatic, {-1.5, -0.2}}, Axes → False
]
```



- Graphics -

Figures 2 and 3:
Generic Sensitivity to Sorption Model

This notebook contains revised figures 2 and 3 for the manuscript "Parameter and model sensitivities for colloid facilitated radionuclide transport on the field scale". Scott Painter
6.2.03

■ Define the Solution

```
<< Graphics`Graphics`
```

```
R = 1800;
```

```
lambda = 3 10^-5 * 500.
```

```
0.015
```

```
Linear[tau_, alphaf_, a_ ] := Module[ {solution, alphar = alphaf / a},
solution = NDSolve[ {x'[t] = -alphaf x[t] + alphar y[t] - lambda R x[t] ,
     y'[t] = alphaf x[t] - alphar y[t] - lambda y[t] , y[0] = 0, x[0] = 1},
    {x, y}, {t, 0, 100}, AccuracyGoal → 30,
    MaxSteps → 5000 ];
  Extract[(x[tau] + y[tau]) /. solution, 1]]
```

```
General::spell1 : Possible spelling error: new symbol name "alphar" is similar to existing symbol "alphaf".
```

```
BiLinear[tau_, Ym_, alphaf_, a_ ] := Module[ {solution, alphar = alphaf / a},
solution = NDSolve[ {x'[t] = -alphaf x[t] * (1 - y[t] / Ym) + alphar y[t] - lambda R x[t] ,
     y'[t] = alphaf x[t] * (1 - y[t] / Ym) - alphar y[t] - lambda y[t] ,
     y[0] = 0, x[0] = 1}, {x, y}, {t, 0, 100}, AccuracyGoal → 30,
    MaxSteps → 5000 ];
  Extract[(x[tau] + y[tau]) /. solution, 1]]
```

```
Langmuir[tau_, Ym_, alphaf_, a_ ] :=
 Module[ {solution, alphar = alphaf / a, beta = a / Ym},
solution = NDSolve[ {x '[t] == - alphaf x[t] / (1 + beta x[t]) + alphar y[t] - lambda R x[t]  ,
     y '[t] == alphaf x[t] / (1 + beta x[t]) - alphar y[t] - lambda y[t] ,
     y[0] == 0, x[0] == 1}, {x, y}, {t, 0, 100}, AccuracyGoal → 30,
    MaxSteps → 5000 ];
  Extract[(x[tau] + y[tau]) /. solution, 1]]
```

```
General::spell1 : Possible spelling error: new symbol name "beta" is similar to existing symbol "Beta".
```

```
BiLinear[0.1, 10^-1, 1, 1]
```

```
0.0895689
```

```
Langmuir[ 0.1, 10^-1, 1, 1]
```

```
0.0719326
```

```
Linear[0.1, 100, 10000]
```

```
0.786174
```

Figure 2 : Small sorption capacity

```
Ym = 10^-5;
```

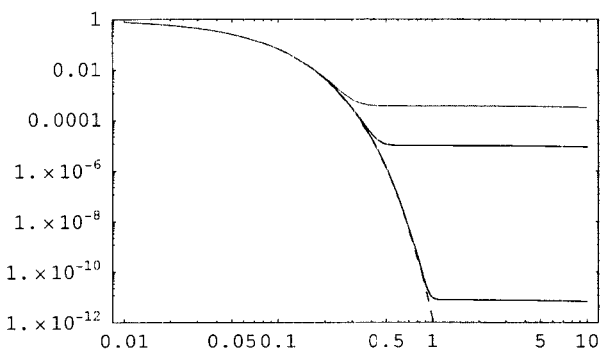
■ Figure 2a

```
a = 10^4;
alphaf = 0.01
```

```
0.01
```

1/18/2010

```
g1 = DisplayTogether[
  LogLogPlot[ Exp[-tau * lambda * R],
   {tau, 0.01, 10}, PlotStyle → Dashing[{0.02, 0.02}] ],
  LogLogPlot[ BiLinear[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Langmuir[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Linear[tau, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200, PlotStyle → RGBColor[1, 0, 0] ],
  PlotRange → {All, {-12, 0}} , Frame → True ]
```



- Graphics -

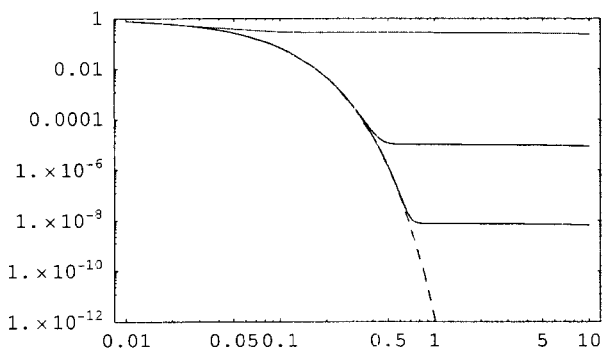
■ Figure 2b

```
a = 10^4;
alphaf = 10.
```

10.

ECP 1/18/2010

```
g2 = DisplayTogether[
  LogLogPlot[ Exp[-tau * lambda * R],
   {tau, 0.01, 10}, PlotStyle → Dashing[{0.02, 0.02}] ],
  LogLogPlot[ BiLinear[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Langmuir[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Linear[tau, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200, PlotStyle → RGBColor[1, 0, 0] ],
  PlotRange → {All, {-12, 0}} , Frame → True ]
```



- Graphics -

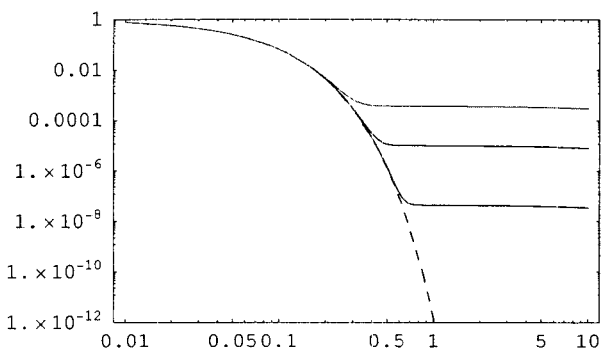
■ Figure 2c

```
a = 1;
alphaf = 0.01
```

0.01

ES 1/18/2010

```
g3 = DisplayTogether[
  LogLogPlot[ Exp[-tau * lambda * R],
   {tau, 0.01, 10}, PlotStyle → Dashing[{0.02, 0.02}]],
  LogLogPlot[ BiLinear[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Langmuir[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Linear[tau,  alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200, PlotStyle → RGBColor[1, 0, 0] ],
  PlotRange → {All, {-12, 0}} , Frame → True ]
```



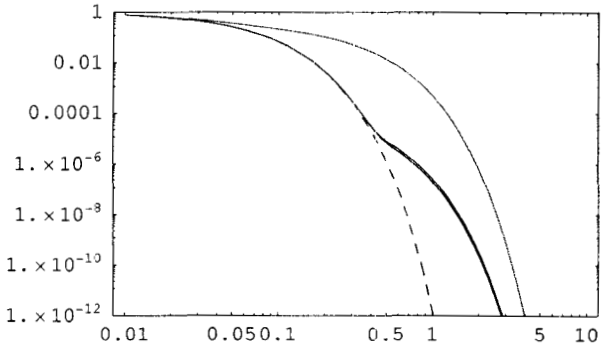
- Graphics -

■ Figure 2d

```
a = 1;
alphaf = 10;
```

ECS 1/18/2010

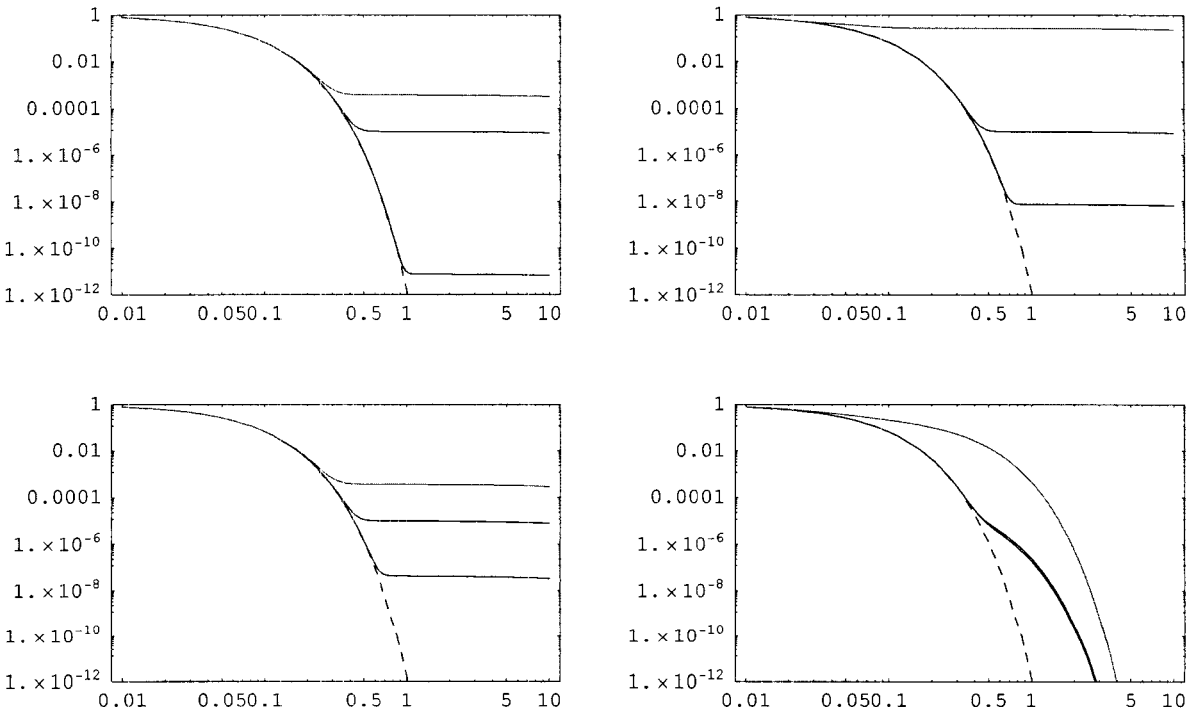
```
g4 = DisplayTogether[
  LogLogPlot[ Exp[-tau * lambda * R],
   {tau, 0.01, 10}, PlotStyle → Dashing[{0.02, 0.02}]],
  LogLogPlot[ BiLinear[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Langmuir[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Linear[tau, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200, PlotStyle → RGBColor[1, 0, 0] ],
  PlotRange → {All, {-12, 0}} , Frame → True ]
```



- Graphics -

ED 1/18/2010

```
Show[ GraphicsArray[ {{g1, g2}, {g3, g4}}] ]
```




```
- GraphicsArray -
```

Figure 3 : Large sorption capacity

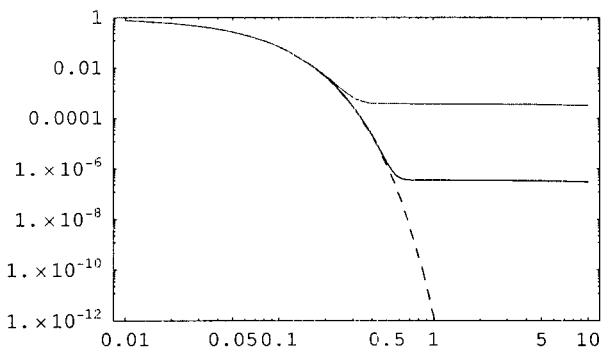
```
Ym = 1;
```

■ Figure 3a

```
a = 10^4;
alphaf = 0.01
```

```
0.01
```

```
g1 = DisplayTogether[
  LogLogPlot[ Exp[-tau * lambda * R],
   {tau, 0.01, 10}, PlotStyle → Dashing[{0.02, 0.02}] ],
  LogLogPlot[ BiLinear[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Langmuir[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Linear[tau, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200, PlotStyle → RGBColor[1, 0, 0] ],
  PlotRange → {All, {-12, 0}} , Frame → True ]
```



- Graphics -

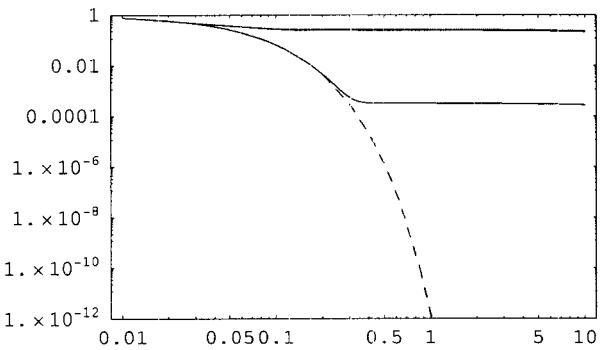
■ Figure 3b

```
a = 10^4;
alphaf = 10.
```

10.

ES 1/18/2010

```
g2 = DisplayTogether[
  LogLogPlot[ Exp[-tau * lambda * R],
   {tau, 0.01, 10}, PlotStyle → Dashing[{0.02, 0.02}] ],
  LogLogPlot[ BiLinear[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Langmuir[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Linear[tau, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200, PlotStyle → RGBColor[1, 0, 0] ],
  PlotRange → {All, {-12, 0}} , Frame → True ]
```



- Graphics -

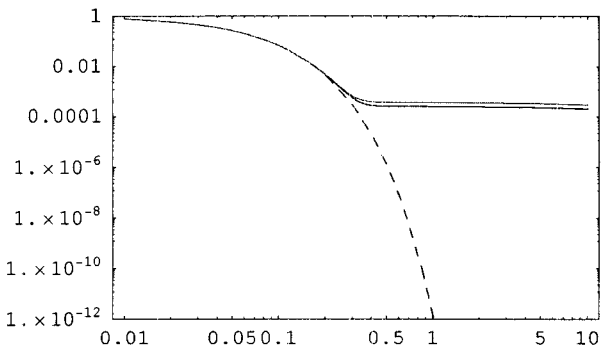
■ Figure 3c

```
a = 1;
alphaf = 0.01
```

0.01

ECS 1/18/2010

```
g3 = DisplayTogether[
  LogLogPlot[ Exp[-tau * lambda * R],
   {tau, 0.01, 10}, PlotStyle → Dashing[{0.02, 0.02}]],
  LogLogPlot[ BiLinear[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Langmuir[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Linear[tau,  alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200, PlotStyle → RGBColor[1, 0, 0] ],
  PlotRange → {All, {-12, 0}} , Frame → True ]
```



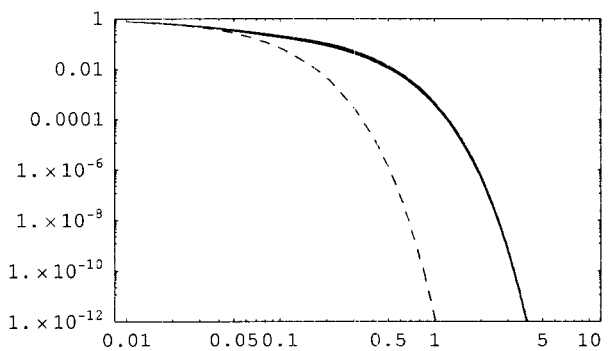
- Graphics -

■ Figure 3d

```
a = 1;
alphaf = 10;
```

RCP 1/18/2010

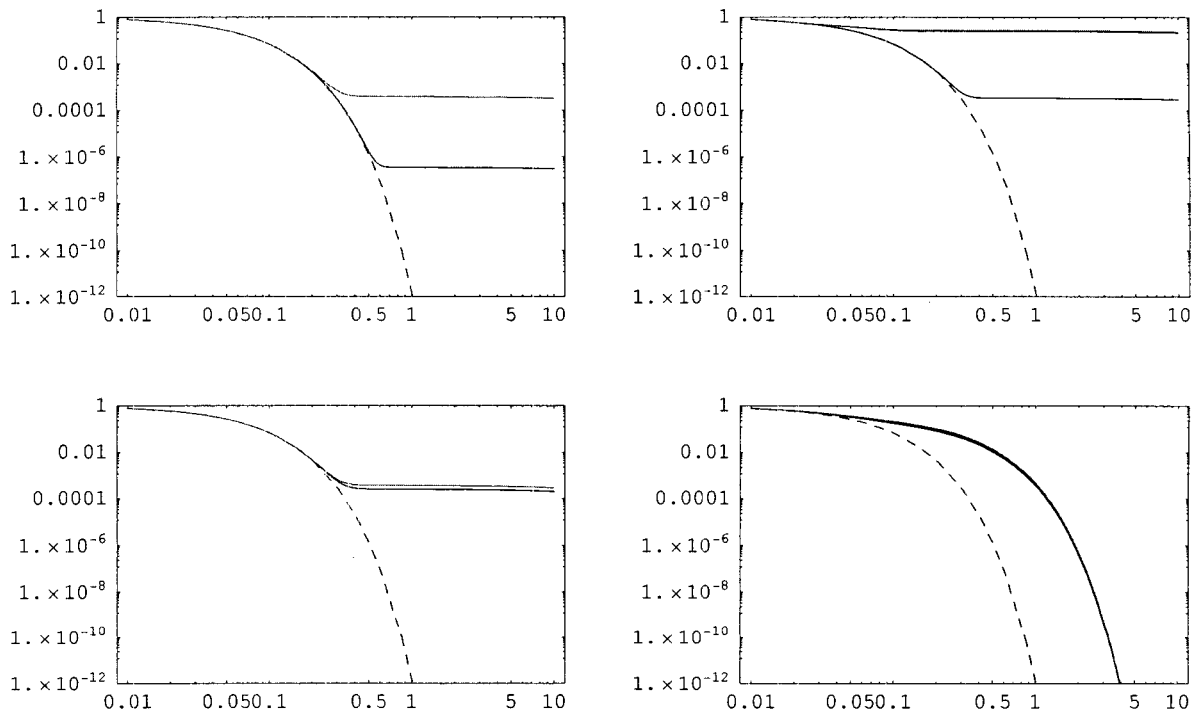
```
g4 = DisplayTogether[
  LogLogPlot[ Exp[-tau * lambda * R],
   {tau, 0.01, 10}, PlotStyle → Dashing[{0.02, 0.02}]],
  LogLogPlot[ BiLinear[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Langmuir[tau, Ym, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200],
  LogLogPlot[ Linear[tau, alphaf, a] ,
   {tau, 0.01, 10}, PlotPoints → 200, PlotStyle → RGBColor[1, 0, 0] ],
  PlotRange → {All, {-12, 0}} , Frame → True ]
```



- Graphics -

ECP 1/18/2010

```
Show[ GraphicsArray[ {{g1, g2}, {g3, g4}}] ]
```



- GraphicsArray -

RCP 1/14/2010

Colloid Facillitated Transport: Linear Model

scott painter
5-22-05

Preliminaries

```
<< Graphics`Graphics`
```

Show the system to be solved:

```
TraditionalForm[
 x'[t] ==
  -(lambda * R + alphaf Rc ) x[t] + (alphar Rc + epsilon alphar / (lambda + alphar) ) y[t] ]
TraditionalForm[ y'[t] == (alphaf Rc ) x[t] - (Rc alphar + epsilon + lambda Rc ) y[t] ]
```

$$x'(t) == (-\text{lambda}\, R - \text{alphaf Rc})\, x(t) + \left(\frac{\text{alphar epsilon}}{\text{alphar} + \text{lambda}} + \text{alphar Rc}\right) y(t)$$

$$y'(t) == \text{alphaf Rc}\, x(t) - (\text{epsilon} + \text{alphar Rc} + \text{lambda Rc})\, y(t)$$

Fixed Parameters

j0 = j0/A0 in manuscript. Other parameters are self evident.

```
lambda = 2.88811 10^-5;
j0 = 0.01;
R = 3401;
porosity = 0.15;
alphaf = 0.1;
L = 5000;
```

linear model (full model)

Set up functions

```
FullLinearModel[ tau_, Rc_, epsilon_, a_] :=
  Module[ {alphar = alphaf / a, alphaf1, alphaf2},
   solution = NDSolve[ {x'[t] == -(lambda * R + alphaf + alphaf * (Rc - 1) ) x[t] +
        (alphar + alphar (Rc - 1) + epsilon alphar / (lambda + alphar) ) y[t] ,
      y'[t] == (alphaf + alphaf * (Rc - 1) ) x[t] -
        (alphar + epsilon + lambda Rc + alphar (Rc - 1)) y[t] , y[0] == 0, x[0] == 1},
     {x, y}, {t, 0, 2702}, AccuracyGoal → 40, MaxSteps → 5000];
   Extract[(x[tau] + y[tau]) /. solution, 1] ];

FullLinearModel[ Rc_, epsilon_, a_] := Module[ {},
  Flatten[{FullLinearModel[40, Rc, epsilon, a],
    FullLinearModel[334, Rc, epsilon, a], FullLinearModel[2702, Rc, epsilon, a]}]]

SetAttributes[FullLinearModel, Listable]
```

test

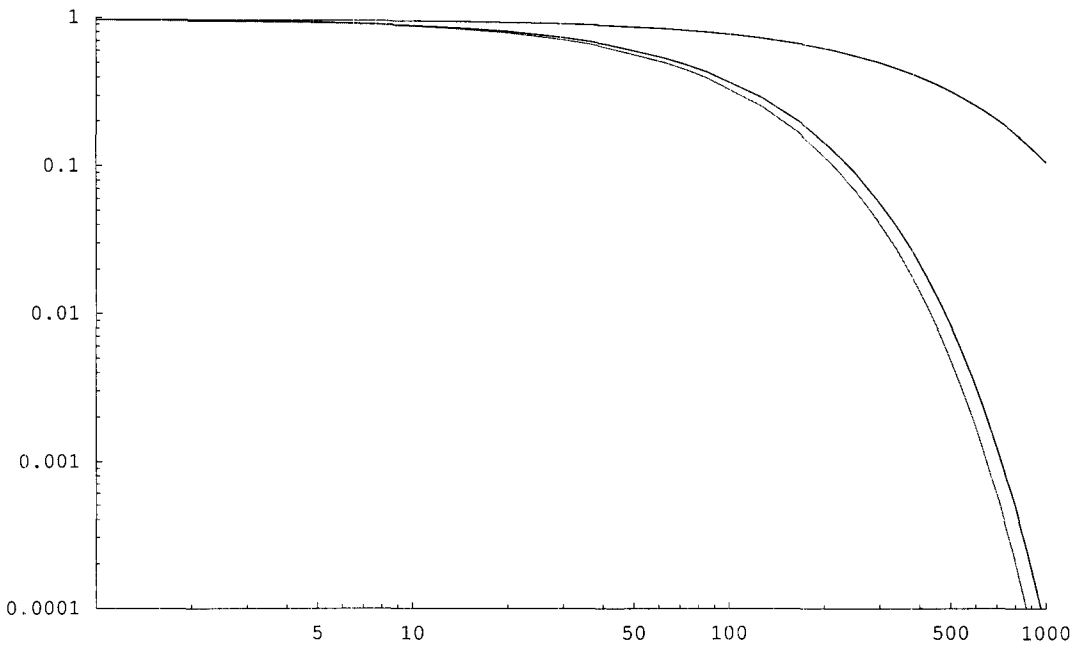
```
Rc = 20;
epsilon = 0.001;
a = 1000;
FullLinearModel[ 334, 20, 0.001, 1000]
FullLinearModel[Rc, epsilon, a]
```

```
0.698917

{0.918825, 0.698917, 0.0771798}
```

```
FullLinearModel[2, 20, 0., 1]
```

```
NDSolve::mxst : Maximum number of 5000 steps reached at the point t == 1469.5772270639875`.

0.895803
```

```
DisplayTogether[
 LogLogPlot[ FullLinearModel[ tau, 20, 0.01, 10],
  {tau, 0.1, 1000}, PlotRange → {{1, 1000}, {10^-4, 1}}],
 LogLogPlot[ FullLinearModel[ tau, 20, 0.01, 100], {tau, 0.1, 1000}],
 LogLogPlot[ FullLinearModel[ tau, 20, 0.01, Infinity],
  {tau, 0.1, 1000}, PlotStyle → RGBColor[1, 0, 0]] ]
```



- Graphics -

New table 2

This is Table 2 of Appendix A, but with full linear model instead of the abbreviated model.

```
Rclist = {1, 20, 100, 500};
Rclist = Flatten[Transpose[Table[ Rclist, {4}]]]
epslist = {0, 0.001, 0.01, 0.1};
epslist = Flatten[ Table[epslist, {4}]]
```

```
{1, 1, 1, 1, 20, 20, 20, 20, 100, 100, 100, 100, 500, 500, 500, 500}

{0, 0.001, 0.01, 0.1, 0, 0.001, 0.01, 0.1, 0, 0.001, 0.01, 0.1, 0, 0.001, 0.01, 0.1}
```

```
Partition[Flatten[ FullLinearModel[40, Rclist, epslist, 1000]], 4];
TableForm[%]
```

0.503401	0.493787	0.416653	0.108997
0.928295	0.918825	0.838101	0.344706
0.879177	0.871106	0.801769	0.35198
0.558424	0.553419	0.510357	0.227308

```
Partition[Flatten[ FullLinearModel[334, Rclist, epslist, 1000]], 4];
TableForm[%]
```

0.491755	0.403519	0.0705474	7.40382×10^{-8}
0.762178	0.698917	0.321318	0.000176671
0.365821	0.338647	0.169154	0.000172281
0.00780719	0.00724163	0.00368092	4.28205×10^{-6}

```
Partition[Flatten[ FullLinearModel[2702, Rclist, epslist, 1000]], 4];
TableForm[%]
```

0.40842	0.0796009	4.33884×10^{-8}	-6.45153×10^{-45}
0.155727	0.0771798	0.000142363	5.56987×10^{-31}
0.0003134	0.00016782	6.101×10^{-7}	3.74873×10^{-31}
9.03302×10^{-18}	4.91608×10^{-18}	2.06093×10^{-20}	6.01249×10^{-42}

Z versus colloid retardation factor: tau=334, different epsilons

```
result1 =
 Table[ Rval = 10^i; {Rval, FullLinearModel[334, Rval, 0, 1000]}, {i, 0, 3, 0.25}]
```

```
{{1, 0.491755}, {1.77828, 0.620393}, {3.16228, 0.722225},
 {5.62341, 0.784581}, {10., 0.802919}, {17.7828, 0.774267}, {31.6228, 0.692952},
 {56.2341, 0.553695}, {100., 0.365821}, {177.828, 0.173515},
 {316.228, 0.0458275}, {562.341, 0.00428233}, {1000., 0.0000631458}}
```

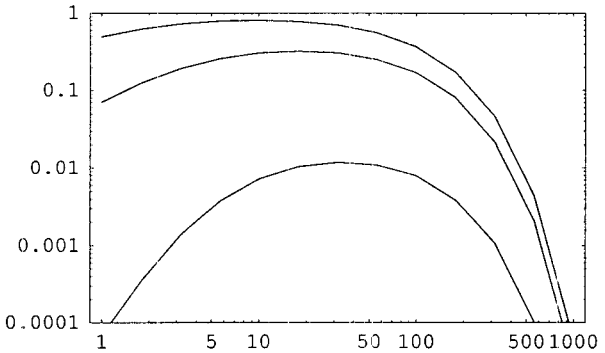
```
result3 =
 Table[ Rval = 10^i; {Rval, FullLinearModel[334, Rval, 0.01, 1000]}, {i, 0, 3, 0.25}]
```

```
{{1, 0.0705474}, {1.77828, 0.123834}, {3.16228, 0.191451},
 {5.62341, 0.257499}, {10., 0.304649}, {17.7828, 0.321966}, {31.6228, 0.304509},
 {56.2341, 0.251308}, {100., 0.169154}, {177.828, 0.0810878},
 {316.228, 0.0215453}, {562.341, 0.00202012}, {1000., 0.0000298451}}
```

```
result4 =
 Table[ Rval = 10^i; {Rval, FullLinearModel[334, Rval, 0.05, 1000]}, {i, 0, 3, 0.25}]
```

```
{{1, 0.0000680437}, {1.77828, 0.000353711}, {3.16228, 0.0013716},
 {5.62341, 0.00371466}, {10., 0.00713671}, {17.7828, 0.0103037}, {31.6228, 0.0117909},
 {56.2341, 0.0108907}, {100., 0.00782398}, {177.828, 0.00389289},
 {316.228, 0.00105645}, {562.341, 0.000100246}, {1000., 1.49105×10^-6}}
```

```
DisplayTogether[
 LogLogListPlot[ result1, PlotJoined → True],
 LogLogListPlot[result3, PlotJoined → True] ,
 LogLogListPlot[result4, PlotJoined → True] ,

  PlotRange → {Automatic, {-4, 0}}, Frame → True, Axes → False
]
```



- Graphics -

Z versus colloid retardation factor: tau=40, different epsilons

```
result1 = Table[ Rval = 10^i; {Rval, FullLinearModel[40, Rval, 0, 1000]}, {i, 0, 3, 0.25}]
```

```
{{1, 0.503401}, {1.77828, 0.641655}, {3.16228, 0.758387},
 {5.62341, 0.843392}, {10., 0.89728}, {17.7828, 0.9253}, {31.6228, 0.931875},
 {56.2341, 0.917814}, {100., 0.879177}, {177.828, 0.807068},
 {316.228, 0.689576}, {562.341, 0.51977}, {1000., 0.313892}}
```

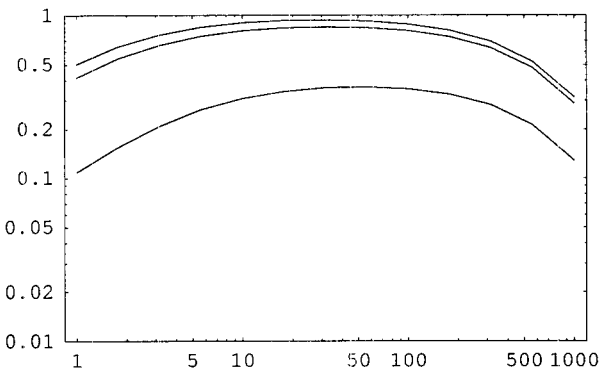
```
result3 =
 Table[ Rval = 10^i; {Rval, FullLinearModel[40, Rval, 0.01, 1000]}, {i, 0, 3, 0.25}]
```

```
{{1, 0.416653}, {1.77828, 0.542026}, {3.16228, 0.654917},
 {5.62341, 0.742456}, {10., 0.801256}, {17.7828, 0.83418}, {31.6228, 0.845097},
 {56.2341, 0.835289}, {100., 0.801769}, {177.828, 0.736874},
 {316.228, 0.630021}, {562.341, 0.47506}, {1000., 0.286952}}
```

```
result4 =
 Table[ Rval = 10^i; {Rval, FullLinearModel[40, Rval, 0.1, 1000]}, {i, 0, 3, 0.25}]
```

```
{{1, 0.108997}, {1.77828, 0.154431}, {3.16228, 0.208261},
 {5.62341, 0.262326}, {10., 0.307878}, {17.7828, 0.339942},
 {31.6228, 0.357667}, {56.2341, 0.361755}, {100., 0.35198}, {177.828, 0.326035},
 {316.228, 0.280007}, {562.341, 0.211673}, {1000., 0.128041}}
```

```
DisplayTogether[
 LogLogListPlot[ result1, PlotJoined → True],
 LogLogListPlot[result3, PlotJoined → True] ,
 LogLogListPlot[result4, PlotJoined → True] ,

  PlotRange → {Automatic, {-2, 0}}, Frame → True, Axes → False
]
```



- Graphics -

Figure 6:
Z versus colloid retardation factor: epsilon=0, a=1000, different taus

```
result1 = Table[ Rval = 10^i; {Rval, FullLinearModel[40, Rval, 0, 1000]}, {i, 0, 3, 0.25}]
```

General::spell1 : Possible spelling error: new symbol name "Rval" is similar to existing symbol "Real".

```
{{1, 0.503401}, {1.77828, 0.641655}, {3.16228, 0.758387},
 {5.62341, 0.843392}, {10., 0.89728}, {17.7828, 0.9253}, {31.6228, 0.931875},
 {56.2341, 0.917814}, {100., 0.879177}, {177.828, 0.807068},
 {316.228, 0.689576}, {562.341, 0.51977}, {1000., 0.313892}}
```

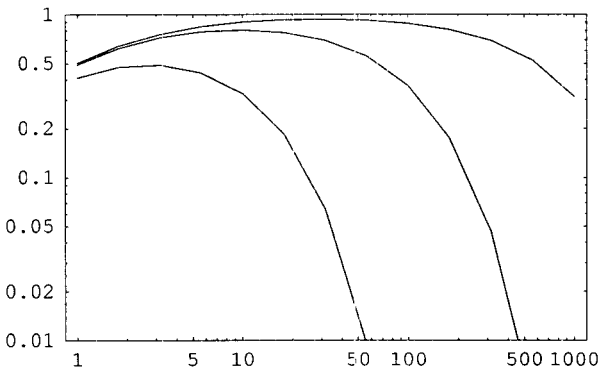
```
result3 =
 Table[ Rval = 10^i; {Rval, FullLinearModel[334, Rval, 0.0, 1000]}, {i, 0, 3, 0.25}]
```

```
{{1, 0.491755}, {1.77828, 0.620393}, {3.16228, 0.722225},
 {5.62341, 0.784581}, {10., 0.802919}, {17.7828, 0.774267}, {31.6228, 0.692952},
 {56.2341, 0.553695}, {100., 0.365821}, {177.828, 0.173515},
 {316.228, 0.0458275}, {562.341, 0.00428233}, {1000., 0.0000631458}}
```

```
result4 =
 Table[ Rval = 10^i; {Rval, FullLinearModel[2702, Rval, 0., 1000]}, {i, 0, 3, 0.25}]
```

$$\{\{1, 0.40842\}, \{1.77828, 0.472961\}, \{3.16228, 0.487273\},$$
$$\{5.62341, 0.438331\}, \{10., 0.328093\}, \{17.7828, 0.184309\}, \{31.6228, 0.0637471\},$$
$$\{56.2341, 0.00945063\}, \{100., 0.0003134\}, \{177.828, 7.28532\times10^{-7}\},$$
$$\{316.228, 1.50462\times10^{-11}\}, \{562.341, 7.00194\times10^{-20}\}, \{1000., 1.06585\times10^{-34}\}\}$$

```
DisplayTogether[
 LogLogListPlot[ result1, PlotJoined → True],
 LogLogListPlot[result3, PlotJoined → True] ,
 LogLogListPlot[result4, PlotJoined → True] ,

  PlotRange → {Automatic, {-2, 0}}, Frame → True, Axes → False
]
```



- Graphics -

Z versus Rval,; tau=334; epsilon=0.01; different reversibility ratios

```
result1 =
 Table[ Rval = 10^i; {Rval, FullLinearModel[334, Rval, 0.01, Infinity]}, {i, 0, 3, 0.25}]
```

{{1, 0.0186491}, {1.77828, 0.0232875}, {3.16228, 0.0268807},
{5.62341, 0.0290213}, {10., 0.0295776}, {17.7828, 0.0284479}, {31.6228, 0.0254186},
{56.2341, 0.020288}, {100., 0.0133917}, {177.828, 0.00634537},
{316.228, 0.00167339}, {562.341, 0.000155984}, {1000., 2.2903×10^{-6}}}

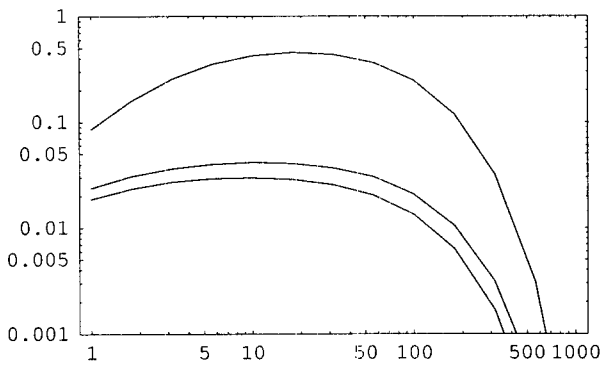
```
result3 =
 Table[ Rval = 10^i; {Rval, FullLinearModel[334, Rval, 0.01, 100]}, {i, 0, 3, 0.25}]
```

{{1, 0.0857058}, {1.77828, 0.157544}, {3.16228, 0.253453},
{5.62341, 0.351374}, {10., 0.424526}, {17.7828, 0.454876}, {31.6228, 0.434154},
{56.2341, 0.360736}, {100., 0.244379}, {177.828, 0.118115},
{316.228, 0.0317864}, {562.341, 0.00304552}, {1000., 0.0000467307}}

```
result4 =
 Table[ Rval = 10^i; {Rval, FullLinearModel[334, Rval, 0.01, 10]}, {i, 0, 3, 0.25}]
```

{{1, 0.0236241}, {1.77828, 0.0305089}, {3.16228, 0.0362326},
{5.62341, 0.0399864}, {10., 0.0414484}, {17.7828, 0.0404655},
{31.6228, 0.0367751}, {56.2341, 0.030077}, {100., 0.0206635}, {177.828, 0.0104922},
{316.228, 0.00312571}, {562.341, 0.000361677}, {1000., 7.79692×10^{-6}}}

```
DisplayTogether[
 LogLogListPlot[ result1, PlotJoined → True],
 LogLogListPlot[result3, PlotJoined → True] ,
 LogLogListPlot[result4, PlotJoined → True] ,

  PlotRange → {Automatic, {-3, 0}}, Frame → True, Axes → False
]
```



- Graphics -

Figure 7:
Z versus epsilon: tau=334; different reversibility factors; Rc=20

```
result1 =
 Table[ eval = 10^i; {eval, FullLinearModel[334, 20, eval, Infinity]}, {i, -3, 0, 0.25}]
```

$\{\{\frac{1}{1000}, 0.563201\}, \{0.00177828, 0.434442\},$
$\{0.00316228, 0.273819\}, \{0.00562341, 0.120497\}, \{0.01, 0.0279928\},$
$\{0.0177828, 0.00208805\}, \{0.0316228, 0.0000206582\},$
$\{0.0562341, 5.62752\times10^{-9}\}, \{0.1, 2.57795\times10^{-15}\}, \{0.177828, 1.37835\times10^{-26}\},$
$\{0.316228, -4.47564\times10^{-46}\}, \{0.562341, 4.07004\times10^{-46}\}, \{1., 5.42618\times10^{-45}\}\}$

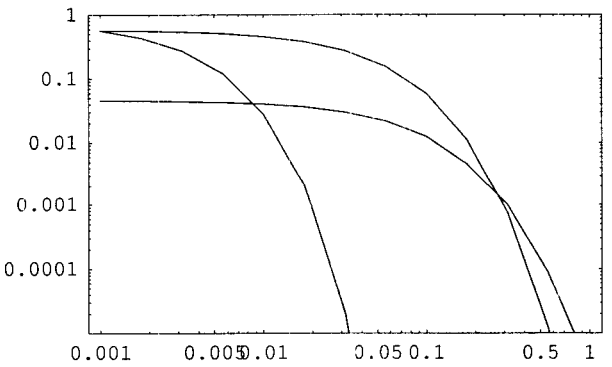
```
result2 =
 Table[ eval = 10^i; {eval, FullLinearModel[334, 20, eval, 100]}, {i, -3, 0, 0.25}]
```

$\{\{\frac{1}{1000}, 0.564428\}, \{0.00177828, 0.553963\}, \{0.00316228, 0.535847\},$
$\{0.00562341, 0.505133\}, \{0.01, 0.454941\}, \{0.0177828, 0.378058\},$
$\{0.0316228, 0.272853\}, \{0.0562341, 0.15425\}, \{0.1, 0.0576044\}, \{0.177828, 0.0109072\},$
$\{0.316228, 0.000726287\}, \{0.562341, 0.0000114414\}, \{1., 3.49391\times10^{-8}\}\}$

```
result3 =
 Table[ eval = 10^i; {eval, FullLinearModel[334, 20, eval, 10]}, {i, -3, 0, 0.25}]
```

$\{\{\frac{1}{1000}, 0.0452239\}, \{0.00177828, 0.0447395\}, \{0.00316228, 0.0438911\},$
$\{0.00562341, 0.0424243\}, \{0.01, 0.0399434\}, \{0.0177828, 0.0359039\},$
$\{0.0316228, 0.0297536\}, \{0.0562341, 0.0214151\}, \{0.1, 0.0121273\}, \{0.177828, 0.0046299\},$
$\{0.316228, 0.000959442\}, \{0.562341, 0.0000845834\}, \{1., 2.74102\times10^{-6}\}\}$

```
DisplayTogether[
 LogLogListPlot[ result1, PlotJoined → True],
 LogLogListPlot[ result2, PlotJoined → True],
 LogLogListPlot[ result3, PlotJoined → True],

  PlotRange → {Automatic, {-5, 0}}, Frame → True, Axes → False
]
```



- Graphics -

ECP 1/18/2010

SCIENTIFIC NOTEBOOK

by

Scott Painter

Southwest Research Institute
Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

May 25, 2004

Table of Contents

INITIAL ENTRIES....3
Radionuclide Transport Analysis for Yucca Mountain....4
Objective:....4
6.1.04 Process-level modeling of colloid facilitated transport in the saturated zone....5
6.01.04 Supporting material for previous entry....30

INITIAL ENTRIES

Scientific notebook: #318E Vol. 9
Issued to: S. L. Painter
Original Issue Date: 22-March-1999

By agreement with CNWRA QA this notebook will be printed at approximate quarterly intervals. This computerized electronic notebook is intended to address the criteria of CNWRA QAP-001.

Radionuclide Transport Analysis for Yucca Mountain

Account number: 20-1402-871
Description: Radionuclide Transport Analysis.
Collaborators: Drs. D. Turner, D. Pickett, and V. Cvetkovic (consultant)

Objective:
Development and testing of transport analysis methodology for spatially variable systems including demonstrations and applications to the proposed repository at Yucca Mountain.

SCP 1/18/2010

6.1.04 Process-level modeling of colloid facilitated transport in the saturated zone

1 Introduction

1.1 Background

Radionuclides that sorb strongly to minerals in the subsurface also tend to have strong affinity for the naturally occurring mineral colloids that are present in small concentrations within groundwater. In the absence of colloids, strongly sorbing radionuclides are relatively immobile. Once attached to colloids, however, they have reduced interactions with the porous medium and may move relatively unretarded through the subsurface. Indeed, colloids have long been recognized as a potentially important factor in enhancing radionuclide migration in the subsurface (Buddenmeir and Hunt, 1988; McCarthy and Zachara, 1989; Penrose et al., 1990). Of particular interest is the fact that colloid-bound plutonium at the Nevada Test Site appears to have traveled hundreds of meters in less than 30 years (Kersting et al., 1999).

The Department of Energy (DOE) has taken steps to include colloid-facilitated transport in performance assessment calculations (BSC, 2003a,b). Multidimensional process modeling and a limited number of experiments (Lu et al., 1998, 2000) have been undertaken to support the abstraction, which presumes either irreversible attachment to colloids or equilibrium partitioning between the colloid-bound, aqueous, and sorbed states. Similar abstractions are being prepared for the TPA 5.0 code (Pickett, 2003).

Although several studies have compared colloid-facilitated transport models with experiments at the column scale (Saiers and Hornberger, 1996; van de Weerd and Leijnse, 1997; Luhrmann and Noseck, 1998; Noell et al., 1998), understanding of colloid-facilitated transport at the field scale is less developed. In particular, the most significant parameters controlling field-scale transport and even which processes that need to be included in a field-scale model remain to be fully defined. Cvetkovic et al. (2004) made a preliminary assessment of parameter and model sensitivities for field-scale colloid-facilitated transport in the saturated zone. That study identified kinetic effects (in particular radionuclide desorption) and colloid filtration as key processes. However, the Cvetkovic et al. (2004) study did not consider matrix diffusion effects and is thus not useful for understanding the fractured tuff segment of the transport path. Moreover, the focus was on steady-state throughput without consideration of transient effects. Such a steady-state analysis is useful as a first step for understanding parameter and model sensitivities, but cannot accommodate a finite compliance period.

This report extends the previous work of Cvetkovic et al. (2004) to include transient solutions, the effects of matrix diffusion, and probabilistic analyses considering key parameter uncertainties. Both the fractured tuff and alluvial segments of the saturated zone transport pathway are considered. The broad objective is to assess potential effects of colloid facilitated transport on the barrier capabilities of the saturated zone, with particular emphasis on understanding sensitivities to model parameters and modeling approaches.

1.2 Risk Significance

The Risk Insights Baseline Report (NRC, 2004) concludes that colloid facilitated transport in the saturated zone is of medium significance to waste isolation based on the potential for colloids to enhance transport of the strongly sorbing actinides americium and plutonium. These two elements comprise 98% of the radiological inventory at 1000 years (NRC, 2004), based on

activity. Performance effects of colloid-facilitated transport have not yet been assessed with the TPA 5.0 code, but an earlier version of the TPA code was used to bound the effect by assigning no retardation to americium, plutonium, and thorium. That conservative analysis increased dose at 10,000 years by nearly two orders of magnitude over the base case. In addition, sorption in the alluvium – a process potentially affected by the presence of colloids – was concluded to be of high significance to waste isolation. The Risk Insights Baseline Report also recognizes that colloid transport in the geosphere is subject to significant uncertainties, and recommends that additional analyses be undertaken to identify more realistic approaches for colloid-facilitated transport.

DOE analyses also identified colloid-facilitated transport as a potentially important contributor to dose. In the TSPA-SR, colloidal plutonium is the second highest contributor to dose after 70,000 years (CRWMS M&O, 2000).

1.3 Objectives

The broad objective is to assess potential effects of colloid-facilitated transport on the barrier capabilities of the saturated zone. The specific objectives are to:

1. Develop more realistic process models that can be used in confirmatory analyses and in support of TPA model abstractions.

2. Identify key processes and parameters affecting colloid-facilitated transport in the saturated zone.

3. Assess the potential for colloids to degrade saturated zone barrier performance.

1.4 Scope

The potential for naturally occurring colloids to enhance transport in the saturated zone is addressed in this report. Transport in the unsaturated zone and engineered barriers is not considered, nor is the effect of anthropogenic colloids produced from degradation of the waste-form or engineered barriers. The focus is on performance of the fractured tuff and alluvial aquifers as transport barriers. Plutonium 239 is used as a representative radionuclide. An idealized release scenario involving a constant rate of release initiated at t = 0 is considered, and normalized breakthrough of plutonium for the time period t= 0 to 10,000 years is monitored. The effects of uncertainties in key transport parameters are included through Monte Carlo simulations.

2 Transport Model

2.1 Transport Scenario

The transport scenario considered here is identical to that used in the TPA code (Mohanty et al., 2002) for the saturated zone. Specifically, transport is conceptualized as occurring along a one-dimensional streamtube or flowpath that starts in the saturated zone directly below the repository and travels through the fractured tuff and alluvial aquifers to pumping locations south of the repository. The TPA central streamtube is considered. The flow rate in the central streamtube is fixed at 252 m^3/day per meter of aquifer thickness. The width of the streamtube varies along the streamtube trajectory. Travel distance is fixed at 18 km, but the proportion of the flowpath in the tuff and alluvium is uncertain because of uncertainty in the position of the tuff/alluvium contact. Uncertainties in the tuff and alluvium porosities introduce additional uncertainties in the total travel time along the path.

The groundwater is assumed to also contain naturally occurring colloids. Colloids are advected with the flowing water, and are subject to longitudinal dispersion. Downstream movement of colloids is slowed by reversible sorption, which is modeled as an equilibrium process. Colloids may also be permanently removed due to physical filtration processes. Colloid concentration is assumed to be constant in time and space.

Within the streamtube, radionuclides in solution are advected by the moving water and are also subject to longitudinal dispersion. Within the alluvium, downstream movement is slowed by equilibrium sorption on the mineral grains. Within the tuff segment of the path, sorption occurs only in the matrix. Radionuclides must diffuse from the fractures to the matrix before they can access the sorption sites. This diffusive mass transfer is modeled as a first-order kinetic process (mobile-immobile model).

Radionuclides may also attach to colloids, thereby facilitating radionuclide transport. Radionuclide attachment to colloids is modeled as a kinetic process with forward (sorption) and reverse (desorption) rates. Because the colloids themselves may exist in up to three states (mobile, temporarily immobilized or sorbed, and permanently immobilized or filtered), mass exchanges between solution and several colloid-bound states are considered. These various exchanges along with associated rate constants are shown in Figure 2-1. By proper specification of the various rate constants, the kinetic model includes the entire range of sorption behaviors. Irreversible sorption is obtained by setting reverse rates to zero, while equilibrium sorption is obtained by letting the forward and reverse rates go to infinity.

2.2 One-dimensional Radionuclide Transport Equations

The conservation equations for radionuclides can be expressed mathematically as

$$\frac{\partial C}{\partial t} + \tau C = -\psi^{C\to S}(C,S) - \psi^{C\to S^*}(C,S^*) - \psi^{C\to S^{**}}(C,S^{**}) - \psi^{C\to C^*}(C,C^*) - \lambda C$$

$$\frac{\partial S}{\partial t} + \tau S = \psi^{C\to S}(C,S) - \psi^{S\to S^*}(S,S^*) - \psi^{S\to S^{**}}(S,S^{**}) - \lambda S$$

$$\frac{\partial C^*}{\partial t} = \psi^{C\to C^*}(C,C^*) - \lambda C^*$$

$$\frac{\partial S^*}{\partial t} = \psi^{S\to S^*}(S,S^*) + \psi^{C\to S^*}(C,S^*) - \lambda S^*$$

$$\frac{\partial S^{**}}{\partial t} = \psi^{S\to S^{**}}(S,S^{**}) + \psi^{C\to S^{**}}(C,S^{**}) - \lambda S^{**}$$

where C is the concentration of radionuclides in solution, C^* is the concentration of immobile radionuclides, S is the concentration of radionuclides attached to mobile colloids, S^* is the concentration of radionuclides attached to temporarily immobilized colloids, and S^{**} is the concentration of radionuclides attached to permanently immobilized colloids. All concentrations are defined on a bulk aquifer volume basis. The ψ terms represent the various exchanges. The differential operator T is the transport operator representing advection and dispersion processes.

Taking into account that the temporarily immobilized colloids attach and detach quickly from the porous matrix and can be modeled with an equilibrium model $S^*=K_c\,S$, the transport equations become

$$\frac{\partial C}{\partial t} + \mathcal{T}C = -\psi^{C\to S}(C,S) - \psi^{C\to S^*}(C,K_C S) - \psi^{C\to S^{**}}(C,S^{**}) - \psi^{C\to C^*}(C,C^*) - \lambda C$$

$$R_C\frac{\partial S}{\partial t} + \mathcal{T}S = \psi^{C\to S}(C,S) + \psi^{C\to S^*}(C,K_C S) - \psi^{S\to S^{**}}(S,S^{**}) - \lambda R_C\,S$$

$$\frac{\partial C^*}{\partial t} = \psi^{C\to C^*}(C,C^*) - \lambda C^*$$

$$\frac{\partial S^{**}}{\partial t} = \psi^{S\to S^{**}}(S,S^{**}) + \psi^{C\to S^{**}}(C,S^{**}) - \lambda S^{**}$$

where K_c is the dimensionless distribution coefficient for colloids on porous matrix and $R_c=K_c-1$ is the colloid retardation factor.

The various exchange terms in Eq. (2-2) can be linear or non-linear, equilibrium or kinetically controlled. Linear models are generally applicable when the sorption capacity is large relative to the local aqueous concentration. Cvetkovic et al. (2004) evaluated linear and nonlinear models for sorption on colloids, and concluded that conditions expected in the Yucca Mountain saturated zone are well within the range of validity for linear models. Assuming linear exchanges,

$$\psi^{C\to C^*}(C,C^*) = k_f C - k_r C^*$$

$$\psi^{C\to S}(C,S) = \alpha_f C - \alpha_r S$$

$$\psi^{C\to S^*}(C,S^*) = \alpha_f^* C - \alpha_r^* S^*$$

$$\psi^{C\to S^{**}}(C,S^{**}) = -\alpha_r^{**} S^{**}$$

$$\psi^{S\to S^{**}}(S,S^{**}) = \varepsilon S$$

All of these exchanges are bidirectional except for the last two, which are unidirectional (irreversible). The *S->S*** exchange is unidirectional by definition because it applies to permanent removal. No forward rate is included in the *C->S*** exchange because a permanently removed colloid is indistinguishable from the porous matrix, and that exchange is presumably already counted in the any measured rates for C->C* exchange.

Two additional assumptions help simplify the set of exchanges. First, the rate coefficients for desorption from colloids are assumed to be identical irrespective of whether the colloid is mobile or immobile. Thus, $\alpha_r = \alpha_r^* = \alpha_r^{**}$. Second, the forward rate for sorption on colloids should be proportional to the concentration of sorption sites, and thus proportional to colloid concentration. This assumption implies $\alpha_f^* = K_C \alpha_f$.

The transport model then becomes

$$\frac{\partial C}{\partial t} + \mathcal{T} C = -k_f C + k_r C^* - \alpha_f R_C C + \alpha_r R_C S + \alpha_r S^{**} - \lambda C$$

$$R_C \frac{\partial S}{\partial t} + \mathcal{T} S = \alpha_f R_C C - \alpha_r R_C S - \varepsilon S - \lambda R_C S$$

$$\frac{\partial C^*}{\partial t} = k_f C - k_r C^* - \lambda C^*$$

$$\frac{\partial S^{**}}{\partial t} = \varepsilon S - \alpha_r S^{**} - \lambda S^{**}$$

The differential operator T in Eq. (2-4) represents the transport processes. In the familiar Eulerian form, the operator is

$$\mathcal{T} = v\frac{\partial}{\partial x} - D\frac{\partial^2}{\partial x^2}$$

where $D = a|v|$ is the longitudinal dispersion coefficient, v is velocity, a is the dispersivity, and x is position along the streamtube. Writing the dispersivity as a fixed fraction f of the travel distance L,

$$\mathcal{T} = v\frac{\partial}{\partial x} - f\,L\,|v|\frac{\partial^2}{\partial x^2}$$

Note that both v and L are uncertain. Uncertainty in L comes from uncertainty in the tuff-alluvium contact, while uncertainty in v comes from uncertainty in porosity and possibly total flow rate in the streamtube. These two uncertain parameters can be collapsed to a single uncertain parameter by re-writing the transport operator in a Lagrangian form. Switching independent variables from position x to travel time $\tau = x/v$, yields the Lagrangian form of the transport operator

$$\mathcal{T} = \frac{\partial}{\partial \tau} - f\,\tau_L\frac{\partial^2}{\partial \tau^2}$$

where $\tau_L = L/v$ is the global travel time. With this form of the transport operator, the concentrations are functions of τ and time, and uncertainties in travel distance and velocity are manifest through the uncertainty in the total travel time $\tau_L = L/v$.

The system Eq. (2-4) with the transport operator Eq. (2-7) is the general model used for this study. With appropriate choice of the various exchanges it represents the transport effects of equilibrium, slowly reversible, or irreversible sorption to colloids for the tuff and alluvial aquifers. For example, if the C->C* exchange is modeled as a fast exchange then the radionuclide sorption (on the matrix) becomes the equilibrium model. If the C->C* exchange is modeled with a first-order kinetic model then the mobile-immobile model used to represent matrix diffusion can be recovered.

2.3 Transport Equations for the Alluvial Aquifer

For the alluvial aquifer, the C->C* exchange is assumed to be fast compared to other rates and transport times of the system. This implies, $C^* = K_d'$ where $K_d' = k_f/k_r$ is the dimensionless *distribution* coefficient. The transport model then becomes

$$R\frac{\partial C}{\partial t} + \mathcal{T}C = -\alpha_f R_C C + \alpha_r R_C S + \alpha_r S^{**} - \lambda RC$$

$$R_C\frac{\partial S}{\partial t} + \mathcal{T}S = \alpha_f R_C C - \alpha_r R_C S - \varepsilon S - \lambda R_C S$$

$$\frac{\partial S^{**}}{\partial t} = \varepsilon S - \alpha_r S^{**} - \lambda S^{**}$$

where $R = 1 + K_d'$ is the retardation factor.

2.4 Transport Equations for the Tuff Aquifer

In the fractured tuff part of the flowpath, sorption of radionuclides on the fracture surfaces is neglected, consistent with the abstractions in the TPA code (Mohanty et al., 2002). Radionuclides must first diffuse from the fractures into the matrix before they can access sorption sites. This process can be modeled with a mobile-immobile approximation.

To obtain the mobile-immobile model from Eq. (2-4), the immobile concentration is divided into concentration in stagnant water regions and concentration sorbed on the porous matrix. Assuming equilibrium partitioning between the two immobile states, $C^* = R_S C'$, where C' is the radionuclide concentration in immobile water, $R_S = 1 + \frac{1-\theta_{im}}{\theta_{im}} \rho_b K_S$ is the matrix retardation factor, K_s is the distribution coefficient, ρ_b is bulk density, and θ_{im} is the matrix porosity. Defining the forward rate as $k_f = \beta_0 \omega$, and the reverse rate as $k_r = \beta_0 / R_S$, with $\omega = \theta_{im} / \theta_m$ and $\beta_0 = \frac{D}{0.28}\left(\frac{2n}{f}\right)^2$ where D is the diffusion coefficient, n is the number of fractures per unit meter, f is the fraction of the matrix porosity that is participating the diffusion process, the following transport equations are obtained

$$\frac{\partial C}{\partial t} + \tau C = -\beta_0(\omega C - C') - \alpha_f R_C C + \alpha_r R_C S + \alpha_r S^{**} - \lambda C$$

$$R_C \frac{\partial S}{\partial t} + \tau S = \alpha_f R_C C - \alpha_r R_C S - \varepsilon S - \lambda R_C S$$

$$R_S \frac{\partial C'}{\partial t} = \beta_0(\omega C - C') - \lambda R_S C'$$

$$\frac{\partial S^{**}}{\partial t} = \varepsilon S - \alpha_r S^{**} - \lambda S^{**}$$

In the absence of colloids, these equations are identical to those solved in the flow and transport module within TPA (Mohanty, et al., 2002).

3 Generic Model and Parameter Sensitivities

Cvetkovic et al. (2004) explored generic model and parameter sensitivities related to the transport model given in Eq. (2-4). They considered a unit step input of plutonium released into the streamtube starting at t=0. Because the focus was on broad-scope sensitivities, they ignored transients and monitored the steady-state discharge as $t \rightarrow \infty$. At steady-state, the plutonium discharge is smaller in magnitude than the plutonium input at the start of the streamtube because of radionuclide decay. The magnitude of the steady-state throughput provides a convenient measure of performance of the saturated zone as a barrier. Sorption on colloids was modeled in a kinetic framework, using the linear, bi-linear and Langmuir sorption models. Sensitivities to the key controlling parameters were identified.

The main conclusions on model and parameter sensitivities are (Cvetkovic et al., 2004):

•Under some conditions, colloid facilitated transport can enhance the throughput of plutonium by many orders of magnitude, depending on the travel time, the rate of desorption from colloids, and the sorption capacity (See Figures 3-1 and 3-2).

•Nonlinear models for sorption on colloids generally predict smaller throughput of plutonium, but for typical field conditions, the differences between the linear and nonlinear models are relatively minor; thus, the linear model provides a useful estimate.

•Temporary immobilization (retardation) of colloids is relatively ineffective at reducing the plutonium throughput, unless the retardation factor is large (> 100). In fact, the throughput of plutonium depends non-monotonically on the colloid retardation factor R_c, first rising with increasing R_c and then falling with increasing R_c for large R_c (see Figure 3-3). This non-monotonic behavior is a consequence of mass exchange between solution and temporarily immobilized colloids. However, this finding of relative insensitivity to colloid retardation is based on an asymptotic analysis $t \rightarrow \infty$. Consideration of a finite time window may change these sensitivities.

•Permanent immobilization of colloids is generally more effective at reducing plutonium throughput. However, with permanent immobilization of colloids, the throughput becomes very sensitive to the rate of plutonium desorption from colloids (Figure 3-4). If permanent immobilization is included, then irreversible binding to colloids is not a conservative assumption.

4. Model Parameters for Colloid Facilitated Transport

Parameter values and probability distributions required for the analysis were selected to be consistent with the TPA 5.0 base case input, where possible. Parameter values are summarized in Table 4-1. The technical bases for these parameters are discussed in the following subsections.

4.1 Radionuclide Transport Parameters

We consider transport of plutonium 239 in the form Pu(V) as a representative radionuclide. The relevant radionuclide transport parameters include the groundwater travel time (τ) distributions for the tuff and alluvium, retardation factors for the fractured tuff and alluvium, and matrix diffusion parameters for the fractured tuff aquifer.

The travel time distributions were calculated as in the TPA 5.0 code. The distance to the tuff-alluvium contact was first sampled. Using this sampled value, an average streamtube width was calculated by integrating along the variable-width streamtube. Only the central streamtube is considered here. Transport porosity is then sampled. Given the travel distance sample, the porosity sample, the calculated average width, and a fixed flow rate in the streamtube, a travel time is then calculated. This represents one sample from the travel time distribution. Note that the tuff and alluvium travel times are correlated because once the travel distance in the tuff is sampled, the travel distance in the alluvium is then fixed. The resulting travel time distributions are shown in Figure 4-1.

Radionuclide retardation factors in the TPA 5.0 basecase are not input directly, but are calculated internally based on sampled water chemistry parameters. For this reason, the TPA 4.1 distributions of R_d are used instead for the alluvium. These distributions are similar to the TPA 5.0 distributions. Once a R_d value is sampled, the same value is used for the immobile retardation Rs in the tuff matrix. Within the TPA 5.0 code, the two retardation factors are perfectly correlated and differ only by a multiplicative factor corresponding to the ratio of specific surface areas. This factor is a relatively minor correction and is ignored.

The matrix diffusion parameters D, f, n, and θ_{im} are identical to those of the TPA 5.0 base case.

4.2 Colloid Transport Parameters

Colloid transport parameters include colloid retardation in tuff and alluvium, and the irreversible removal rate for colloids ε.

DOE (BSC, 2003) compiled results from laboratory and field experiments on colloid retardation in alluvial material and in fractured volcanics and used these to generate probability distributions for use in performance assessment. Some of the retardation factors were calculated from breakthrough curves from short-duration column experiments by numerical parameter estimation. The colloid retardation factors for some of these experiments were nominally estimated to be small; however, the uncertainty in the estimated parameter is so large as to make the estimate indeterminant. For this reason, DOE truncated the distributions at $R_c = 6$ for the tuff and $R_c = 8$ for alluvium (see Figure 4-2). DOE's truncated distributions are used here. In addition, we also include, as a sensitivity case, the untruncated distributions.
In considering permanent removal rates, it is important to make the distinction between naturally

occurring colloids and anthropogenic colloids such as corrosion products. The latter are much more likely to be permanently removed due to filtration. It is also important to realize that "permanent removal" means immobilization for transport time scales of interest. For natural occurring colloids, no evidence exists for colloid immobilization for the time scales of interest (hundreds to thousands of years). For this reason, the conservative assumption of no irreversible colloid filtration is used (ε=0).

4.3 Parameters for Radionuclide Sorption onto Colloids

The experiments of Lu et al. (1998, 2000) provide data on the rates for radionuclide sorption and desorption on colloids. They considered 239Pu and 243Am, with the plutonium being prepared in both colloidal Pu(IV) and soluble Pu(V) forms. Strong sorption and relatively weak desorption on hematite, goethite, montmorillonite, smectite, and silica colloids was observed. Americium and plutonium sorption showed similar trends and similar magnitudes for the sorption parameters.

Painter et al. (2002) re-analyzed the Lu et al. (2000) data. They pointed out that the data clearly show evidence for two forward sorption rates, and that the slower rate is the one that is relevant for field scale applications. Cvetkovic et al. (2004) assumed a linear relationship between the forward sorption rate and the colloid concentration, and scaled forward rates estimated from the Lu et al. data to field-relevant colloid concentrations. They estimate forward sorption rates of about 0.1 yr^{-1}. This value is used here.

Lu et al. (1998, 2000) also performed desorption experiments. In the Lu et al. (1998) experiments, 1% of Pu(V) desorbed from smectite colloids after 150 days. For montmorillonite and silica, the fraction desorbed was about 0.5% and 1%, respectively. Desorption from hematite and goethite colloids was even slower. In the Lu et al. (2000) experiments desorption proceeded at a faster rate, with 17-21% of Pu(V) desorbing from montmorillonite and 17-20% desorbing from silica colloids in 295 days.

Clearly desorption rates are slow and highly uncertain. We define an reversibility ratio $a = \alpha_f / \alpha_r$ and treat it as a sensitivity parameter. Small values of a indicate strongly reversible sorption and large values indicate weakly reversible. The 0.5-1% desorbed fraction observed in the 150 day Lu et al. (1998) experiments implies a desorption rate of 0.01-0.02 yr^{-1}, which corresponds to a reversibility ratio of 5-10. The reversibility factor a is varied in the range 5 to infinity.

5.0 Potential Effects of Colloids on Saturated Zone Barrier Performance

5.1 Fractured Tuff Aquifer

Normalized breakthrough curves at the tuff-alluvium interface are shown in Figure 5-1 for different values of the reversibility ratio *a*. In these simulations, the fractured tuff aquifer is initially free of radionuclides. At the beginning of the simulation, the inlet concentration is set to a nonzero value, which is held constant throughout the simulation. The breakthrough curves are normalized by the inlet concentration and are averages over 300 realizations.

The case *a*=5 corresponds to a desorption rate of 0.02 yr^{-1}, which is in the range inferred from multi-step desorption experiments (Lu et al., 1998) involving $^{239}Pu(V)$ sorbed to smectite. Although slow on the time scale of laboratory experiments, such a rate represents relatively reversible attachment of radionuclides to colloids on transport time scales of 10,000 years. For the *a*=5 situation, the breakthrough is about 0.1% at 10,000 years; the fractured tuff aquifer has attenuated the radionuclide discharge by a factor of 1000. In the situation of no colloids, the breakthrough is negligible in 10,000 years because of matrix diffusion and strong plutonium sorption on the tuff matrix. Thus, colloid facilitated transport is the dominant transport mechanism for plutonium in the fractured tuff aquifer, even for relatively rapid desorption.

The calculated breakthroughs are very sensitive to the reversibility ratio (or desorption rate). Increasing *a* to 50 increases the 10,000 year breakthrough to about 6%. For *a*=500, the 10,000 year breakthrough is about 20% and the 20,000 year breakthrough is almost 40%. For this combination of input parameters, the situation of *a*=500 is close to the irreversible limit; increasing *a* beyond 500 results in no significant increase in breakthrough.

5.2 Alluvial Aquifer

Similar results are shown in Figure 5-2 for the alluvial aquifer. For the alluvium, the no-colloids breakthrough is less than 0.1% at 10,000 years. With colloids and a reversibility factor of *a*=5, the 10,000 year breakthrough is increased to about 3%. Thus, colloid-facilitated transport is also the dominant transport mechanism in the alluvial aquifer. The breakthrough is less sensitive to the reversibility ratio, as compared with tuff aquifer. For nearly irreversible sorption (*a*=5000), the breakthrough is about 18% at 10,000 years.

5.3 Saturated Zone

Results for the entire saturated zone are shown in Figure 5-3. These simulations also use a step function input into the fractured tuff aquifer. For each realization, the calculated breakthrough at the tuff-alluvium interface is then used as the input for the alluvial aquifer. As in Figures 5-1 and 5-2, the breakthrough curves are mean curves averaged over multiple realizations (150 in this case).

For the reversible case with *a*=5, the 10,000 year breakthrough concentration is about 10^{-7}, indicating that the saturated zone attenuates the radionuclide concentrations by a large amount. For the case *a*=5000 (nearly irreversible), the 10,000 year breakthrough is roughly 10%. The 10,000 year breakthrough is similar in magnitude to that calculated by DOE for irreversibly attached colloids. However, the early breakthrough in Figure 5-3 is significantly larger than that the DOE calculations. The differences are due to differences in the travel time distributions.

Figure 5-4 shows results for simulations similar to those of Figure 5-3, but with a irreversible filtration rate of 0.05 yr^{-1}. This relative small value of filtration implies a half-life of about 14 years for a mobile colloid. Irreversible filtration reduces the breakthrough for all values of the desorption rate (reversibility ratio), but the effect is most dramatic for the cases of nearly irreversible attachment. The breakthrough becomes relatively insensitive to the reversibility ratio for a > 50. This relative insensitivity is caused by desorption from immobilized colloids, which allows plutonium become mobile again.

SP 1/15/2010

6. Discussion

Based on the results presented here and in previous sensitivity studies (Painter et al., 2001; Cvetkovic et al., 2004), colloid-facilitated transport is likely to be the dominant transport mechanism in the saturated zone for strongly sorbing radionuclides like plutonium. The magnitude of the effect is most sensitive to two uncertain parameters: the rate at which colloids are irreversibly filtered from the system, and the rate of radionuclide desorption from colloids. Results are also moderately sensitive to the colloid retardation factor, but this parameter is better constrained by field and laboratory data.

When considering the irreversible filtration rate, it is important to make the distinction between natural colloids and anthropogenic colloids. The latter are more likely to be permanently filtered, and this filtration is easier to demonstrate in field and laboratory experiments compared with natural colloids. Natural colloids, especially clay or zeolite colloids, are considered to be more relevant for saturated zone transport. Irreversible removal of natural colloids has not been demonstrated in the field, and direct demonstration of colloid immobilization for the transport time scales of interest is unlikely. Moreover, if permanent removal of colloids is assumed, then the breakthrough depends sensitively and non-monotonically on the rate of radionuclide desorption from colloids. Irreversible attachment to colloids is not a conservative assumption in this situation, and the data requirements for desorption rates would be increased. DOE is currently not taking credit for permanent removal of natural colloids in the saturated zone, but does use the combination of permanent colloid removal and irreversible attachment to colloids in the unsaturated zone performance abstractions (BSC, 2003a). This permanent removal is meant to represent the expected behavior of colloids generated from the waste form and from corrosion products, and is outside the scope of this report. Consequences of assuming the combination of irreversible attachment and irreversible colloid removal in the unsaturated zone should be analyzed in future studies.

Indirect demonstration of colloid removal may be possible. If natural colloids are being permanently immobilized in aquifers, then this removal must be balanced by colloid generation in order to maintain the non-zero colloid concentrations ubiquitous in groundwaters. Colloid generation rates measured, for example, by flushing the system with colloid-free water and measuring the colloid concentration in the effluent, could be used to infer colloid removal rates, as has been suggested by Kersting and Reimus (2003). Such experiments would need to be conducted for relatively long times (months) to eliminate transient effects of relatively rapidly desorbing colloids.

If the rates of desorption from colloids are on the order of 0.01 yr^{-1} or faster, then the saturated zone appears to be an effective barrier to radionuclide transport after taking into account colloid facilitated transport. Rates of this magnitude are slow on the time scale of laboratory or field experiments, but are fast relative to the transport time scales of interest. Taken at face value, batch tests of plutonium desorption appear to show rates that are considerably faster than 0.01 yr^{-1} for clays and zeolites; desorption from silica is even faster (Lu et al., 1998,2000; Kersting et al., 2003a,b). Moreover, Reimus et al. (2003) have shown that desorption from colloids in flowthrough experiments on fractured cores is considerable faster than in bach experiments. They attribute this difference to collisions between colloids and the fracture surfaces, which allow the fracture surfaces to compete with colloids for radionuclides. However, these laboratory experiments should be interpreted with caution. The duration of the existing batch experiments are too short to establish whether the desorption can be described by a single-rate desorption

model. Indeed, indications are that sorption is best described by multiple sorption rates (Painter et al., 2002; Reimus, 2003); if this is true for desorption, then the rates inferred from batch experiments may simply be the fastest of several rates. Indeed, irreversible sorption on a fraction of sorption sites cannot be ruled out for the clay and zeolite colloids.

The existing laboratory and field data as well as modeling results on colloid sorption and colloid facilitated transport can be reconciled by considering that colloids in groundwater exist as heterogeneous populations of several minerals (clays, zeolites, silica, calcite) with a wide range of particle sizes and binding energies for sorption sites. It is reasonable to expect that there would be a range of filtration rates and desorption rates for these heterogeneous populations, which can be represented by a joint probability density function $f(\varepsilon, \alpha_r)$[1]. Clearly the situation of small ε and small α_r produces the largest impact of colloids. Indications are that the bulk of the $f(\varepsilon, \alpha_r)$ distribution probably lies outside the small ε and small α_r region, and the small fraction of the $f(\varepsilon, \alpha_r)$ distribution that lies within the critical region is responsible for colloid-facilitated transport observed in the field. This situation is illustrated qualitatively in Figure 6-1. Determination of the fraction of $f(\varepsilon, \alpha_r)$ that is contained in the critical region will require additional field and laboratory tests. Long-term desorption experiments in which plutonium-bearing colloids are repeatedly contacted with plutonium-free water are required to better constrain the desorption rate. From a modeling perspective, a distribution of rate constants and filtration rates would require a multigroup approach in which colloids are partitioned into several groups with different transport and plutonium sorption properties. Nonlinear sorption models with a finite sorption capacity would be required for the nearly irreversible groups to prevent them from gathering all available radionuclides as the simulation progresses. Such a multigroup approach is a conceptually straightforward generalization of the models presented here, but would require additional model and software development.

ECP 1/18/2010

[1]This should not be confused with the uncertainty distributions used for performance assessment calculations, which represent lack of knowledge about a single parameter instead of a heterogeneous population governed by multiple rates.

7. Conclusions

Based on the results presented here and on information in previously published reports and journal articles, our conclusions about colloid-facilitated transport in the saturated zone are as follows.

•Colloid-facilitated transport is likely the dominant transport mechanism for strongly sorbing radionuclides like plutonium.

•Colloid-facilitated transport depends most sensitively on two parameters: the desorption rate from colloids, and the rate of permanent immobilization of colloids.

•Permanent immobilization of natural colloids is an uncertain process that has not been demonstrated in the field. The conservative assumption of no permanent colloid filtration is probably unavoidable, at present.

•The rate of desorption from colloids is uncertain. Existing desorption experiments suggest that it is slow relative to the time scale of laboratory tests, but fast relative to transport time scales of interest. However, existing laboratory tests cannot rule out irreversible binding to clay or zeolite colloids.

•For plutonium irreversibly bound to colloids, the saturated zone attenuates the plutonium discharge by approximately one order of magnitude.

•If the rates for plutonium desorption from clay colloids measured by Lu et al. (1998, 2000) are assumed to apply to all sorption sites on all colloids, then the saturated zone attenuates the plutonium discharge by about seven orders of magnitude.

•Natural colloids exist as heterogeneous populations that are likely to have a wide range of colloid removal and radionuclide desorption rates acting simultaneously. The existence of multiple rates complicates the interpretation of laboratory and field experiments, as well as modeling. Existing data is insufficient to constrain the joint distribution of rates.

•Conservative (bounding) assumptions about colloid facilitated transport will probably be required in performance assessment calculations, at least in the near term. The models developed in this study provide an additional tool that can be used to help bound the effects of colloid facilitated transport.

•Additional field and laboratory tests may be able to constrain the distribution of multiple rates in natural colloid populations. New numerical simulation tools that are able to handle multiple classes of colloids with different rates would be needed to interpret such experiments.

References

BSC. "Technical Basis Document No. 8: Colloids." ??? Rev. 02. Las Vegas, Nevada: Bechtel SAIC Company, 2003a.

BSC. "Saturated zone colloid transport." ANL-NBS-HS-000031 Rev. 01. Las Vegas, Nevada: Bechtel SAIC Company, 2003b.

Buddemeir, R. and J. R. Hunt. "Transport of colloidal contaminants in groundwater: Radionuclide migration at the Nevada Test Site." *Applied Geochemistry*. Vol. 3. pp. 535–548. 1988.

CRWMS M&O. "Total system performance assessment for the Site Recommendation." TDR-WIS-PA-000001, Rev. 00. Las Vegas, Nevada: Civilian Radioactive Waste Management System Management and Operating Contractor. 2000.

Cvetkovic, V., S. Painter, D. Turner, D. Pickett and P. Bertetti. "Parameter and model sensitivities for colloid-facilitated radionuclide transport on the field scale." *Water Resources Research*. Vol. 40. doi:10.1029/2004WR003048. 2004.

Kersting, A., D. Efured, D. Finnegan, D. Rokop, D. Smith, and J. Thompson. "Migration of plutonium in ground water at the Nebada Test Site." *Nature*. Vol. 397. pp. 56–59. 1999.

Kersting, A. B., P. Zhao, M. Zavarin, E. R. Sylwester, P. G. Allen, L. Wang, E. J. Nelson, and R. W. Williams. "Chapter 4: Sorption of Pu(IV) on mineral colloids" in Colloid-facilitated transport of low-solubility radionuclides: A field, experimental, and modeling investigation, eds. A. B. Kersting and P. W. Reimus. Livermore, California: Lawrence Livermore National Laboratory, 2003.

Kersting, A. B., P. Zhao, M. Zavarin, E. R. Sylwester, P. G. Allen, L. Wang, and R. W. Williams. "Chapter 5: Sorption of Pu(V) on mineral colloids" in Colloid-facilitated transport of low-solubility radionuclides: A field, experimental, and modeling investigation, eds. A. B. Kersting and P. W. Reimus. Livermore, California: Lawrence Livermore National Laboratory, 2003.

Kersting, A. B. and P. W. Reimus. "Chapter 12: Recommendations" in Colloid-facilitated transport of low-solubility radionuclides: A field, experimental, and modeling investigation, eds. A. B. Kersting and P. W. Reimus. Livermore, California: Lawrence Livermore National Laboratory, 2003.

Lu, N., I. Triay, C. Cotter, H.D. Kitten, and J. Bentley. "Reversibility of sorption of Plutonium-239 onto colloids of hematitie, goethite, smectite, and silica: A milestone final report of YMP." LA-UR-98-3057. Los Alamos, New Mexico: Los Alamos National Laboratory, 1998.

Lu, N., J. Conca, G. Parker, P. Leonard, B. Moore, B. Strietelmeier, and I. Triay. "Adsorption of actinides onto colloids as function of time, temperature, ionic strength, and colloid concentration." LA-UR-00-5121. Los Alamos, New Mexico: Los Alamos National Laboratory, 2000.

Luhrmann, L. and U. Noseck, "Model of contaminant transport in porous media in the presence of colloids applied to actinide migration in column experiments." *Water Resources Research*, Vol. 34. pp. 421–426.

McCarthy, J. and J. Zachara. "Subsufrace transport of contaminants." *Environmental Science and Technology*. Vol 23. pp. 496–502. 1989.

Mohanty, S., T. J. McCartin, D. W. Esh. "Total-system performance assessment (TPA) Version 4.0 Code: Module Descriptions and User's Guide." San Antonio, Texas: Center for Nuclear Waste Regulatory Analyses. 2002.

Noell, A., J. Thompson, M. Corapcioglu and I. Triay. "The role of silica colloids in facilitating cesium transport through glass bead columns: Experiments and modeling", *Journal of Contaminant Hydrology*. Vol. 31. pp. 23–56. 1998.

NRC. "Risk Insights Baseline Report." Washington, DC: Nuclear Regulatory Commission, 2004.

Painter, S., V. Cvetkovic, D. Turner and D. Pickett. "Significance of kinetics for sorption on inorganic colloids: Modeling and experimental interpretation issues." *Environmental Science and Technology*. Vol. 24. pp. 228–234. 2002.

Penrose, W., W. Plozer, E. Essington. D. Nelson, and K. Orlandini, "Mobility of plutonium and americium through a shallow aquifer in a semiarid region." *Environmental Science and Technology*. Vol. 36. pp 5369–5375. 1990.

Reimus, P. W., S. D. Ware, N. Lu, K. S. Kung, A. Abdel-Fattah, I. Anghel, M. P. Neu, S. D. Reilly, "Chapter 7: Colloid-facilitated plutonium fracture transport experiments" in Colloid-facilitated transport of low-solubility radionuclides: A field, experimental, and modeling investigation, eds. A. B. Kersting and P. W. Reimus. Livermore, California: Lawrence Livermore National Laboratory, 2003.

Reimus, P. W. "Chapter 8: Quantitative interpretations of fracture transport experiments and associated batch sorption/desorption experiments" in Colloid-facilitated transport of low-solubility radionuclides: A field, experimental, and modeling investigation, eds. A. B. Kersting and P. W. Reimus. Livermore, California: Lawrence Livermore National Laboratory, 2003.

Saiers, J. and G. M. Hornberger. "The role of colloidal kaolinite in transport of cesium through laboratory sand columns." *Water Resources Research*. Vol. 32. pp. 33–41. 1996.

van de Weerd, H. and A. Leijnse. "Assessment of the effect of kinetics on colloid facilitated radionuclide transport in porous media." *Journal of Contaminant Hydrology*. Vol. 26. pp. 245–256. 1997.

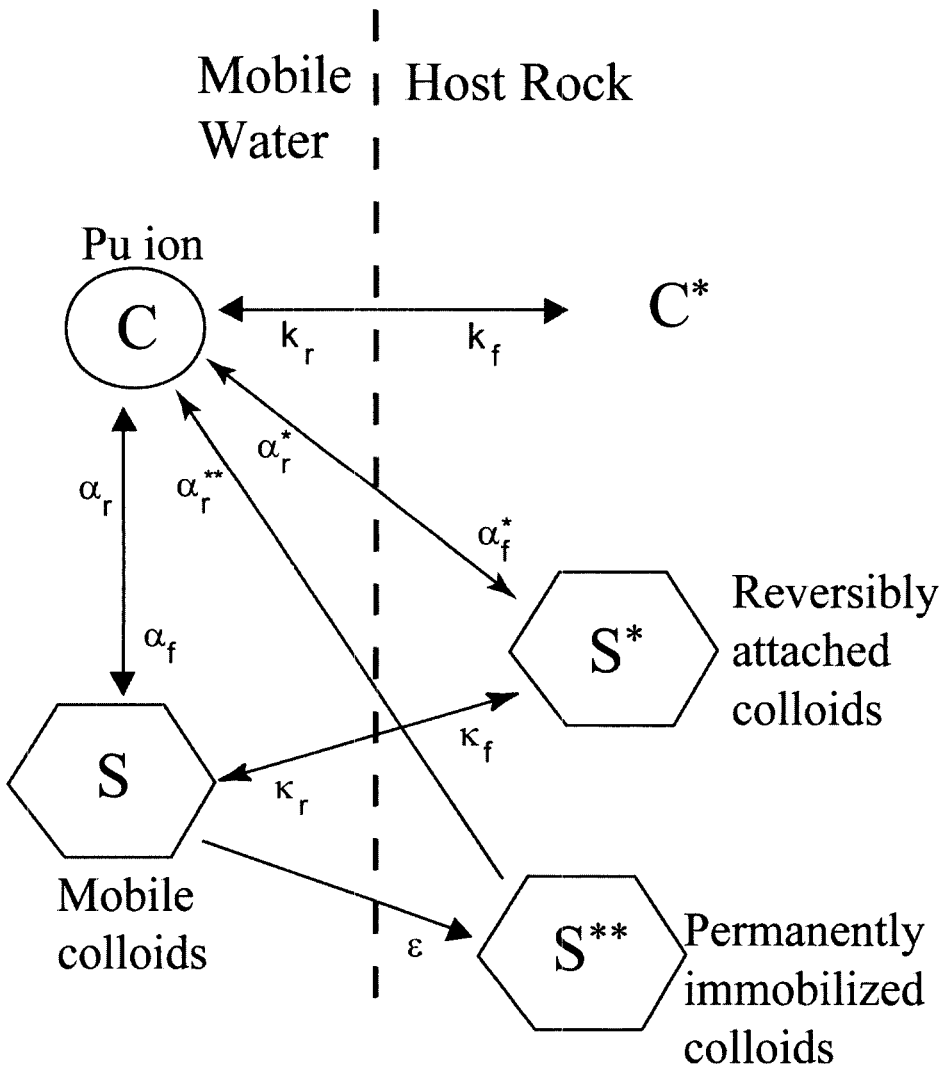


Figure 2-1. Sketch of the solution-colloid-porous matrix exchanges considered in this report.

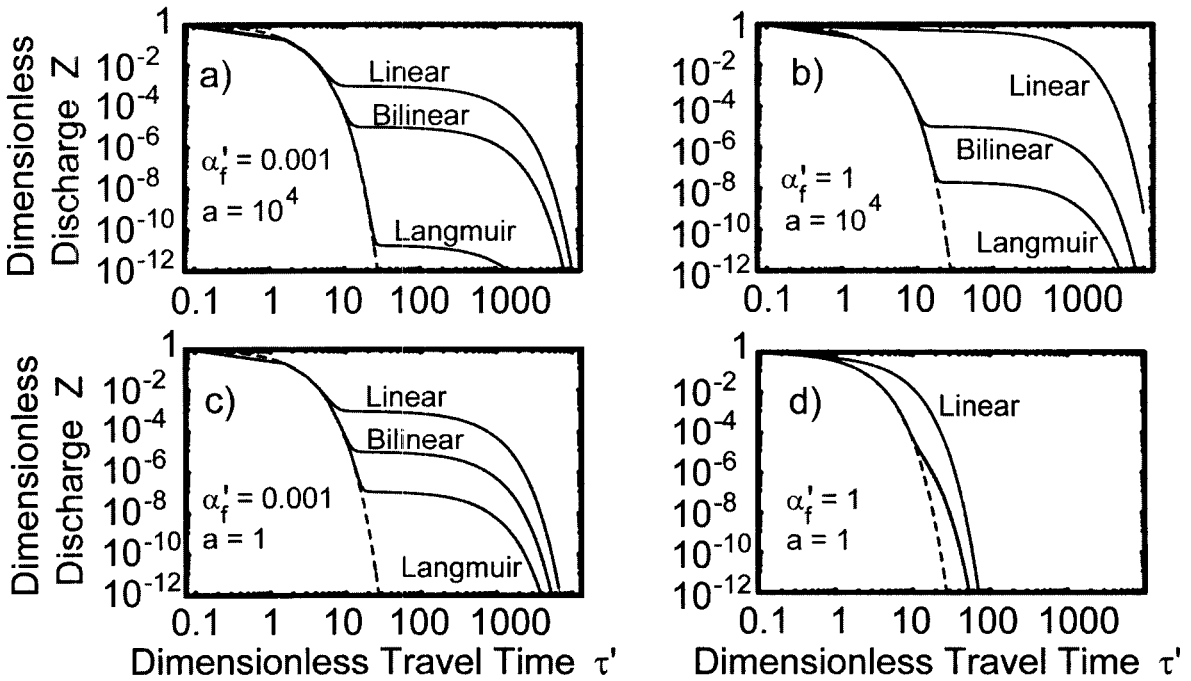


Figure 3-1. Generic example showing dependence of normalized steady-state radionuclide breakthrough on residence time τ' for linear and two nonlinear sorption models assuming a small sorption capacity (Cvetkovic et al, 2004). The dimensionless residence time is $\tau' = \tau R \lambda$ where R is the retardation factor, λ is the radionuclide decay rate, and τ is the water residence time. Four combinations of normalized forward rate and reversibility factors are shown: (a) slow forward rate and slowly reversible; (b) fast forward rate and slowly reversible; (c) slow forward rate and strongly reversible, and (d) fast forward rate and strongly reversibly. The case without colloids is shown as a dashed curve.

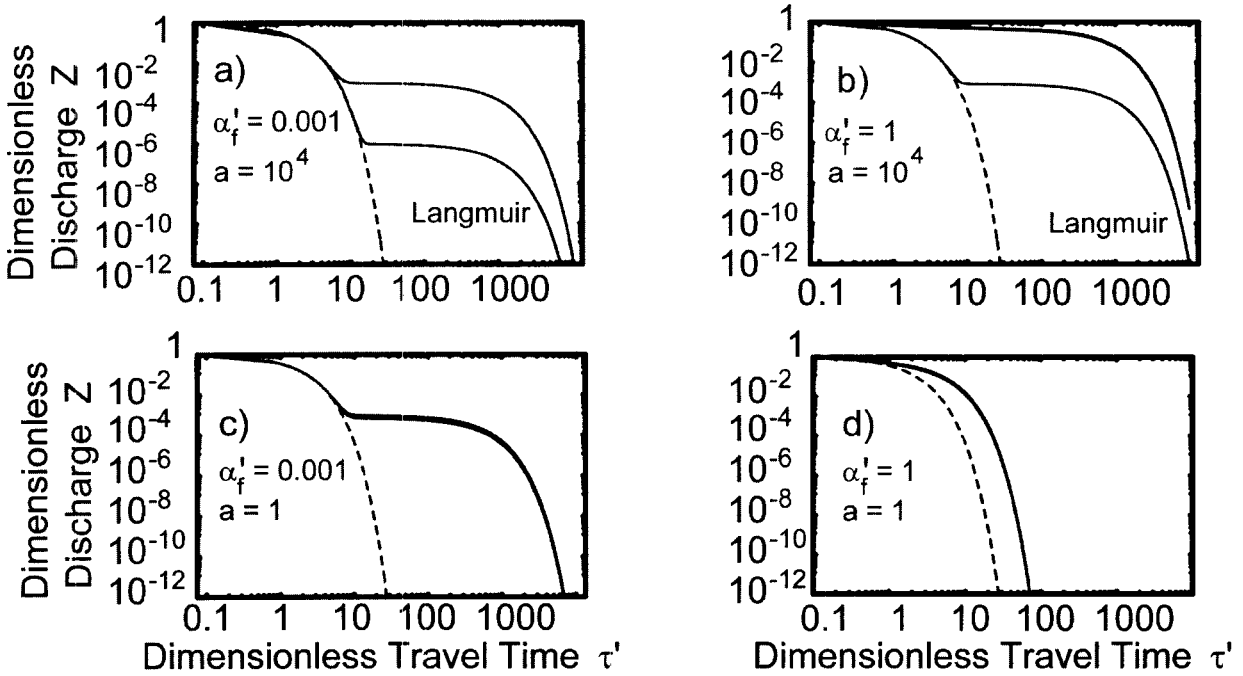


Figure 3-2. Same as Figure 2-1, but for large sorption capacity (Cvetkovic et al., 2004). In the situation of large sorption capacity, the linear and bilinear models coincide over the entire range of dimensionless travel time and for all four combinations of rate parameters. In the strongly reversible cases c) and d), the two nonlinear models coincide with the linear model.

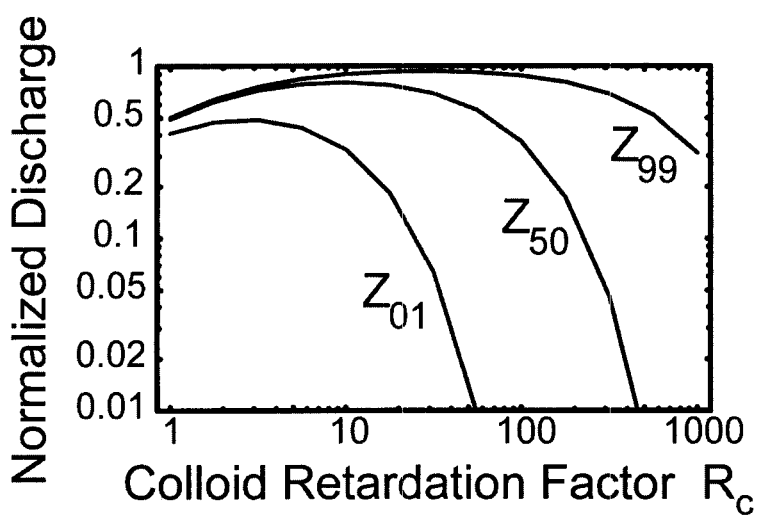


Figure 3-3. Normalized steady-state breakthrough versus colloid retardation factor for three values of travel time (Cvetkovic et al., 2004). The reversibility factor is fixed at 1000 in this example and there is no colloid filtration.

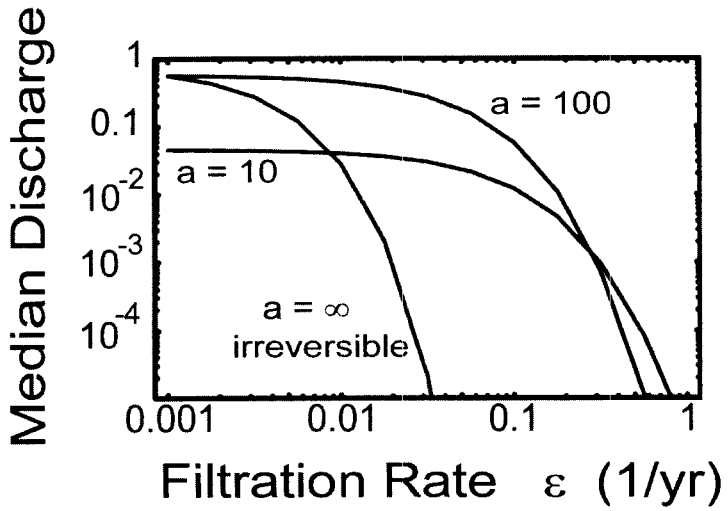


Figure 3-4. Normalized breakthrough versus colloid filtration rate for different values of the reversibility ratio *a*. The colloid retardation factor is 20 in this example, and the τ = 334 years.

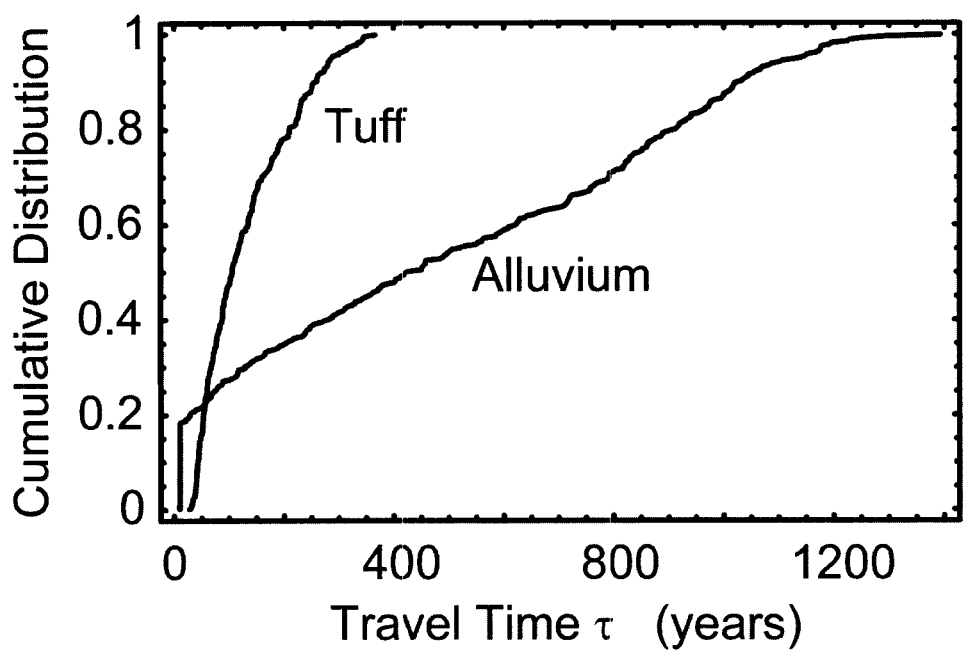


Figure 4-1. Travel time distributions computed using algorithms identical to those in TPA 5.0. Results are shown for the tuff and alluvial segments of the TPA central streamtube.

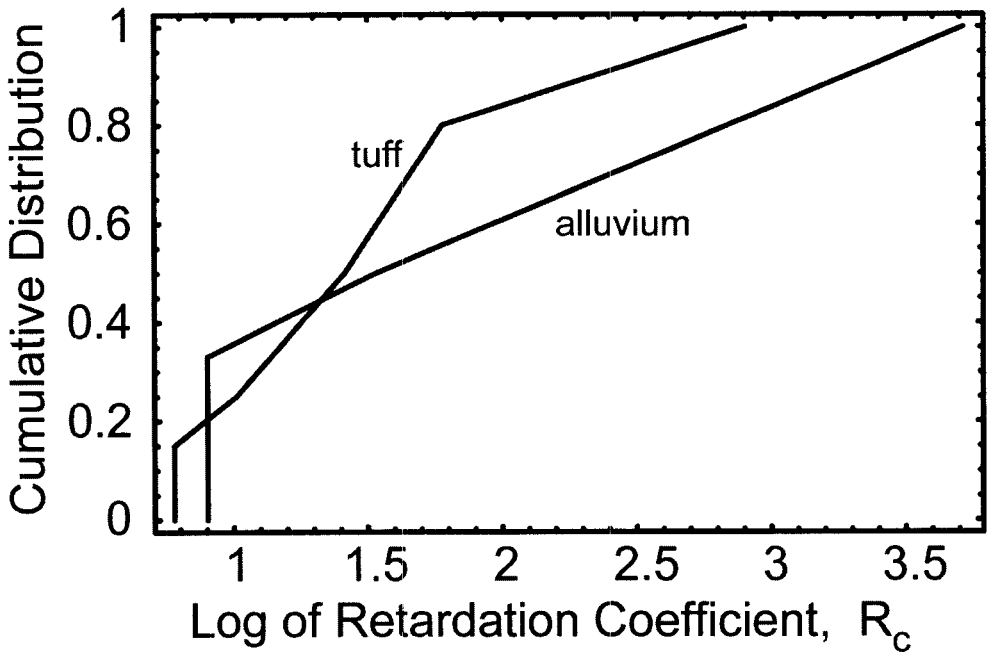


Figure 4-2. Distributions of colloid retardation factors R_c used by DOE.

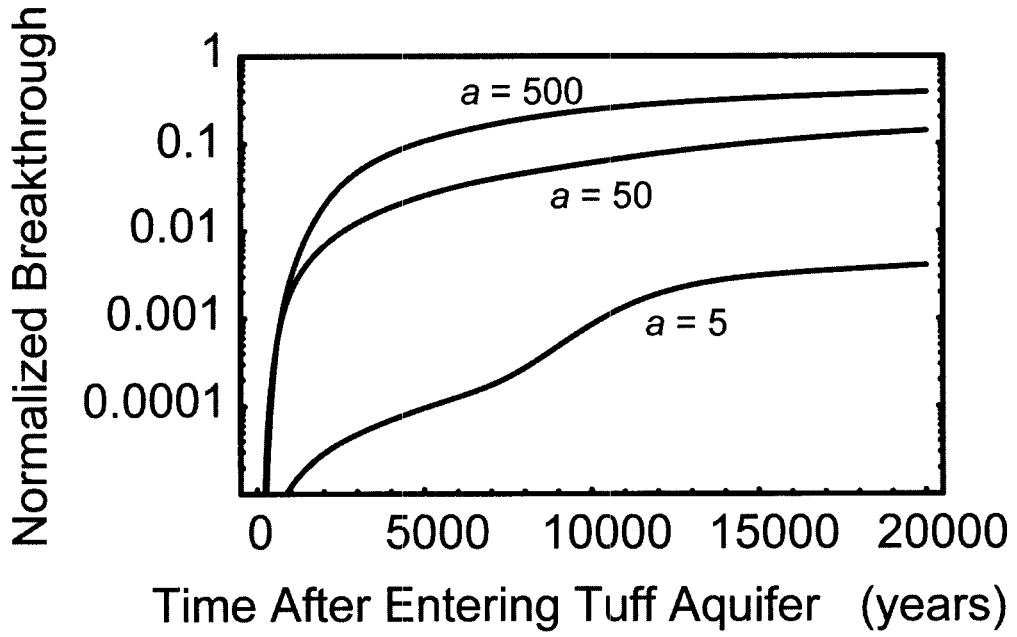


Figure 5-1. Normalized breakthrough at the tuff-alluvium interface for different values of the reversibility ratio *a*. The curves are averages over 300 realizations.

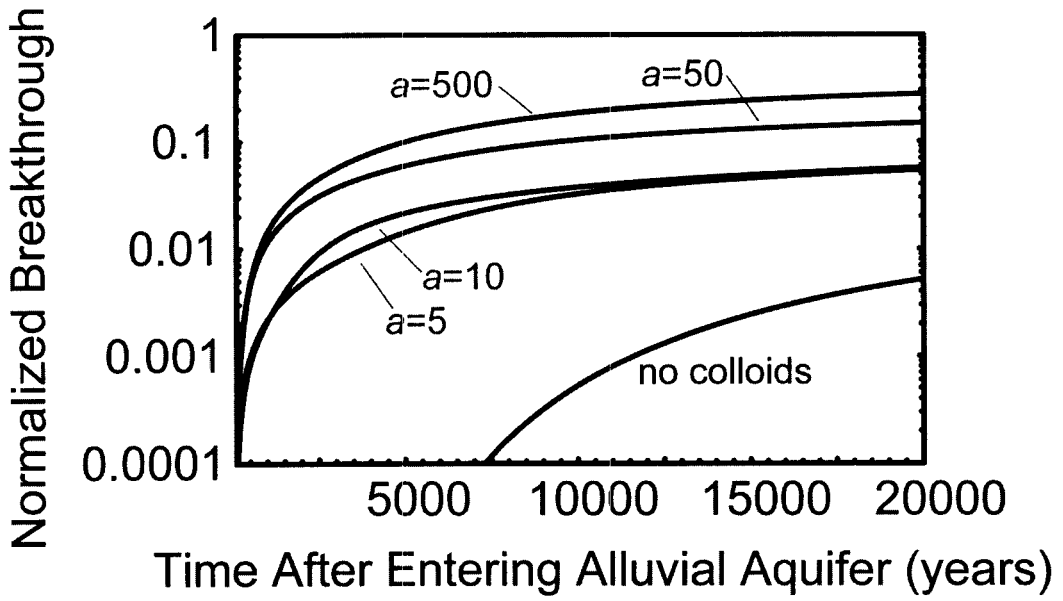


Figure 5-2. Normalized breakthrough for the alluvial aquifer considered in isolation for different values of the reversibility ratio *a*. Also shown is the case without colloids. The curves are averages over 300 realizations.

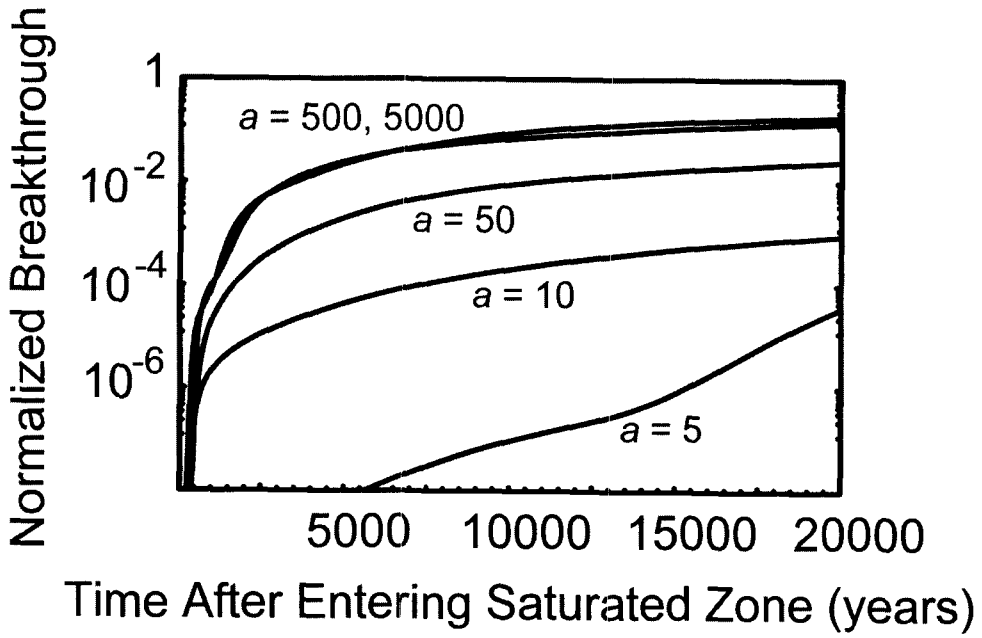


Figure 5-3. Normalized breakthrough for the entire saturated zone for different values of the reversibility ratio *a*. For each realization, the breakthrough for the fractured tuff aquifer is used as input into the alluvial aquifer. The curves are averages over 150 realizations.

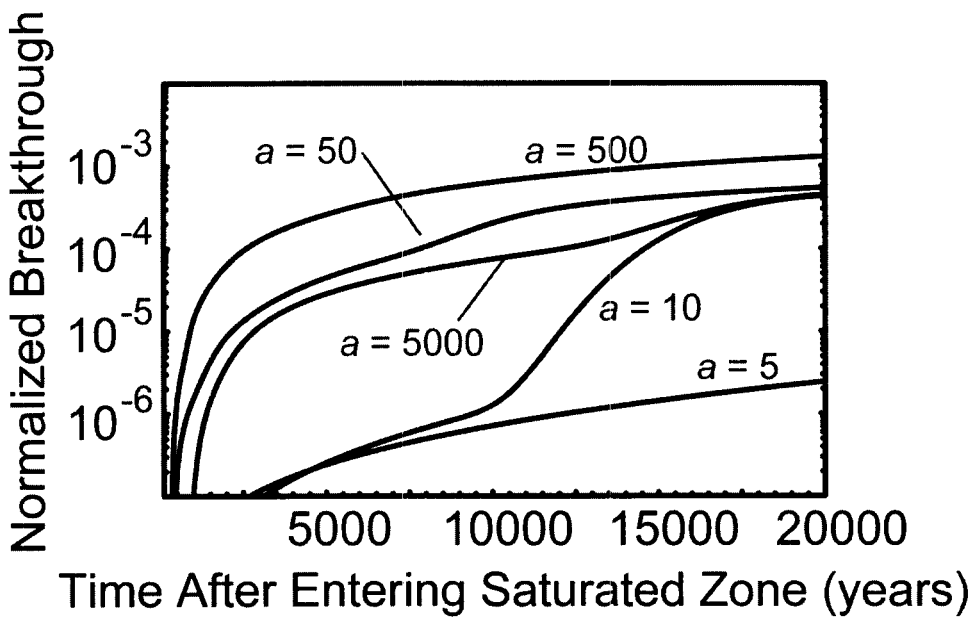


Figure 5-4. Same as in Figure 5-3, but with a colloid irreversible filtration rate of 0.05 yr^{-1}. Note that the normalized breakthrough does not have a monotonic dependence on the desorption rate (reversibility ratio).

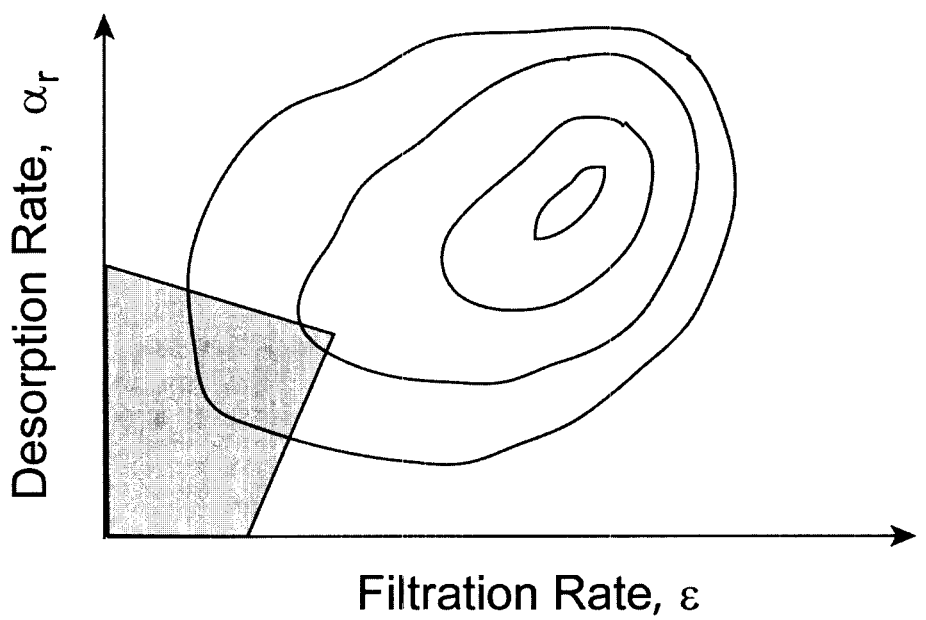


Figure 6-1. Qualitative depiction of critical region of ε, α_r parameter space. The shaded region is the most critical area for transport. The curves represent contour levels of hypothetical probability density for a heterogeneous population of colloids governed by multiple rates. The (presumably small) fraction of the population contained in the critical region is expected to be responsible for most of the transport.

6.01.04 Supporting material for previous entry

The calculations for the previous entry are documented in the following mathematica notebooks, which are attached to this scientific notebook as appendices.

Tau Dist in Alluvium and Tuff.nb	Figure 4.1 from previous entry.
CFT in Alluvium 2.nb	Figure 5.2 from previous entry.
CFT in Tuff.nb	Figure 5.1 from previous entry.
CFT in Tuff and Alluvium.nb	Figure 5.3 from previous entry.
CFT in Tuff and Alluvium Filtration.nb	Figure 5.4 from previous entry.

SCP 1/18/2010

Entries into Scientific Notebook No. 318E for the period 6.01.04 to 6.1.04were made by Scott Painter, as provided by V. Cvetkovic. No original text entered into this scientific notebook has been removed.

Scott Painter
6 1-18-10
SP

SP 1/18/2010

Colloid Facilitated Transport in Tuff and Alluvium with Filtration

Scott Painter
5.27.04

Preliminaries

```
<< Statistics`DescriptiveStatistics`
<< Graphics`Graphics`
<< Statistics`ContinuousDistributions`
$HistoryLength = 5
```

```
5
```

Set Functions for defining tau distribution

These are based on TPA 5.0 input

■ Central Streamtube

```
centralst = Partition[{0.0, 1500., 0.,
1.0, 1500., 1350.,
3.0, 450., 4350.,
9.5, 250., 10850.,
13.0, 400., 14350.,
13.5 , 375., 14850.,
15.0, 325., 16350.,
18.0 , 225., 19350.}, 3] ;
centralst = Transpose[centralst];
centralst1 = Transpose[{centralst[[1]], centralst[[3]]}]
centralst2 = Transpose[{centralst[[3]], centralst[[2]]}]
```

```
{{0., 0.}, {1., 1350.}, {3., 4350.}, {9.5, 10850.},
 {13., 14350.}, {13.5, 14850.}, {15., 16350.}, {18., 19350.}}

{{0., 1500.}, {1350., 1500.}, {4350., 450.}, {10850., 250.},
 {14350., 400.}, {14850., 375.}, {16350., 325.}, {19350., 225.}}
```

```
cstfunc1 = Interpolation[centralst1]
cstfunc2 = Interpolation[centralst2]
```

```
InterpolatingFunction[{{0., 18.}}, <>]

InterpolatingFunction[{{0., 19350.}}, <>]
```

Define the breakthrough solutions

rates are in units of 1/year
u is aqueous concentration
v is concentration sorbed to colloids
w is concentration sorbed to permanently immobilized colloids
x represents travel time
t is time
other parameters same as in report.

```
SolveTuff[alphaf_, Rc_, a_, Rs_, f_,
  thetam_, lambda_, epsilon_, tauL_] := Module[
  {alphar = alphaf / a,
Deff = 0.01 * tauL,
   dcoef = 0.001,
   thetaim = 0.2,
   n = 0.05,
   beta0, X, Y, solution},
  beta0 = dcoef / (0.28 * (f / 2 n) ^2);
  omega = f thetaim / thetam;
solution = NDSolve[
    {∂t u[t, x] == -∂x u[t, x] + Deff ∂x,x u[t, x] +
       -beta0 * (omega * u[t, x] - w[t, x])  - alphaf Rc
        u[t, x] + alphar Rc v[t, x] - lambda u[t, x],
     Rc ∂t v[t, x] == -∂x v[t, x] + Deff ∂x,x v[t, x] +
       alphaf Rc u[t, x] - alphar Rc v[t, x] -
       lambda Rc v[t, x] - epsilon v[t, x] ,
     Rs ∂t w[t, x] == beta0 * (omega * u[t, x] - w[t, x])  -
```

```
        lambda Rs w[t, x],
      ∂t z[t, x] == epsilon v[t, x] -
        alphar z[t, x] - lambda z[t, x],
      u[0, x] == 0.,
      u[t, 0] == Min[t / 5., 1],
      u[t, 3 * tauL ] == 0,
      v[0, x] == 0.,
      v[t, 3 * tauL] == 0.,
       v[t, 0] == 0.,
       w[0, x] == 0.,
       z[0, x] == 0.},
     {u, v, w, z}, {t, 0, 20000}, {x, 0, 3 * tauL},
     StartingStepSize → {1, 3 * tauL / 2000},
     AccuracyGoal → 7,
     MaxStepSize → {100, 3 * tauL / 300},
     MaxSteps → {10000, 10000}];
{X, Y} = {u, v} /. First[solution] ;
  Clear[solution];
  {X, Y}
 ]
```

```
General::spell1 :
 Possible spelling error: new symbol name "alphar" is similar to existing symbol "alphaf". More...

General::spell1 :
 Possible spelling error: new symbol name "thetaim" is similar to existing symbol "thetam". More...
```

Same notation as in SolveTuff.

Note that this function presumes that the result of SolveTuff[] is saved as the X2 and Y2 interpolation function

```
SolveAlluvium[alphaf_, R_, Rc_,
  eps_, a_, lambda_, tauL_] := Module[
  {alphar = alphaf / a,
Deff = 0.1 * tauL,
   a11, a12, a13, a21, a22, a23, a31, a32, a33,
   solution, X, Y},
  a11 = -(alphaf Rc + lambda R);
```

```
a12 = Rc alphar;
a13 = alphar;
a21 = Rc alphaf;
a22 = -(Rc alphar + eps + lambda Rc);
a23 = 0.;
a31 = 0.;
a32 = eps;
a33 = -(lambda + alphar);
solution = NDSolve[
    {R ∂t u[t, x] == -∂x u[t, x] + Deff ∂x,x u[t, x] +
       a11 u[t, x] + a12 v[t, x] + a13 w[t, x],
     Rc ∂t v[t, x] == -∂x v[t, x] + Deff ∂x,x v[t, x] +
       a22 v[t, x] + a21 u[t, x],
     ∂t w[t, x] == a32 v[t, x] + a33 w[t, x],
     u[0, x] == 0.,
     u[t, 0] == X1[t, tauL1],
     u[t, 5 * tauL] == 0,
     v[0, x] == 0.,
     v[t, 5 * tauL] == 0.,
     v[t, 0] == Y1[t, tauL1],
     w[0, x] == 0.},
    {u, v, w}, {t, 0, 20000}, {x, 0, 5 tauL},
    StartingStepSize → {1, 5 tauL / 2000},
    AccuracyGoal → 7,
    MaxStepSize → {100, 5 tauL / 300},
    MaxSteps → {10000, 10000}];
{X, Y} = {u, v} /. First[solution];
  Clear[solution];
  {X, Y}
 ]
```

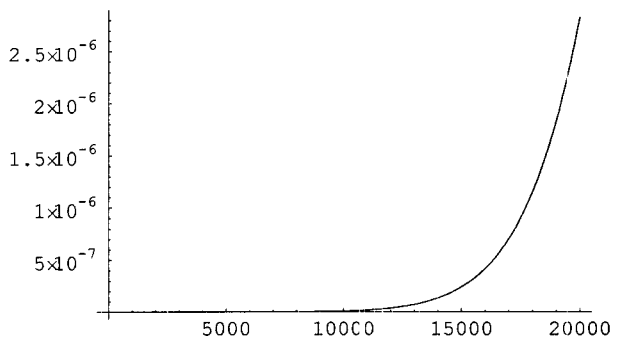
■ Reference Case Parameters

```
alphaf = 0.1;
Rc = 20;
a = 5;
lambda = 2.88 10^-5;
f = 0.032;
Rs = 1800;
thetam = 0.0032;
tauL1 = 100;
epsilon = 0.05;
```

```
{X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, epsilon, tauL1]
```

```
{InterpolatingFunction[{{0., 20000.}, {0., 300.}}, <>],
 InterpolatingFunction[{{0., 20000.}, {0., 300.}}, <>]}
```

```
result = Table[ X1[t, 100] + Y1[t, 100] , {t, 0, 20000, 100}];
time = (Range[201] - 1) * 100;
ListPlot[ Transpose[{time, result}], PlotJoined → True, PlotRange → All]
```



- Graphics -

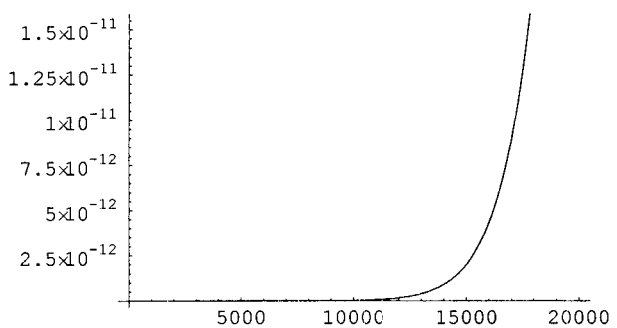
■ Alluvium

```
alphaf = 0.1;
ldist = LogNormalDistribution[9.45708, 1.10];
Rc = 20;
a = 5;
lambda = 2.88 10^-5;
tauL = 100;
R = 3400;
```

```
{X2, Y2} = SolveAlluvium[alphaf, R, Rc, epsilon, a, lambda, tauL]
```

```
{InterpolatingFunction[{{0., 20000.}, {0., 500.}}, <>],
 InterpolatingFunction[{{0., 20000.}, {0., 500.}}, <>]}
```

```
result = Table[ X2[t, 100] + Y2[t, 100] , {t, 0, 20000, 100}];
time = (Range[201] - 1) * 100;
ListPlot[ Transpose[{time, result}], PlotJoined → True ]
```




```
- Graphics -
```

---

a=5

150 realizations. Sample travel time, porosity, retardation R, and colloid retardation Rc in tuff and alluvium.
Travel time in tuff and in alluvium are correlated.
Rs is assumed perfectly correlated to R.

```
Q = 252;
nreal = 150;
a = 5;
ldist = LogNormalDistribution[9.45708, 1.10];
Rdist1 = Interpolation[ Transpose[ {{0, 0.15, 0.25, 0.5, 0.8, 1.0},
     {Log[6], Log[6], Log[10.23], Log[26.], Log[59.98], Log[800]}} ]];
Rdist2 = Interpolation[ Transpose[ {{0, 0.331, 0.5, 1.0},
     {Log[8], Log[8], Log[33.96], Log[5188.]}} ]];

bigresult = Table[
   Clear[X1, Y1];
   Clear[X2, Y2];
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   thetam = 10^Random[Real, {-3, -2}];
   f = 10^Random[Real, {-2, -1}];
   velocity = darcyv / thetam;
   tauL1 = (tddist) / velocity;
   tauL1 = Max[tauL1, 10];

   avgwidth = NIntegrate[
      cstfunc2[td], {td, tddist, 18000},
      MinRecursion → 5, MaxRecursion → 12] / (18000 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tauL2 = (18000 - tddist) / velocity;
   tauL2 = Max[tauL2, 10.];

   Rc = Exp[ Rdist1[Random[]]];
   Rs = Random[ ldist];
   {X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, epsilon, tauL1];
   R = Rs;
   Rc = Exp[ Rdist2[Random[]]];
    {X2, Y2} = SolveAlluvium[alphaf, R, Rc, epsilon, a, lambda, tauL2];

Table[ X2[t, tauL2] + Y2[t, tauL2] , {t, 0, 20000, 100}],
   {i, 1, nreal}];
```

General::stop : Further output of NDSolve::eerr will be suppressed during this calculation. More...

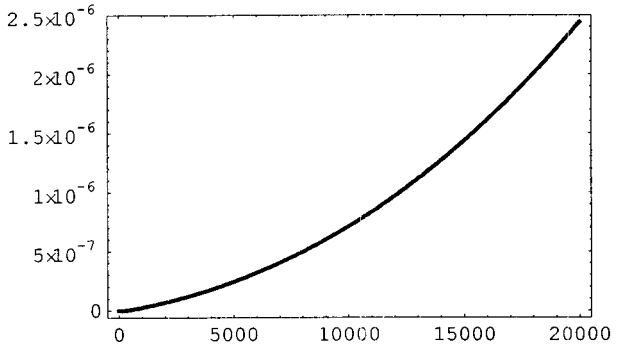
```
time = (Range[201] - 1) * 100;
```

```
meancurve5 = Apply[Plus, bigresult] / Length[bigresult]
```

{0., 1.63731×10^{-12}, 7.20366×10^{-11}, 1.41105×10^{-9}, 4.47622×10^{-9}, 7.80969×10^{-9},
1.12353×10^{-8}, 1.47242×10^{-8}, 1.82835×10^{-8}, 2.19227×10^{-8}, 2.56415×10^{-8}, 2.94397×10^{-8},
3.33175×10^{-8}, 3.72723×10^{-8}, 4.13068×10^{-8}, 4.54213×10^{-8}, 4.96218×10^{-8}, 5.38987×10^{-8},
5.82618×10^{-8}, 6.27067×10^{-8}, 6.72268×10^{-8}, 7.18302×10^{-8}, 7.65142×10^{-8}, 8.12823×10^{-8},
8.61341×10^{-8}, 9.10682×10^{-8}, 9.60858×10^{-8}, 1.01188×10^{-7}, 1.06372×10^{-7}, 1.1164×10^{-7},
1.16991×10^{-7}, 1.22426×10^{-7}, 1.27946×10^{-7}, 1.3355×10^{-7}, 1.39234×10^{-7}, 1.45003×10^{-7},
1.50856×10^{-7}, 1.56794×10^{-7}, 1.62818×10^{-7}, 1.68927×10^{-7}, 1.75122×10^{-7},
1.81403×10^{-7}, 1.87771×10^{-7}, 1.94225×10^{-7}, 2.00766×10^{-7}, 2.07394×10^{-7},
2.14109×10^{-7}, 2.20912×10^{-7}, 2.27803×10^{-7}, 2.34781×10^{-7}, 2.41848×10^{-7},
2.49003×10^{-7}, 2.56246×10^{-7}, 2.63578×10^{-7}, 2.71×10^{-7}, 2.7851×10^{-7}, 2.8611×10^{-7},
2.938×10^{-7}, 3.0158×10^{-7}, 3.0945×10^{-7}, 3.1741×10^{-7}, 3.2546×10^{-7}, 3.33602×10^{-7},
3.41834×10^{-7}, 3.50158×10^{-7}, 3.58573×10^{-7}, 3.6708×10^{-7}, 3.75679×10^{-7}, 3.8437×10^{-7},
3.93153×10^{-7}, 4.02029×10^{-7}, 4.10997×10^{-7}, 4.20059×10^{-7}, 4.29214×10^{-7}, 4.38462×10^{-7},
4.47804×10^{-7}, 4.5724×10^{-7}, 4.6677×10^{-7}, 4.76395×10^{-7}, 4.86114×10^{-7}, 4.95927×10^{-7},
5.05836×10^{-7}, 5.1584×10^{-7}, 5.2594×10^{-7}, 5.36135×10^{-7}, 5.46426×10^{-7}, 5.56813×10^{-7},
5.67297×10^{-7}, 5.77877×10^{-7}, 5.88554×10^{-7}, 5.99328×10^{-7}, 6.102×10^{-7}, 6.21169×10^{-7},
6.32235×10^{-7}, 6.434×10^{-7}, 6.54663×10^{-7}, 6.66024×10^{-7}, 6.77484×10^{-7}, 6.89042×10^{-7},
7.007×10^{-7}, 7.12458×10^{-7}, 7.24314×10^{-7}, 7.36271×10^{-7}, 7.48328×10^{-7}, 7.60485×10^{-7},
7.72743×10^{-7}, 7.85101×10^{-7}, 7.97561×10^{-7}, 8.10121×10^{-7}, 8.22783×10^{-7}, 8.35547×10^{-7},
8.48413×10^{-7}, 8.61382×10^{-7}, 8.74452×10^{-7}, 8.87626×10^{-7}, 9.00902×10^{-7}, 9.14282×10^{-7},
9.27765×10^{-7}, 9.41352×10^{-7}, 9.55043×10^{-7}, 9.68838×10^{-7}, 9.82738×10^{-7}, 9.96743×10^{-7},
1.01085×10^{-6}, 1.02507×10^{-6}, 1.03939×10^{-6}, 1.05381×10^{-6}, 1.06835×10^{-6}, 1.08299×10^{-6},
1.09773×10^{-6}, 1.11259×10^{-6}, 1.12755×10^{-6}, 1.14262×10^{-6}, 1.15779×10^{-6}, 1.17308×10^{-6},
1.18847×10^{-6}, 1.20397×10^{-6}, 1.21958×10^{-6}, 1.2353×10^{-6}, 1.25113×10^{-6}, 1.26708×10^{-6},
1.28313×10^{-6}, 1.29929×10^{-6}, 1.31556×10^{-6}, 1.33195×10^{-6}, 1.34844×10^{-6}, 1.36505×10^{-6},
1.38177×10^{-6}, 1.39861×10^{-6}, 1.41555×10^{-6}, 1.43262×10^{-6}, 1.44979×10^{-6}, 1.46708×10^{-6},
1.48448×10^{-6}, 1.502×10^{-6}, 1.51963×10^{-6}, 1.53738×10^{-6}, 1.55525×10^{-6}, 1.57323×10^{-6},
1.59133×10^{-6}, 1.60955×10^{-6}, 1.62788×10^{-6}, 1.64633×10^{-6}, 1.6649×10^{-6}, 1.68359×10^{-6},
1.7024×10^{-6}, 1.72133×10^{-6}, 1.74038×10^{-6}, 1.75955×10^{-6}, 1.77884×10^{-6}, 1.79825×10^{-6},
1.81778×10^{-6}, 1.83744×10^{-6}, 1.85721×10^{-6}, 1.87711×10^{-6}, 1.89714×10^{-6}, 1.91729×10^{-6},
1.93756×10^{-6}, 1.95796×10^{-6}, 1.97849×10^{-6}, 1.99914×10^{-6}, 2.01991×10^{-6}, 2.04082×10^{-6},
2.06185×10^{-6}, 2.08301×10^{-6}, 2.1043×10^{-6}, 2.12572×10^{-6}, 2.14727×10^{-6}, 2.16895×10^{-6},
2.19076×10^{-6}, 2.2127×10^{-6}, 2.23478×10^{-6}, 2.25699×10^{-6}, 2.27933×10^{-6}, 2.30181×10^{-6},
2.32442×10^{-6}, 2.34717×10^{-6}, 2.37005×10^{-6}, 2.39308×10^{-6}, 2.41624×10^{-6}, 2.43954×10^{-6}}

ECP 1/18/2010

```
ListPlot[ Transpose[{time, meancurve5}],
 PlotJoined → True, PlotStyle → AbsoluteThickness[2] , Frame → True ]
```



- Graphics -

RCP 1/18/2010

a=10

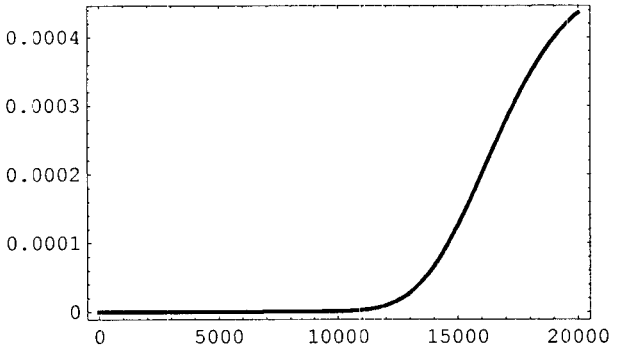
```
Q = 252;
nreal = 150;
a = 10;
ldist = LogNormalDistribution[9.45708, 1.10];
Rdist1 = Interpolation[ Transpose[ {{0, 0.15, 0.25, 0.5, 0.8, 1.0},
     {Log[6], Log[6], Log[10.23], Log[26.], Log[59.98], Log[800]}} ]];
Rdist2 = Interpolation[ Transpose[ {{0, 0.331, 0.5, 1.0},
     {Log[8], Log[8], Log[33.96], Log[5188.]}} ]];

bigresult = Table[
   Clear[X1, Y1];
   Clear[X2, Y2];
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   thetam = 10^Random[Real, {-3, -2}];
   f = 10^Random[Real, {-2, -1}];
   velocity = darcyv / thetam;
   tauL1 = (tddist) / velocity;
   tauL1 = Max[tauL1, 10];

   avgwidth = NIntegrate[
      cstfunc2[td], {td, tddist, 18000},
      MinRecursion → 5, MaxRecursion → 12] / (18000 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tauL2 = (18000 - tddist) / velocity;
   tauL2 = Max[tauL2, 10.];

   Rc = Exp[ Rdist1[Random[]]];
   Rs = Random[ ldist];
   {X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, epsilon, tauL1];
   R = Rs;
   Rc = Exp[ Rdist2[Random[]]];
    {X2, Y2} = SolveAlluvium[alphaf, R, Rc, epsilon, a, lambda, tauL2];

Table[ X2[t, tauL2] + Y2[t, tauL2] , {t, 0, 20000, 100}],
   {i, 1, nreal}];

time = (Range[201] - 1) * 100;
```

```
meancurve10 = Apply[Plus, bigresult] / Length[bigresult]
```

{0., 1.6349×10^{-18}, 6.70454×10^{-17}, 7.52789×10^{-13}, 3.59745×10^{-12}, 3.93939×10^{-11},
3.06033×10^{-10}, 1.29511×10^{-9}, 3.15878×10^{-9}, 5.46495×10^{-9}, 7.96754×10^{-9},
1.06234×10^{-8}, 1.34395×10^{-8}, 1.64136×10^{-8}, 1.95453×10^{-8}, 2.28371×10^{-8}, 2.62903×10^{-8},
2.99076×10^{-8}, 3.36913×10^{-8}, 3.76432×10^{-8}, 4.1766×10^{-8}, 4.60614×10^{-8}, 5.0532×10^{-8},
5.51797×10^{-8}, 6.00067×10^{-8}, 6.5015×10^{-8}, 7.02065×10^{-8}, 7.55839×10^{-8}, 8.11488×10^{-8},
8.6904×10^{-8}, 9.28506×10^{-8}, 9.89907×10^{-8}, 1.05327×10^{-7}, 1.1186×10^{-7}, 1.18593×10^{-7},
1.25527×10^{-7}, 1.32665×10^{-7}, 1.40008×10^{-7}, 1.47558×10^{-7}, 1.55317×10^{-7}, 1.63287×10^{-7},
1.71469×10^{-7}, 1.79866×10^{-7}, 1.88478×10^{-7}, 1.97309×10^{-7}, 2.06359×10^{-7}, 2.15631×10^{-7},
2.25126×10^{-7}, 2.34845×10^{-7}, 2.44792×10^{-7}, 2.54966×10^{-7}, 2.6537×10^{-7}, 2.76005×10^{-7},
2.86874×10^{-7}, 2.97976×10^{-7}, 3.09315×10^{-7}, 3.20892×10^{-7}, 3.32707×10^{-7}, 3.44763×10^{-7},
3.57062×10^{-7}, 3.69604×10^{-7}, 3.8239×10^{-7}, 3.95424×10^{-7}, 4.08705×10^{-7}, 4.22235×10^{-7},
4.36017×10^{-7}, 4.5005×10^{-7}, 4.64336×10^{-7}, 4.78878×10^{-7}, 4.93676×10^{-7}, 5.08732×10^{-7},
5.24047×10^{-7}, 5.39624×10^{-7}, 5.55464×10^{-7}, 5.71571×10^{-7}, 5.87948×10^{-7}, 6.04599×10^{-7},
6.2153×10^{-7}, 6.3875×10^{-7}, 6.56254×10^{-7}, 6.7406×10^{-7}, 6.92188×10^{-7}, 7.10654×10^{-7},
7.29489×10^{-7}, 7.48733×10^{-7}, 7.68441×10^{-7}, 7.88684×10^{-7}, 8.09554×10^{-7}, 8.31171×10^{-7},
8.53689×10^{-7}, 8.77299×10^{-7}, 9.02246×10^{-7}, 9.28845×10^{-7}, 9.57486×10^{-7}, 9.88648×10^{-7},
1.02291×10^{-6}, 1.06099×10^{-6}, 1.10372×10^{-6}, 1.15211×10^{-6}, 1.20738×10^{-6}, 1.27092×10^{-6},
1.34438×10^{-6}, 1.42965×10^{-6}, 1.52893×10^{-6}, 1.64473×10^{-6}, 1.77986×10^{-6}, 1.93751×10^{-6},
2.12126×10^{-6}, 2.33508×10^{-6}, 2.58334×10^{-6}, 2.87085×10^{-6}, 3.20285×10^{-6},
3.58502×10^{-6}, 4.02348×10^{-6}, 4.52476×10^{-6}, 5.09582×10^{-6}, 5.74402×10^{-6},
6.47706×10^{-6}, 7.30299×10^{-6}, 8.23016×10^{-6}, 9.26714×10^{-6}, 0.0000104227, 0.0000117057,
0.0000131253, 0.0000146902, 0.0000164095, 0.0000182919, 0.000020346, 0.0000225799,
0.0000250015, 0.0000276182, 0.0000304368, 0.0000334636, 0.0000367041, 0.0000401631,
0.0000438446, 0.0000477518, 0.0000518867, 0.0000562509, 0.0000608444, 0.0000656666,
0.0000707158, 0.0000759892, 0.0000814831, 0.0000871926, 0.0000931121, 0.0000992347,
0.000105553, 0.000112058, 0.000118741, 0.000125591, 0.000132598, 0.000139751,
0.000147036, 0.000154442, 0.000161956, 0.000169564, 0.000177252, 0.000185008,
0.000192817, 0.000200665, 0.000208538, 0.000216423, 0.000224305, 0.000232173,
0.000240012, 0.000247809, 0.000255554, 0.000263233, 0.000270837, 0.000278353,
0.000285772, 0.000293084, 0.00030028, 0.000307353, 0.000314294, 0.000321097,
0.000327755, 0.000334263, 0.000340616, 0.000346809, 0.000352839, 0.000358703,
0.000364398, 0.000369924, 0.000375277, 0.000380458, 0.000385467, 0.000390304,
0.000394969, 0.000399464, 0.000403791, 0.000407951, 0.000411947, 0.000415781,
0.000419456, 0.000422976, 0.000426343, 0.000429562, 0.000432636, 0.000435569}

ECP 1/18/2010

```
ListPlot[ Transpose[{time, meancurve10}],
 PlotJoined → True, PlotStyle → AbsoluteThickness[2] , Frame → True ]
```



- Graphics -

RB 1/18/2010

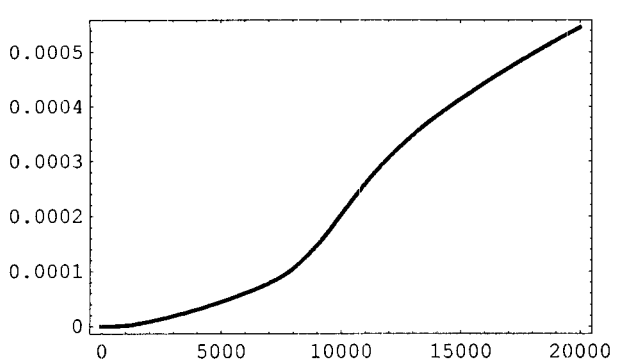
a=50

```
Q = 252;
nreal = 150;
a = 50;
ldist = LogNormalDistribution[9.45708, 1.10];
Rdist1 = Interpolation[ Transpose[ {{0, 0.15, 0.25, 0.5, 0.8, 1.0},
     {Log[6], Log[6], Log[10.23], Log[26.], Log[59.98], Log[800]}} ]];
Rdist2 = Interpolation[ Transpose[ {{0, 0.331, 0.5, 1.0},
     {Log[8], Log[8], Log[33.96], Log[5188.]}} ]];

bigresult = Table[
   Clear[X1, Y1];
   Clear[X2, Y2];
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   thetam = 10^Random[Real, {-3, -2}];
   f = 10^Random[Real, {-2, -1}];
   velocity = darcyv / thetam;
   tauL1 = (tddist) / velocity;
   tauL1 = Max[tauL1, 10];

   avgwidth = NIntegrate[
      cstfunc2[td], {td, tddist, 18000},
      MinRecursion → 5, MaxRecursion → 12] / (18000 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tauL2 = (18000 - tddist) / velocity;
   tauL2 = Max[tauL2, 10.];

   Rc = Exp[ Rdist1[Random[]]];
   Rs = Random[ ldist];
   {X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, epsilon, tauL1];
   R = Rs;
   Rc = Exp[ Rdist2[Random[]]];
    {X2, Y2} = SolveAlluvium[alphaf, R, Rc, epsilon, a, lambda, tauL2];

Table[ X2[t, tauL2] + Y2[t, tauL2] , {t, 0, 20000, 100}],
   {i, 1, nreal}];

time = (Range[201] - 1) * 100;
```

```
meancurve50 = Apply[Plus, bigresult] / Length[bigresult]
```

{0., 1.01418×10^{-17}, 4.13124×10^{-12}, 1.63298×10^{-9}, 5.41723×10^{-8}, 2.2328×10^{-7}, 4.46518×10^{-7}, 6.97618×10^{-7}, 9.67798×10^{-7}, 1.25662×10^{-6}, 1.57682×10^{-6}, 1.94636×10^{-6}, 2.39028×10^{-6}, 2.93414×10^{-6}, 3.58278×10^{-6}, 4.31659×10^{-6}, 5.10842×10^{-6}, 5.93708×10^{-6}, 6.79103×10^{-6}, 7.66536×10^{-6}, 8.55816×10^{-6}, 9.46887×10^{-6}, 0.0000103978, 0.0000113455, 0.0000123122, 0.0000132985, 0.0000143043, 0.00001533, 0.0000163757, 0.0000174411, 0.0000185262, 0.0000196311, 0.0000207555, 0.0000218996, 0.0000230631, 0.0000242459, 0.0000254482, 0.0000266696, 0.0000279101, 0.0000291699, 0.0000304482, 0.0000317454, 0.0000330613, 0.0000343958, 0.0000357487, 0.0000371198, 0.0000385091, 0.0000399164, 0.0000413416, 0.0000427845, 0.0000442451, 0.0000457232, 0.0000472187, 0.0000487317, 0.0000502622, 0.0000518104, 0.0000533765, 0.0000549611, 0.0000565648, 0.0000581886, 0.0000598341, 0.0000615029, 0.0000631973, 0.0000649204, 0.0000666756, 0.0000684673, 0.0000703005, 0.0000721809, 0.0000741153, 0.0000761109, 0.000078176, 0.0000803192, 0.0000825498, 0.0000848775, 0.0000873123, 0.0000898639, 0.0000925423, 0.0000953568, 0.0000983163, 0.000101429, 0.000104701, 0.000108139, 0.000111747, 0.000115528, 0.000119483, 0.000123612, 0.000127913, 0.000132382, 0.000137013, 0.0001418, 0.000146734, 0.000151805, 0.000157003, 0.000162315, 0.000167728, 0.000173229, 0.000178805, 0.000184441, 0.000190123, 0.000195838, 0.000201572, 0.000207311, 0.000213045, 0.00021876, 0.000224448, 0.000230094, 0.000235693, 0.000241237, 0.000246718, 0.00025213, 0.000257467, 0.000262725, 0.0002679, 0.000272993, 0.000277999, 0.000282917, 0.000287748, 0.000292491, 0.000297148, 0.000301719, 0.000306207, 0.000310612, 0.000314937, 0.000319184, 0.000323357, 0.000327457, 0.000331487, 0.00033545, 0.000339349, 0.000343187, 0.000346966, 0.000350689, 0.000354358, 0.000357977, 0.000361547, 0.00036507, 0.00036855, 0.000371987, 0.000375384, 0.000378743, 0.000382066, 0.000385353, 0.000388608, 0.00039183, 0.000395022, 0.000398185, 0.00040132, 0.000404427, 0.000407509, 0.000410566, 0.000413599, 0.000416609, 0.000419596, 0.000422562, 0.000425506, 0.00042843, 0.000431335, 0.00043422, 0.000437087, 0.000439936, 0.000442767, 0.000445581, 0.000448379, 0.00045116, 0.000453925, 0.000456675, 0.00045941, 0.00046213, 0.000464836, 0.000467528, 0.000470206, 0.00047287, 0.000475522, 0.00047816, 0.000480786, 0.0004834, 0.000486002, 0.000488592, 0.00049117, 0.000493737, 0.000496293, 0.000498838, 0.000501372, 0.000503896, 0.000506409, 0.000508913, 0.000511406, 0.00051389, 0.000516364, 0.000518829, 0.000521285, 0.000523732, 0.000526169, 0.000528598, 0.000531019, 0.000533431, 0.000535834, 0.000538229, 0.000540617, 0.000542996, 0.000545367}

```
ListPlot[ Transpose[{time, meancurve50}],
 PlotJoined → True, PlotStyle → AbsoluteThickness[2] , Frame → True ]
```



- Graphics -

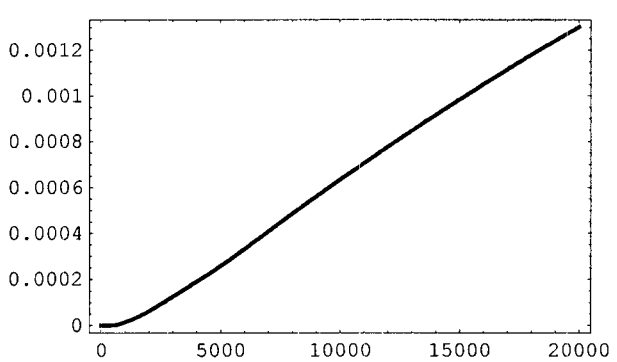
a=500

```
Q = 252;
nreal = 150;
a = 500;
ldist = LogNormalDistribution[9.45708, 1.10];
Rdist1 = Interpolation[ Transpose[ {{0, 0.15, 0.25, 0.5, 0.8, 1.0},
     {Log[6], Log[6], Log[10.23], Log[26.], Log[59.98], Log[800]}} ]];
Rdist2 = Interpolation[ Transpose[ {{0, 0.331, 0.5, 1.0},
     {Log[8], Log[8], Log[33.96], Log[5188.]}} ]];

bigresult = Table[
   Clear[X1, Y1];
   Clear[X2, Y2];
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   thetam = 10^Random[Real, {-3, -2}];
   f = 10^Random[Real, {-2, -1}];
   velocity = darcyv / thetam;
   tauL1 = (tddist) / velocity;
   tauL1 = Max[tauL1, 10];

   avgwidth = NIntegrate[
      cstfunc2[td], {td, tddist, 18000},
      MinRecursion → 5, MaxRecursion → 12] / (18000 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tauL2 = (18000 - tddist) / velocity;
   tauL2 = Max[tauL2, 10.];

   Rc = Exp[ Rdist1[Random[]]];
   Rs = Random[ ldist];
   {X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, epsilon, tauL1];
   R = Rs;
   Rc = Exp[ Rdist2[Random[]]];
    {X2, Y2} = SolveAlluvium[alphaf, R, Rc, epsilon, a, lambda, tauL2];

Table[ X2[t, tauL2] + Y2[t, tauL2] , {t, 0, 20000, 100}],
   {i, 1, nreal}];

time = (Range[201] - 1) * 100;
```

```
meancurve500 = Apply[Plus, bigresult] / Length[bigresult]
```

{0., 7.71613×10^{-12}, 7.23824×10^{-11}, 3.85968×10^{-8}, 5.64126×10^{-7}, 1.55382×10^{-6},
2.72048×10^{-6}, 4.2733×10^{-6}, 6.77077×10^{-6}, 0.0000102193, 0.0000141333, 0.0000181835,
0.0000222848, 0.000026473, 0.0000308376, 0.0000354741, 0.0000404358, 0.0000457158,
0.0000512664, 0.0000570293, 0.0000629536, 0.0000690011, 0.0000751439, 0.0000813604,
0.0000876329, 0.0000939475, 0.000100293, 0.000106662, 0.000113046, 0.00011944,
0.000125842, 0.000132249, 0.00013866, 0.000145075, 0.000151494, 0.000157917,
0.000164344, 0.000170778, 0.00017722, 0.000183673, 0.00019014, 0.000196627,
0.000203139, 0.000209684, 0.000216268, 0.000222901, 0.000229591, 0.000236345,
0.000243172, 0.000250078, 0.000257066, 0.00026414, 0.000271299, 0.000278542,
0.000285866, 0.000293265, 0.000300734, 0.000308264, 0.000315849, 0.000323481,
0.000331151, 0.000338853, 0.000346579, 0.000354324, 0.000362081, 0.000369845,
0.000377612, 0.000385378, 0.00039314, 0.000400896, 0.000408643, 0.000416379,
0.000424104, 0.000431815, 0.000439512, 0.000447194, 0.000454861, 0.000462512,
0.000470147, 0.000477765, 0.000485366, 0.000492949, 0.000500516, 0.000508065,
0.000515597, 0.000523112, 0.000530609, 0.000538088, 0.000545551, 0.000552996,
0.000560423, 0.000567834, 0.000575228, 0.000582606, 0.000589967, 0.000597312,
0.000604642, 0.000611955, 0.000619254, 0.000626538, 0.000633808, 0.000641063,
0.000648305, 0.000655533, 0.000662748, 0.000669951, 0.000677142, 0.00068432,
0.000691487, 0.000698642, 0.000705786, 0.00071292, 0.000720043, 0.000727155,
0.000734257, 0.000741349, 0.000748431, 0.000755503, 0.000762566, 0.000769618,
0.000776661, 0.000783695, 0.000790718, 0.000797732, 0.000804736, 0.00081173,
0.000818714, 0.000825688, 0.000832652, 0.000839606, 0.00084655, 0.000853483,
0.000860405, 0.000867317, 0.000874218, 0.000881108, 0.000887987, 0.000894854,
0.000901711, 0.000908555, 0.000915388, 0.00092221, 0.000929019, 0.000935816,
0.000942602, 0.000949375, 0.000956135, 0.000962883, 0.000969619, 0.000976342,
0.000983052, 0.000989749, 0.000996433, 0.0010031, 0.00100976, 0.00101641,
0.00102304, 0.00102966, 0.00103626, 0.00104285, 0.00104943, 0.00105599,
0.00106254, 0.00106908, 0.0010756, 0.00108211, 0.0010886, 0.00109508, 0.00110155,
0.001108, 0.00111444, 0.00112087, 0.00112728, 0.00113367, 0.00114005, 0.00114642,
0.00115277, 0.00115911, 0.00116543, 0.00117174, 0.00117804, 0.00118432,
0.00119059, 0.00119684, 0.00120307, 0.0012093, 0.0012155, 0.0012217, 0.00122788,
0.00123404, 0.00124019, 0.00124632, 0.00125244, 0.00125855, 0.00126464,
0.00127071, 0.00127677, 0.00128282, 0.00128885, 0.00129486, 0.00130086}

```
ListPlot[ Transpose[{time, meancurve500}],
 PlotJoined → True, PlotStyle → AbsoluteThickness[2] , Frame → True ]
```



- Graphics -

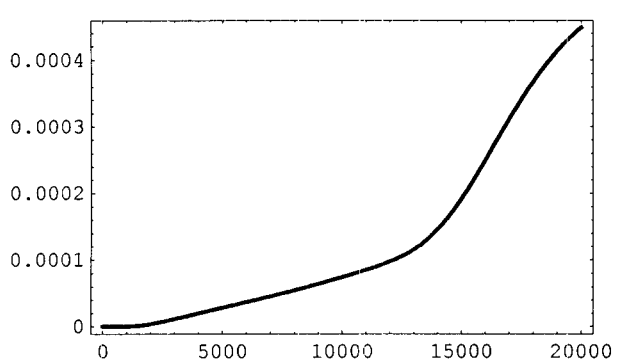
a=5000

```
Q = 252;
nreal = 150;
a = 5000;
ldist = LogNormalDistribution[9.45708, 1.10];
Rdist1 = Interpolation[ Transpose[ {{0, 0.15, 0.25, 0.5, 0.8, 1.0},
     {Log[6], Log[6], Log[10.23], Log[26.], Log[59.98], Log[800]}} ]];
Rdist2 = Interpolation[ Transpose[ {{0, 0.331, 0.5, 1.0},
     {Log[8], Log[8], Log[33.96], Log[5188.]}} ]];

bigresult = Table[
   Clear[X1, Y1];
   Clear[X2, Y2];
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   thetam = 10^Random[Real, {-3, -2}];
   f = 10^Random[Real, {-2, -1}];
   velocity = darcyv / thetam;
   tauL1 = (tddist) / velocity;
   tauL1 = Max[tauL1, 10];

   avgwidth = NIntegrate[
      cstfunc2[td], {td, tddist, 18000},
      MinRecursion → 5, MaxRecursion → 12] / (18000 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tauL2 = (18000 - tddist) / velocity;
   tauL2 = Max[tauL2, 10.];

   Rc = Exp[ Rdist1[Random[]]];
   Rs = Random[ ldist];
   {X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, epsilon, tauL1];
   R = Rs;
   Rc = Exp[ Rdist2[Random[]]];
    {X2, Y2} = SolveAlluvium[alphaf, R, Rc, epsilon, a, lambda, tauL2];

Table[ X2[t, tauL2] + Y2[t, tauL2] , {t, 0, 20000, 100}],
   {i, 1, nreal}];

time = (Range[201] - 1) * 100;
```

```
meancurve5000 = Apply[Plus, bigresult] / Length[bigresult]
```

{0., 1.55439×10^{-24}, 1.14527×10^{-15}, 5.11669×10^{-15}, 1.7717×10^{-14}, 1.64619×10^{-12},
5.13714×10^{-11}, 5.57299×10^{-10}, 5.71807×10^{-9}, 3.56426×10^{-8}, 1.24439×10^{-7},
2.85348×10^{-7}, 5.05407×10^{-7}, 7.69582×10^{-7}, 1.07243×10^{-6}, 1.41565×10^{-6}, 1.80362×10^{-6},
2.24088×10^{-6}, 2.73089×10^{-6}, 3.27399×10^{-6}, 3.86712×10^{-6}, 4.50471×10^{-6},
5.18006×10^{-6}, 5.88696×10^{-6}, 6.62009×10^{-6}, 7.37516×10^{-6}, 8.14899×10^{-6},
8.93846×10^{-6}, 9.74151×10^{-6}, 0.0000105555, 0.000011379, 0.00001221, 0.0000130472,
0.0000138892, 0.000014735, 0.0000155837, 0.0000164345, 0.0000172866, 0.0000181396,
0.0000189932, 0.0000198471, 0.0000207011, 0.000021555, 0.0000224088, 0.0000232625,
0.000024116, 0.0000249694, 0.0000258227, 0.0000266759, 0.0000275289, 0.0000283819,
0.0000292347, 0.0000300876, 0.0000309403, 0.0000317932, 0.0000326464, 0.0000334999,
0.0000343541, 0.000035209, 0.0000360649, 0.0000369221, 0.0000377808, 0.0000386413,
0.000039504, 0.000040369, 0.0000412367, 0.0000421073, 0.0000429811, 0.0000438583,
0.0000447392, 0.0000456239, 0.0000465127, 0.0000474056, 0.0000483029, 0.0000492045,
0.0000501106, 0.0000510213, 0.0000519366, 0.0000528564, 0.0000537809, 0.0000547099,
0.0000556435, 0.0000565817, 0.0000575242, 0.0000584712, 0.0000594225, 0.0000603781,
0.0000613379, 0.0000623018, 0.0000632697, 0.0000642416, 0.0000652175, 0.0000661971,
0.0000671807, 0.000068168, 0.0000691591, 0.0000701541, 0.0000711531, 0.0000721561,
0.0000731633, 0.000074175, 0.0000751916, 0.0000762134, 0.000077241, 0.0000782752,
0.0000793167, 0.0000803665, 0.0000814259, 0.0000824963, 0.0000835792, 0.0000846767,
0.0000857908, 0.0000869241, 0.0000880793, 0.0000892596, 0.0000904684, 0.0000917094,
0.0000929869, 0.0000943053, 0.0000956695, 0.0000970845, 0.000098556, 0.00010009,
0.000101691, 0.000103367, 0.000105124, 0.000106968, 0.000108906, 0.000110945,
0.000113091, 0.000115351, 0.000117731, 0.000120237, 0.000122875, 0.000125651,
0.00012857, 0.000131635, 0.000134852, 0.000138223, 0.000141751, 0.000145438,
0.000149286, 0.000153296, 0.000157465, 0.000161795, 0.000166282, 0.000170924,
0.000175718, 0.000180659, 0.000185742, 0.000190961, 0.000196311, 0.000201783,
0.00020737, 0.000213063, 0.000218855, 0.000224735, 0.000230694, 0.000236722,
0.00024281, 0.000248947, 0.000255122, 0.000261325, 0.000267547, 0.000273776,
0.000280002, 0.000286217, 0.000292409, 0.00029857, 0.000304692, 0.000310765,
0.000316781, 0.000322733, 0.000328613, 0.000334416, 0.000340134, 0.000345762,
0.000351295, 0.000356729, 0.000362059, 0.000367283, 0.000372396, 0.000377397,
0.000382283, 0.000387054, 0.000391708, 0.000396244, 0.000400663, 0.000404964,
0.000409148, 0.000413217, 0.00041717, 0.000421011, 0.00042474, 0.000428359,
0.000431872, 0.000435279, 0.000438585, 0.000441791, 0.000444901, 0.000447917}

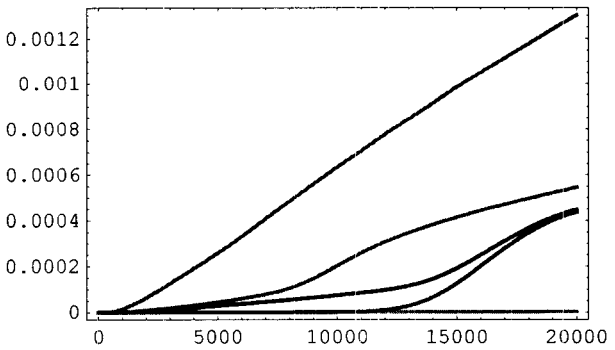
```
ListPlot[Transpose[{time, meancurve5000}],
 PlotJoined → True, PlotStyle → AbsoluteThickness[2], Frame → True]
```



- Graphics -

Display Together

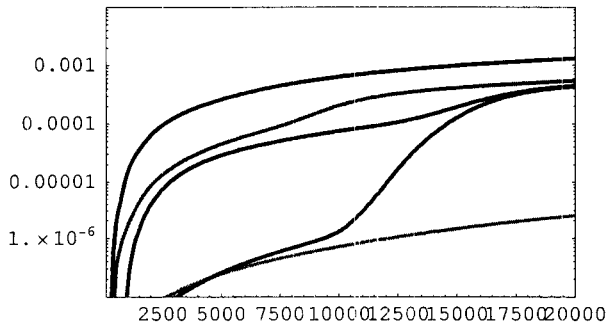
```
DisplayTogether[
 ListPlot[ Transpose[{time, meancurve5000}],
  PlotJoined → True, PlotStyle → AbsoluteThickness[2] ],
 ListPlot[ Transpose[{time, meancurve50}], PlotJoined → True,
  PlotStyle → {AbsoluteThickness[2], RGBColor[1, 0, 0]} ] ,
 ListPlot[ Transpose[{time, meancurve500}], PlotJoined → True,
  PlotStyle → {AbsoluteThickness[2], RGBColor[1, 0, 0]} ] ,
 ListPlot[ Transpose[{time, meancurve5}], PlotJoined → True,
  PlotStyle → {AbsoluteThickness[2], RGBColor[0, 1, 0]} ] ,
 ListPlot[ Transpose[{time, meancurve10}], PlotJoined → True,
  PlotStyle → {AbsoluteThickness[2], RGBColor[0, 0, 1]} ] ,
 Frame → True, PlotRange → All
]
```



- Graphics -

ECP 1/18/2010

```
DisplayTogether[
 LogListPlot[ Transpose[{time, meancurve5000}],
  PlotJoined → True, PlotStyle → AbsoluteThickness[2] ],
 LogListPlot[ Transpose[{time, meancurve50}], PlotJoined → True,
  PlotStyle → {AbsoluteThickness[2], RGBColor[1, 0, 0]} ] ,
 LogListPlot[ Transpose[{time, meancurve5}], PlotJoined → True,
  PlotStyle → {AbsoluteThickness[2], RGBColor[0, 1, 0]} ] ,
 LogListPlot[ Transpose[{time, meancurve10}], PlotJoined → True,
  PlotStyle → {AbsoluteThickness[2], RGBColor[0, 0, 1]} ] ,
 LogListPlot[ Transpose[{time, meancurve500}],
  PlotJoined → True, PlotStyle → AbsoluteThickness[2] ],
 Frame → True, PlotRange → { {100, 20000}, {-7, -2}}, Axes → False
]
```



- Graphics -

ED 1/18/2010

Colloid Facilitated Transport in Tuff and Alluvium

Scott Painter
5.27.04

Preliminaries

```
<< Statistics`DescriptiveStatistics`
<< Graphics`Graphics`
<< Statistics`ContinuousDistributions`
$HistoryLength = 5
```

```
5
```

Set Functions for defining tau distribution

These are based on TPA 5.0 input

Central Streamtube

```
centralst = Partition[{0.0, 1500., 0.,
1.0, 1500., 1350.,
3.0, 450., 4350.,
9.5, 250., 10850.,
13.0, 400., 14350.,
13.5 , 375., 14850.,
15.0, 325., 16350.,
18.0 , 225., 19350.}, 3] ;
centralst = Transpose[centralst];
centralst1 = Transpose[{centralst[[1]], centralst[[3]]}]
centralst2 = Transpose[{centralst[[3]], centralst[[2]]}]
```

```
{{0., 0.}, {1., 1350.}, {3., 4350.}, {9.5, 10850.},
 {13., 14350.}, {13.5, 14850.}, {15., 16350.}, {18., 19350.}}

{{0., 1500.}, {1350., 1500.}, {4350., 450.}, {10850., 250.},
 {14350., 400.}, {14850., 375.}, {16350., 325.}, {19350., 225.}}
```

```
cstfunc1 = Interpolation[centralst1]
cstfunc2 = Interpolation[centralst2]
```

```
InterpolatingFunction[{{0., 18.}}, <>]
```

```
InterpolatingFunction[{{0., 19350.}}, <>]
```

Define the breakthrough solutions

rates are in units of 1/year
u is aqueous concentration
v is concentration sorbed to colloids
w is concentration sorbed to permanently immobilized colloids
x represents travel time
t is time
other parameters same as in report.

```
SolveTuff[alphaf_, Rc_, a_, Rs_,
  f_, thetam_, lambda_, tauL_] := Module[
  {alphar = alphaf / a,
Deff = 0.01 * tauL,
   dcoef = 0.001,
   thetaim = 0.2,
   n = 0.05,
   beta0, X, Y, solution},
  beta0 = dcoef / (0.28 * (f / 2 n) ^2);
  omega = f thetaim / thetam;
solution = NDSolve[
    {∂t u[t, x] == -∂x u[t, x] + Deff ∂x,x u[t, x] +
       -beta0 * (omega * u[t, x] - w[t, x])  - alphaf Rc
        u[t, x] + alphar Rc v[t, x] - lambda u[t, x],
     Rc ∂t v[t, x] == -∂x v[t, x] + Deff ∂x,x v[t, x] +
       alphaf Rc u[t, x] -
       alphar Rc v[t, x] - lambda Rc v[t, x] ,
     Rs ∂t w[t, x] == beta0 * (omega * u[t, x] - w[t, x])  -
```

```
        lambda Rs w[t, x],
      u[0, x] == 0.,
      u[t, 0] == Min[t / 5., 1],
      u[t, 3 * tauL ] == 0,
      v[0, x] == 0.,
      v[t, 3 * tauL] == 0.,
       v[t, 0] == 0.,
       w[0, x] == 0.},
     {u, v, w}, {t, 0, 20000}, {x, 0, 3 * tauL},
     StartingStepSize → {1, 3 * tauL / 2000},
     AccuracyGoal → 7,
     MaxStepSize → {100, 3 * tauL / 300},
     MaxSteps → {10000, 10000}];
{X, Y} = {u, v} /. First[solution] ;
  Clear[solution];
  {X, Y}
 ]
```

```
General::spell1 :
 Possible spelling error: new symbol name "thetaim" is similar to existing symbol "thetam". More...
```

Same notation as in SolveTuff.

Note that this function presumes that the result of SolveTuff[] is saved as the X2 and Y2 interpolation function

```
SolveAlluvium[alphaf_, R_, Rc_,
  eps_, a_, lambda_, tauL_] := Module[
  {alphar = alphaf / a,
Deff = 0.1 * tauL,
   a11, a12, a13, a21, a22, a23, a31, a32, a33,
   solution, X, Y},
  a11 = - (alphaf Rc + lambda R ) ;
a12 = Rc alphar;
a13 = alphar;
a21 = Rc alphaf;
a22 = - (Rc alphar + eps + lambda Rc) ;
```

```
a23 = 0.;
a31 = 0.;
a32 = eps;
a33 = -(lambda + alphar);
solution = NDSolve[
    {R ∂t u[t, x] == -∂x u[t, x] + Deff ∂x,x u[t, x] +
       a11 u[t, x] + a12 v[t, x] + a13 w[t, x],
     Rc ∂t v[t, x] == -∂x v[t, x] + Deff ∂x,x v[t, x] +
       a22 v[t, x] + a21 u[t, x],
     ∂t w[t, x] == a32 v[t, x] + a33 w[t, x],
     u[0, x] == 0.,
     u[t, 0] == X1[t, tauL1],
     u[t, 5 * tauL] == 0,
     v[0, x] == 0.,
     v[t, 5 * tauL] == 0.,
     v[t, 0] == Y1[t, tauL1],
     w[0, x] == 0.},
    {u, v, w}, {t, 0, 20000}, {x, 0, 5 tauL},
    StartingStepSize → {1, 5 tauL / 2000},
    AccuracyGoal → 7,
    MaxStepSize → {100, 5 tauL / 300},
    MaxSteps → {10000, 10000}];
{X, Y} = {u, v} /. First[solution];
  Clear[solution];
  {X, Y}
 ]
```

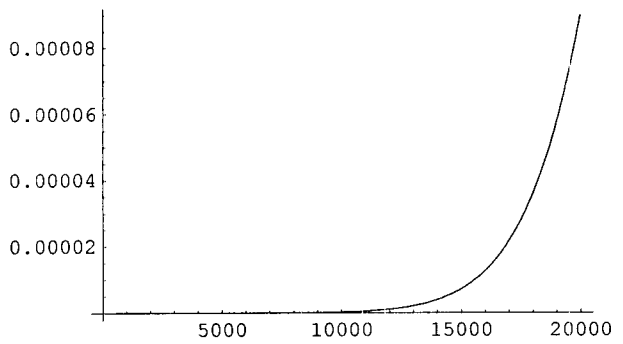
■ Reference Case Parameters

```
alphaf = 0.1;
Rc = 20;
a = 5;
lambda = 2.88 10^-5;
f = 0.032;
Rs = 1800;
thetam = 0.0032;
tauL1 = 100;
```

```
{X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, tauL1]
```

```
{InterpolatingFunction[{{0., 20000.}, {0., 300.}}, <>],
 InterpolatingFunction[{{0., 20000.}, {0., 300.}}, <>]}
```

```
result = Table[X1[t, 100] + Y1[t, 100] , {t, 0, 20000, 100}];
time = (Range[201] - 1) * 100;
ListPlot[Transpose[{time, result}], PlotJoined → True, PlotRange → All]
```



- Graphics -

RP 1/18/2010

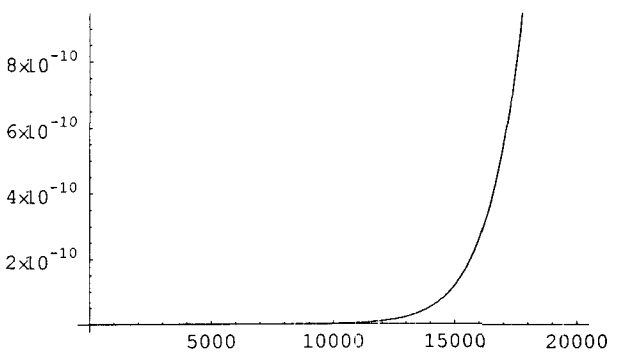
■ Alluvium

```
alphaf = 0.1;
ldist = LogNormalDistribution[9.45708, 1.10];
Rc = 20;
eps = 0.0;
a = 5;
lambda = 2.88 10^-5;
tauL = 100;
R = 3400;
```

```
{X2, Y2} = SolveAlluvium[alphaf, R, Rc, eps, a, lambda, tauL]
```

```
{InterpolatingFunction[{{0., 20000.}, {0., 500.}}, <>],
 InterpolatingFunction[{{0., 20000.}, {0., 500.}}, <>]}
```

```
result = Table[ X2[t, 100] + Y2[t, 100] , {t, 0, 20000, 100}];
time = (Range[201] - 1) * 100;
ListPlot[ Transpose[{time, result}], PlotJoined → True ]
```



- Graphics -

---

a=5

150 realizations. Sample travel time, porosity, retardation R, and colloid retardation Rc in tuff and alluvium.
Travel time in tuff and in alluvium are correlated.
Rs is assumed perfectly correlated to R.
Reversibility factor is 5.

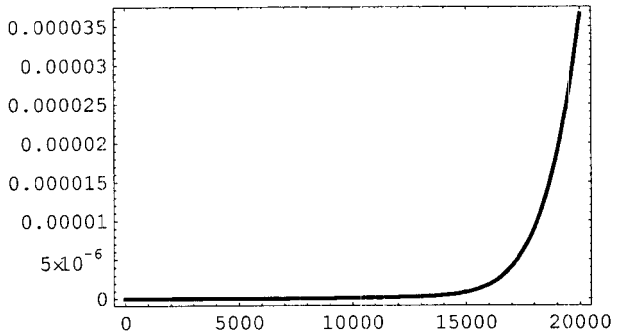
```
Q = 252;
nreal = 150;
a = 5;
ldist = LogNormalDistribution[9.45708, 1.10];
Rdist1 = Interpolation[ Transpose[ {{0, 0.15, 0.25, 0.5, 0.8, 1.0},
     {Log[6], Log[6], Log[10.23], Log[26.], Log[59.98], Log[800]}} ]];
Rdist2 = Interpolation[ Transpose[ {{0, 0.331, 0.5, 1.0},
     {Log[8], Log[8], Log[33.96], Log[5188.]}} ]];

bigresult = Table[
   Clear[X1, Y1];
   Clear[X2, Y2];
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   thetam = 10^Random[Real, {-3, -2}];
   f = 10^Random[Real, {-2, -1}];
   velocity = darcyv / thetam;
   tauL1 = (tddist) / velocity;
   tauL1 = Max[tauL1, 10];

   avgwidth = NIntegrate[
      cstfunc2[td], {td, tddist, 18000},
      MinRecursion → 5, MaxRecursion → 12] / (18000 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tauL2 = (18000 - tddist) / velocity;
   tauL2 = Max[tauL2, 10.];

   Rc = Exp[ Rdist1[Random[]]];
   Rs = Random[ ldist];
   {X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, tauL1];
   R = Rs;
   Rc = Exp[ Rdist2[Random[]]];
    {X2, Y2} = SolveAlluvium[alphaf, R, Rc, eps, a, lambda, tauL2];

Table[ X2[t, tauL2] + Y2[t, tauL2] , {t, 0, 20000, 100}],
   {i, 1, nreal}];

time = (Range[201] - 1) * 100;
```

```
meancurve5 = Apply[Plus, bigresult] / Length[bigresult]
```

{0., 2.83435×10^{-25}, 3.40956×10^{-17}, 5.80452×10^{-14}, 1.2523×10^{-12}, 6.93262×10^{-12}, 2.07206×10^{-11}, 4.33561×10^{-11}, 7.27922×10^{-11}, 1.06417×10^{-10}, 1.42752×10^{-10}, 1.81054×10^{-10}, 2.21853×10^{-10}, 2.65097×10^{-10}, 3.1116×10^{-10}, 3.59789×10^{-10}, 4.11148×10^{-10}, 4.64789×10^{-10}, 5.21467×10^{-10}, 5.8127×10^{-10}, 6.44227×10^{-10}, 7.1095×10^{-10}, 7.81854×10^{-10}, 8.57303×10^{-10}, 9.37427×10^{-10}, 1.0224×10^{-9}, 1.11283×10^{-9}, 1.20928×10^{-9}, 1.31218×10^{-9}, 1.42177×10^{-9}, 1.53809×10^{-9}, 1.66216×10^{-9}, 1.79454×10^{-9}, 1.9359×10^{-9}, 2.08701×10^{-9}, 2.24744×10^{-9}, 2.4188×10^{-9}, 2.60214×10^{-9}, 2.79741×10^{-9}, 3.00637×10^{-9}, 3.2301×10^{-9}, 3.46961×10^{-9}, 3.726×10^{-9}, 4.00038×10^{-9}, 4.29388×10^{-9}, 4.60772×10^{-9}, 4.94314×10^{-9}, 5.3014×10^{-9}, 5.68383×10^{-9}, 6.09176×10^{-9}, 6.52656×10^{-9}, 6.98964×10^{-9}, 7.48243×10^{-9}, 8.0064×10^{-9}, 8.56303×10^{-9}, 9.15382×10^{-9}, 9.78028×10^{-9}, 1.04439×10^{-8}, 1.11463×10^{-8}, 1.1889×10^{-8}, 1.26735×10^{-8}, 1.35013×10^{-8}, 1.43742×10^{-8}, 1.52935×10^{-8}, 1.62609×10^{-8}, 1.7278×10^{-8}, 1.83462×10^{-8}, 1.94671×10^{-8}, 2.06422×10^{-8}, 2.1873×10^{-8}, 2.31611×10^{-8}, 2.45078×10^{-8}, 2.59147×10^{-8}, 2.73832×10^{-8}, 2.89146×10^{-8}, 3.05105×10^{-8}, 3.2172×10^{-8}, 3.39007×10^{-8}, 3.56977×10^{-8}, 3.75645×10^{-8}, 3.95022×10^{-8}, 4.15122×10^{-8}, 4.35956×10^{-8}, 4.57536×10^{-8}, 4.79875×10^{-8}, 5.02983×10^{-8}, 5.26872×10^{-8}, 5.51554×10^{-8}, 5.77039×10^{-8}, 6.03338×10^{-8}, 6.30464×10^{-8}, 6.58427×10^{-8}, 6.8724×10^{-8}, 7.16912×10^{-8}, 7.47458×10^{-8}, 7.78889×10^{-8}, 8.11219×10^{-8}, 8.44443×10^{-8}, 8.78587×10^{-8}, 9.13667×10^{-8}, 9.49699×10^{-8}, 9.86701×10^{-8}, 1.02469×10^{-7}, 1.06369×10^{-7}, 1.10373×10^{-7}, 1.14482×10^{-7}, 1.18701×10^{-7}, 1.23033×10^{-7}, 1.27482×10^{-7}, 1.32053×10^{-7}, 1.36751×10^{-7}, 1.41583×10^{-7}, 1.46554×10^{-7}, 1.51674×10^{-7}, 1.56952×10^{-7}, 1.62397×10^{-7}, 1.68023×10^{-7}, 1.73843×10^{-7}, 1.79872×10^{-7}, 1.86129×10^{-7}, 1.92634×10^{-7}, 1.9941×10^{-7}, 2.06482×10^{-7}, 2.13881×10^{-7}, 2.21639×10^{-7}, 2.29794×10^{-7}, 2.38387×10^{-7}, 2.47467×10^{-7}, 2.57086×10^{-7}, 2.67303×10^{-7}, 2.78185×10^{-7}, 2.89805×10^{-7}, 3.02246×10^{-7}, 3.15598×10^{-7}, 3.29964×10^{-7}, 3.45455×10^{-7}, 3.62196×10^{-7}, 3.80323×10^{-7}, 3.99988×10^{-7}, 4.21357×10^{-7}, 4.44613×10^{-7}, 4.69957×10^{-7}, 4.97608×10^{-7}, 5.27806×10^{-7}, 5.60815×10^{-7}, 5.9692×10^{-7}, 6.36433×10^{-7}, 6.79694×10^{-7}, 7.2707×10^{-7}, 7.7896×10^{-7}, 8.35798×10^{-7}, 8.9805×10^{-7}, 9.6622×10^{-7}, 1.04085×10^{-6}, 1.12253×10^{-6}, 1.21189×10^{-6}, 1.3096×10^{-6}, 1.41639×10^{-6}, 1.53303×10^{-6}, 1.66036×10^{-6}, 1.79926×10^{-6}, 1.95067×10^{-6}, 2.11561×10^{-6}, 2.29514×10^{-6}, 2.49041×10^{-6}, 2.70261×10^{-6}, 2.93304×10^{-6}, 3.18304×10^{-6}, 3.45405×10^{-6}, 3.74758×10^{-6}, 4.06522×10^{-6}, 4.40866×10^{-6}, 4.77965×10^{-6}, 5.18006×10^{-6}, 5.61184×10^{-6}, 6.07702×10^{-6}, 6.57774×10^{-6}, 7.11623×10^{-6}, 7.69483×10^{-6}, 8.31597×10^{-6}, 8.98219×10^{-6}, 9.69612×10^{-6}, 0.0000104605, 0.0000112782, 0.0000121521, 0.0000130854, 0.0000140811, 0.0000151425, 0.000016273, 0.0000174762, 0.0000187554, 0.0000201145, 0.0000215572, 0.0000230874, 0.0000247091, 0.0000264262, 0.000028243, 0.0000301637, 0.0000321927, 0.0000343342, 0.0000365929}

ECP 1/18/2010

```
ListPlot[ Transpose[{time, meancurve5}], PlotJoined → True,
 PlotStyle → AbsoluteThickness[2] , Frame → True, PlotRange → All ]
```



- Graphics -

RD 1/18/2010

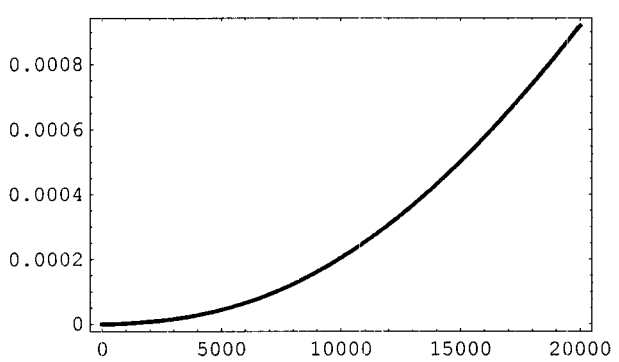
a=10

```
Q = 252;
nreal = 150;
a = 10;
ldist = LogNormalDistribution[9.45708, 1.10];
Rdist1 = Interpolation[ Transpose[ {{0, 0.15, 0.25, 0.5, 0.8, 1.0},
     {Log[6], Log[6], Log[10.23], Log[26.], Log[59.98], Log[800]}} ]];
Rdist2 = Interpolation[ Transpose[ {{0, 0.331, 0.5, 1.0},
     {Log[8], Log[8], Log[33.96], Log[5188.]}} ]];

bigresult = Table[
   Clear[X1, Y1];
   Clear[X2, Y2];
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   thetam = 10^Random[Real, {-3, -2}];
   f = 10^Random[Real, {-2, -1}];
   velocity = darcyv / thetam;
   tauL1 = (tddist) / velocity;
   tauL1 = Max[tauL1, 10];

   avgwidth = NIntegrate[
      cstfunc2[td], {td, tddist, 18000},
      MinRecursion → 5, MaxRecursion → 12] / (18000 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tauL2 = (18000 - tddist) / velocity;
   tauL2 = Max[tauL2, 10.];

   Rc = Exp[ Rdist1[Random[]]];
   Rs = Random[ ldist];
   {X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, tauL1];
   R = Rs;
   Rc = Exp[ Rdist2[Random[]]];
    {X2, Y2} = SolveAlluvium[alphaf, R, Rc, eps, a, lambda, tauL2];

Table[ X2[t, tauL2] + Y2[t, tauL2] , {t, 0, 20000, 100}],
   {i, 1, nreal}];

time = (Range[201] - 1) * 100;
```

```
meancurve10 = Apply[Plus, bigresult] / Length[bigresult]
```

{0., 2.95812×10^{-18}, 1.40118×10^{-12}, 4.63109×10^{-10}, 3.64265×10^{-8}, 2.44228×10^{-7},
5.69145×10^{-7}, 9.31982×10^{-7}, 1.31761×10^{-6}, 1.72266×10^{-6}, 2.14583×10^{-6},
2.58668×10^{-6}, 3.04486×10^{-6}, 3.52057×10^{-6}, 4.01438×10^{-6}, 4.52699×10^{-6},
5.05918×10^{-6}, 5.61189×10^{-6}, 6.18618×10^{-6}, 6.7832×10^{-6}, 7.40419×10^{-6}, 8.05044×10^{-6},
8.72329×10^{-6}, 9.42412×10^{-6}, 0.0000101543, 0.0000109153, 0.0000117083,
0.0000125349, 0.0000133962, 0.0000142935, 0.0000152281, 0.0000162011, 0.0000172137,
0.0000182668, 0.0000193616, 0.0000204989, 0.0000216795, 0.0000229044, 0.0000241743,
0.0000254898, 0.0000268517, 0.0000282606, 0.0000297171, 0.0000312216, 0.0000327746,
0.0000343766, 0.0000360281, 0.0000377293, 0.0000394807, 0.0000412826, 0.0000431352,
0.0000450389, 0.0000469939, 0.0000490005, 0.0000510588, 0.0000531691, 0.0000553317,
0.0000575466, 0.000059814, 0.0000621341, 0.000064507, 0.000066933, 0.0000694121,
0.0000719444, 0.0000745301, 0.0000771692, 0.000079862, 0.0000826084, 0.0000854086,
0.0000882627, 0.0000911703, 0.000094133, 0.0000971492, 0.00010022, 0.000103344,
0.000106524, 0.000109757, 0.000113045, 0.000116388, 0.000119785, 0.000123237,
0.000126743, 0.000130304, 0.00013392, 0.000137591, 0.000141317, 0.000145097,
0.000148933, 0.000152823, 0.000156768, 0.000160769, 0.000164824, 0.000168934,
0.000173099, 0.00017732, 0.000181595, 0.000185925, 0.00019031, 0.000194751,
0.000199246, 0.000203796, 0.000208401, 0.000213061, 0.000217775, 0.000222545,
0.000227369, 0.000232248, 0.000237182, 0.00024217, 0.000247213, 0.000252311,
0.000257463, 0.000262669, 0.00026793, 0.000273245, 0.000278614, 0.000284037,
0.000289515, 0.000295046, 0.000300631, 0.00030627, 0.000311962, 0.000317708,
0.000323508, 0.00032936, 0.000335267, 0.000341226, 0.000347238, 0.000353303,
0.000359421, 0.000365592, 0.000371815, 0.00037809, 0.000384418, 0.000390798,
0.00039723, 0.000403714, 0.000410249, 0.000416836, 0.000423475, 0.000430164,
0.000436905, 0.000443697, 0.00045054, 0.000457433, 0.000464377, 0.000471371,
0.000478415, 0.00048551, 0.000492654, 0.000499848, 0.000507091, 0.000514384,
0.000521726, 0.000529117, 0.000536557, 0.000544045, 0.000551582, 0.000559167,
0.000566801, 0.000574482, 0.000582212, 0.000589988, 0.000597813, 0.000605684,
0.000613603, 0.000621569, 0.000629581, 0.00063764, 0.000645745, 0.000653897,
0.000662094, 0.000670338, 0.000678627, 0.000686962, 0.000695342, 0.000703767,
0.000712237, 0.000720753, 0.000729312, 0.000737917, 0.000746566, 0.000755258,
0.000763995, 0.000772776, 0.000781601, 0.000790469, 0.00079938, 0.000808335,
0.000817333, 0.000826374, 0.000835458, 0.000844584, 0.000853753, 0.000862965,
0.000872218, 0.000881514, 0.000890852, 0.000900231, 0.000909653, 0.000919116}

```
ListPlot[Transpose[{time, meancurve10}], PlotJoined → True,
 PlotStyle → AbsoluteThickness[2], Frame → True, PlotRange → All]
```



- Graphics -

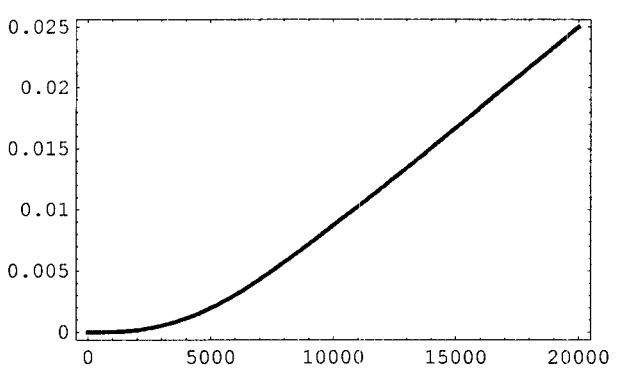
a=50

```
Q = 252;
nreal = 150;
a = 50;
ldist = LogNormalDistribution[9.45708, 1.10];
Rdist1 = Interpolation[ Transpose[ {{0, 0.15, 0.25, 0.5, 0.8, 1.0},
     {Log[6], Log[6], Log[10.23], Log[26.], Log[59.98], Log[800]}} ]];
Rdist2 = Interpolation[ Transpose[ {{0, 0.331, 0.5, 1.0},
     {Log[8], Log[8], Log[33.96], Log[5188.]}} ]];

bigresult = Table[
   Clear[X1, Y1];
   Clear[X2, Y2];
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   thetam = 10^Random[Real, {-3, -2}];
   f = 10^Random[Real, {-2, -1}];
   velocity = darcyv / thetam;
   tauL1 = (tddist) / velocity;
   tauL1 = Max[tauL1, 10];

   avgwidth = NIntegrate[
      cstfunc2[td], {td, tddist, 18000},
      MinRecursion → 5, MaxRecursion → 12] / (18000 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tauL2 = (18000 - tddist) / velocity;
   tauL2 = Max[tauL2, 10.];

   Rc = Exp[ Rdist1[Random[]]];
   Rs = Random[ ldist];
   {X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, tauL1];
   R = Rs;
   Rc = Exp[ Rdist2[Random[]]];
    {X2, Y2} = SolveAlluvium[alphaf, R, Rc, eps, a, lambda, tauL2];

Table[ X2[t, tauL2] + Y2[t, tauL2] , {t, 0, 20000, 100}],
   {i, 1, nreal}];

time = (Range[201] - 1) * 100;
```

```
meancurve50 = Apply[Plus, bigresult] / Length[bigresult]
```

{0., 1.45477×10^{-25}, 2.7525×10^{-14}, 8.90896×10^{-11}, 5.61909×10^{-9}, 2.13186×10^{-7},
1.47485×10^{-6}, 4.04141×10^{-6}, 7.85247×10^{-6}, 0.0000129166, 0.0000190056, 0.0000260157,
0.0000341804, 0.0000438578, 0.0000552834, 0.0000685591, 0.0000837513, 0.000100926,
0.000120143, 0.000141443, 0.000164861, 0.000190436, 0.000218197, 0.000248161,
0.000280326, 0.000314685, 0.000351226, 0.000389948, 0.00043086, 0.000473983,
0.000519348, 0.000566991, 0.00061695, 0.000669268, 0.000723987, 0.000781142,
0.000840771, 0.000902902, 0.000967565, 0.00103478, 0.00110456, 0.00117693, 0.00125189,
0.00132944, 0.00140958, 0.00149231, 0.0015776, 0.00166546, 0.00175584, 0.00184874,
0.00194412, 0.00204194, 0.00214218, 0.00224478, 0.00234971, 0.00245692, 0.00256635,
0.00267797, 0.00279171, 0.00290752, 0.00302533, 0.0031451, 0.00326676, 0.00339024,
0.00351549, 0.00364244, 0.00377103, 0.00390119, 0.00403286, 0.00416597, 0.00430046,
0.00443627, 0.00457333, 0.00471157, 0.00485095, 0.00499139, 0.00513284, 0.00527524,
0.00541854, 0.00556267, 0.00570758, 0.00585323, 0.00599955, 0.00614652, 0.00629407,
0.00644217, 0.00659077, 0.00673984, 0.00688935, 0.00703925, 0.00718952, 0.00734013,
0.00749105, 0.00764227, 0.00779375, 0.0079455, 0.00809749, 0.0082497, 0.00840214,
0.00855479, 0.00870765, 0.00886071, 0.00901399, 0.00916747, 0.00932116, 0.00947508,
0.00962921, 0.00978358, 0.00993818, 0.010093, 0.0102481, 0.0104035, 0.0105592,
0.0107151, 0.0108714, 0.0110279, 0.0111848, 0.011342, 0.0114996, 0.0116575,
0.0118157, 0.0119743, 0.0121332, 0.0122925, 0.0124522, 0.0126122, 0.0127725,
0.0129333, 0.0130943, 0.0132557, 0.0134175, 0.0135795, 0.0137419, 0.0139046,
0.0140675, 0.0142308, 0.0143943, 0.014558, 0.014722, 0.0148862, 0.0150506, 0.0152152,
0.0153799, 0.0155448, 0.0157099, 0.015875, 0.0160403, 0.0162056, 0.0163711,
0.0165366, 0.0167021, 0.0168676, 0.0170332, 0.0171988, 0.0173644, 0.0175299,
0.0176955, 0.017861, 0.0180265, 0.018192, 0.0183574, 0.0185227, 0.018688, 0.0188533,
0.0190185, 0.0191836, 0.0193487, 0.0195137, 0.0196787, 0.0198436, 0.0200085,
0.0201734, 0.0203382, 0.020503, 0.0206678, 0.0208326, 0.0209973, 0.0211621,
0.0213268, 0.0214916, 0.0216564, 0.0218213, 0.0219862, 0.0221511, 0.0223161,
0.0224812, 0.0226464, 0.0228116, 0.022977, 0.0231424, 0.023308, 0.0234737, 0.0236395,
0.0238055, 0.0239716, 0.0241379, 0.0243043, 0.024471, 0.0246378, 0.0248048, 0.024972}

```
ListPlot[Transpose[{time, meancurve50}],
 PlotJoined → True, PlotStyle → AbsoluteThickness[2], Frame → True]
```



- Graphics -

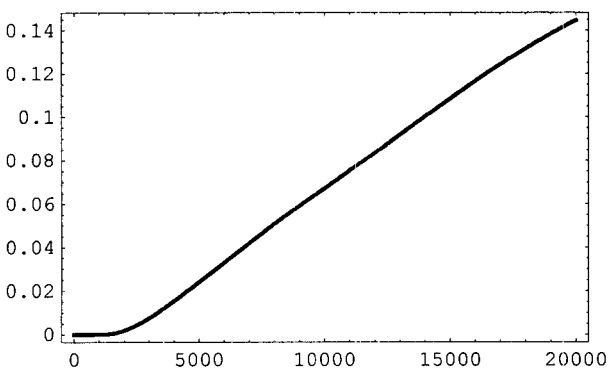
a=500

```
Q = 252;
nreal = 150;
a = 500;
ldist = LogNormalDistribution[9.45708, 1.10];
Rdist1 = Interpolation[ Transpose[ {{0, 0.15, 0.25, 0.5, 0.8, 1.0},
     {Log[6], Log[6], Log[10.23], Log[26.], Log[59.98], Log[800]}} ]];
Rdist2 = Interpolation[ Transpose[ {{0, 0.331, 0.5, 1.0},
     {Log[8], Log[8], Log[33.96], Log[5188.]}} ]];

bigresult = Table[
   Clear[X1, X2];
   Clear[Y1, Y2];
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   thetam = 10^Random[Real, {-3, -2}];
   f = 10^Random[Real, {-2, -1}];
   velocity = darcyv / thetam;
   tauL1 = (tddist) / velocity;
   tauL1 = Max[tauL1, 10];

   avgwidth = NIntegrate[
      cstfunc2[td], {td, tddist, 18000},
      MinRecursion → 5, MaxRecursion → 12] / (18000 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tauL2 = (18000 - tddist) / velocity;
   tauL2 = Max[tauL2, 10.];

   Rc = Exp[ Rdist1[Random[]]];
   Rs = Random[ ldist];
   {X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, tauL1];
   Share[];

   R = Rs;
   Rc = Exp[ Rdist2[Random[]]];
    {X2, Y2} = SolveAlluvium[alphaf, R, Rc, eps, a, lambda, tauL2];
   Share[];

Table[ X2[t, tauL2] + Y2[t, tauL2] , {t, 0, 20000, 100}],
   {i, 1, nreal}];

time = (Range[201] - 1) * 100;
```

```
meancurve500 = Apply[Plus, bigresult] / Length[bigresult]
```

{0., 6.36915×10^{-22}, 2.31985×10^{-11}, 1.48217×10^{-10}, 4.89112×10^{-8}, 2.04738×10^{-6},
0.0000119917, 0.0000293609, 0.0000496016, 0.0000712071, 0.0000946078, 0.000121384,
0.000155358, 0.00020337, 0.000274949, 0.000381495, 0.000534978, 0.000745623,
0.00101904, 0.00135463, 0.00174638, 0.00218557, 0.00266358, 0.00317378, 0.00371199,
0.00427615, 0.00486555, 0.0054801, 0.00611979, 0.00678439, 0.00747328, 0.00818545,
0.00891955, 0.00967396, 0.0104469, 0.0112364, 0.0120406, 0.0128577, 0.0136859,
0.0145235, 0.0153689, 0.0162208, 0.0170778, 0.0179391, 0.0188036, 0.0196706,
0.0205394, 0.0214097, 0.0222812, 0.0231537, 0.0240272, 0.0249018, 0.0257777,
0.0266552, 0.0275344, 0.0284156, 0.0292992, 0.0301853, 0.0310741, 0.0319657,
0.0328601, 0.0337571, 0.0346566, 0.0355583, 0.0364618, 0.0373667, 0.0382726,
0.0391787, 0.0400846, 0.0409897, 0.0418933, 0.042795, 0.0436942, 0.0445904,
0.0454831, 0.046372, 0.0472566, 0.0481367, 0.0490121, 0.0498825, 0.050748,
0.0516083, 0.0524636, 0.0533139, 0.0541592, 0.0549997, 0.0558357, 0.0566672,
0.0574946, 0.0583182, 0.0591382, 0.0599549, 0.0607687, 0.0615799, 0.0623889,
0.0631959, 0.0640014, 0.0648056, 0.0656089, 0.0664116, 0.0672139, 0.0680163,
0.0688187, 0.0696215, 0.070425, 0.0712292, 0.0720344, 0.0728408, 0.0736484,
0.0744573, 0.0752677, 0.0760796, 0.076893, 0.077708, 0.0785245, 0.0793426,
0.0801623, 0.0809835, 0.0818062, 0.0826303, 0.0834558, 0.0842826, 0.0851106,
0.0859398, 0.0867701, 0.0876013, 0.0884335, 0.0892664, 0.0901001, 0.0909344,
0.0917691, 0.0926043, 0.0934398, 0.0942754, 0.0951112, 0.0959469, 0.0967824,
0.0976177, 0.0984527, 0.0992871, 0.100121, 0.100954, 0.101786, 0.102618, 0.103448,
0.104277, 0.105105, 0.105931, 0.106755, 0.107578, 0.108399, 0.109218, 0.110034,
0.110849, 0.111661, 0.11247, 0.113277, 0.11408, 0.114881, 0.115679, 0.116473,
0.117264, 0.118051, 0.118835, 0.119615, 0.12039, 0.121162, 0.12193, 0.122693,
0.123453, 0.124207, 0.124957, 0.125703, 0.126443, 0.127179, 0.12791, 0.128636,
0.129357, 0.130073, 0.130784, 0.131489, 0.13219, 0.132885, 0.133575, 0.13426,
0.134939, 0.135613, 0.136282, 0.136945, 0.137603, 0.138256, 0.138904, 0.139546,
0.140183, 0.140815, 0.141441, 0.142062, 0.142678, 0.143289, 0.143895, 0.144496}

```
ListPlot[Transpose[{time, meancurve500}],
 PlotJoined → True, PlotStyle → AbsoluteThickness[2], Frame → True]
```



- Graphics -

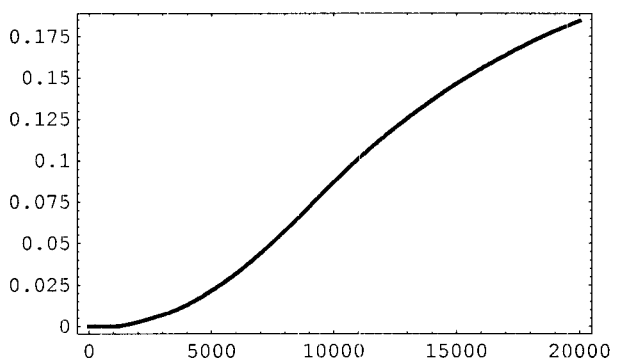
a=5000

```
Q = 252;
nreal = 200;
a = 5000;
ldist = LogNormalDistribution[9.45708, 1.10];
Rdist1 = Interpolation[ Transpose[ {{0, 0.15, 0.25, 0.5, 0.8, 1.0},
     {Log[6], Log[6], Log[10.23], Log[26.], Log[59.98], Log[800]}} ]];
Rdist2 = Interpolation[ Transpose[ {{0, 0.331, 0.5, 1.0},
     {Log[8], Log[8], Log[33.96], Log[5188.]}} ]];

bigresult = Table[
   Clear[X1, X2];
   Clear[Y1, Y2];
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   thetam = 10^Random[Real, {-3, -2}];
   f = 10^Random[Real, {-2, -1}];
   velocity = darcyv / thetam;
   tauL1 = (tddist) / velocity;
   tauL1 = Max[tauL1, 10];

   avgwidth = NIntegrate[
      cstfunc2[td], {td, tddist, 18000},
      MinRecursion → 5, MaxRecursion → 12] / (18000 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tauL2 = (18000 - tddist) / velocity;
   tauL2 = Max[tauL2, 10.];

   Rc = Exp[ Rdist1[Random[]]];
   Rs = Random[ ldist];
   {X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, tauL1];
   Share[];

   R = Rs;
   Rc = Exp[ Rdist2[Random[]]];
    {X2, Y2} = SolveAlluvium[alphaf, R, Rc, eps, a, lambda, tauL2];
   Share[];

Table[ X2[t, tauL2] + Y2[t, tauL2] , {t, 0, 20000, 100}],
   {i, 1, nreal}];

time = (Range[201] - 1) * 100;
```

```
meancurve5000 = Apply[Plus, bigresult] / Length[bigresult]
```

{0., 2.09731×10^{-14}, 1.51475×10^{-10}, 8.62959×10^{-8}, 2.83497×10^{-6}, 0.0000110034,
0.0000205802, 0.0000302842, 0.0000407465, 0.0000539146, 0.00007401, 0.00011247,
0.00018936, 0.000322007, 0.000513944, 0.000757387, 0.00104212, 0.0013602, 0.00170603,
0.00207519, 0.0024638, 0.00286819, 0.00328472, 0.00370976, 0.00413996, 0.00457272,
0.00500667, 0.00544192, 0.00588018, 0.00632471, 0.00677992, 0.00725096, 0.0077432,
0.00826158, 0.00881022, 0.00939206, 0.0100087, 0.0106604, 0.0113463, 0.0120647,
0.0128132, 0.0135891, 0.0143897, 0.0152127, 0.0160559, 0.0169175, 0.0177963,
0.0186915, 0.0196026, 0.0205295, 0.0214722, 0.022431, 0.0234063, 0.0243985,
0.0254078, 0.0264345, 0.0274787, 0.0285407, 0.0296202, 0.0307172, 0.0318315,
0.0329628, 0.0341107, 0.0352748, 0.0364548, 0.0376504, 0.0388612, 0.040087,
0.0413275, 0.0425825, 0.0438519, 0.0451356, 0.0464337, 0.0477462, 0.049073,
0.0504141, 0.0517696, 0.0531393, 0.0545232, 0.0559209, 0.0573321, 0.0587563,
0.0601929, 0.0616411, 0.0631001, 0.0645688, 0.0660461, 0.0675307, 0.0690213,
0.0705164, 0.0720148, 0.0735148, 0.075015, 0.0765139, 0.0780103, 0.0795027,
0.0809898, 0.0824706, 0.083944, 0.085409, 0.0868647, 0.0883106, 0.0897459, 0.0911701,
0.092583, 0.0939841, 0.0953732, 0.0967503, 0.0981153, 0.0994682, 0.100809, 0.102138,
0.103455, 0.104761, 0.106055, 0.107337, 0.108609, 0.10987, 0.11112, 0.11236,
0.11359, 0.11481, 0.116019, 0.11722, 0.11841, 0.119591, 0.120763, 0.121926, 0.12308,
0.124224, 0.12536, 0.126487, 0.127605, 0.128715, 0.129816, 0.130908, 0.131991,
0.133066, 0.134132, 0.13519, 0.136239, 0.13728, 0.138311, 0.139335, 0.140349,
0.141355, 0.142353, 0.143341, 0.144321, 0.145293, 0.146255, 0.14721, 0.148155,
0.149092, 0.15002, 0.150939, 0.15185, 0.152752, 0.153646, 0.154531, 0.155408,
0.156276, 0.157135, 0.157987, 0.15883, 0.159664, 0.160491, 0.161309, 0.162119,
0.162922, 0.163716, 0.164502, 0.16528, 0.166051, 0.166814, 0.16757, 0.168318,
0.169058, 0.169791, 0.170517, 0.171236, 0.171947, 0.172652, 0.173349, 0.17404,
0.174723, 0.1754, 0.17607, 0.176734, 0.177391, 0.178042, 0.178686, 0.179323,
0.179955, 0.18058, 0.181199, 0.181812, 0.182419, 0.18302, 0.183615, 0.184204}

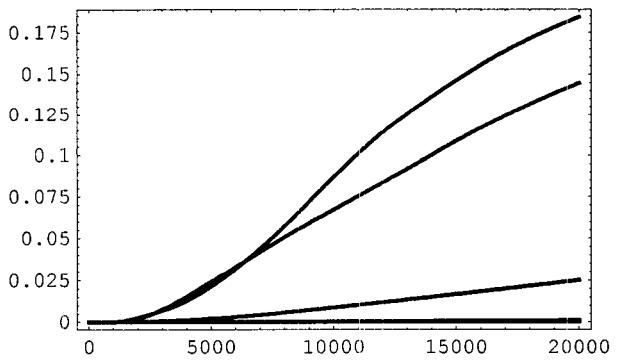
```
ListPlot[ Transpose[{time, meancurve5000}],
 PlotJoined → True, PlotStyle → AbsoluteThickness[2] , Frame → True ]
```



- Graphics -

Plot Together

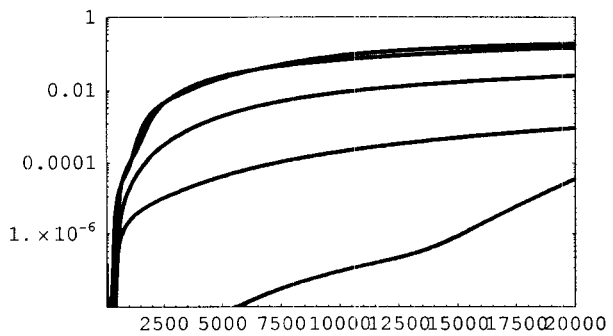
```
DisplayTogether[
 ListPlot[ Transpose[{time, meancurve5}],
  PlotJoined → True, PlotStyle → AbsoluteThickness[2] , Frame → True ],
 ListPlot[ Transpose[{time, meancurve10}], PlotJoined → True,
  PlotStyle → AbsoluteThickness[2] , Frame → True ],
 ListPlot[ Transpose[{time, meancurve50}], PlotJoined → True,
  PlotStyle → AbsoluteThickness[2] , Frame → True ],
 ListPlot[ Transpose[{time, meancurve500}], PlotJoined → True,
  PlotStyle → AbsoluteThickness[2] , Frame → True ],
 ListPlot[ Transpose[{time, meancurve5000}], PlotJoined → True,
  PlotStyle → AbsoluteThickness[2] , Frame → True ]
]
```



- Graphics -

ESP 1/18/2010

```
DisplayTogether[
 LogListPlot[ Transpose[{time, meancurve5}],
  PlotJoined → True, PlotStyle → AbsoluteThickness[2] , Frame → True ],
 LogListPlot[ Transpose[{time, meancurve10}], PlotJoined → True,
  PlotStyle → AbsoluteThickness[2] , Frame → True ],
 LogListPlot[ Transpose[{time, meancurve50}], PlotJoined → True,
  PlotStyle → AbsoluteThickness[2] , Frame → True ],
 LogListPlot[ Transpose[{time, meancurve500}], PlotJoined → True,
  PlotStyle → AbsoluteThickness[2] , Frame → True ],
 LogListPlot[ Transpose[{time, meancurve5000}], PlotJoined → True,
  PlotStyle → AbsoluteThickness[2] , Frame → True ],
 PlotRange → {{50, 20000}, {-8, 0}}
]
```



- Graphics -

RCP 1/18/2010

Colloid Facilitated Transport in Tuff

Scott Painter
5.27.04

Preliminaries

```
<< Statistics`DescriptiveStatistics`
<< Graphics`Graphics`
<< Statistics`ContinuousDistributions`
```

Set Functions for defining tau distribution

These are based on TPA 5.0 input

Central Streamtube

```
centralst = Partition[{0.0, 1500., 0.,
1.0, 1500., 1350.,
3.0, 450., 4350.,
9.5, 250., 10850.,
13.0, 400., 14350.,
13.5 , 375., 14850.,
15.0, 325., 16350.,
18.0 , 225., 19350.}, 3] ;
centralst = Transpose[centralst];
centralst1 = Transpose[{centralst[[1]], centralst[[3]]}]
centralst2 = Transpose[{centralst[[3]], centralst[[2]]}]
```

```
{{0., 0.}, {1., 1350.}, {3., 4350.}, {9.5, 10850.},
 {13., 14350.}, {13.5, 14850.}, {15., 16350.}, {18., 19350.}}

{{0., 1500.}, {1350., 1500.}, {4350., 450.}, {10850., 250.},
 {14350., 400.}, {14850., 375.}, {16350., 325.}, {19350., 225.}}
```

```
cstfunc1 = Interpolation[centralst1]
cstfunc2 = Interpolation[centralst2]
```

```
InterpolatingFunction[{{0., 18.}}, <>]

InterpolatingFunction[{{0., 19350.}}, <>]
```

Define the breakthrough solution

rates are in units of 1/year
u is aqueous concentration
v is concentration sorbed to colloids
w is concentration sorbed to permanently immobilized colloids
x represents travel time
t is time
other parameters same as in paper

```
SolveTuff[alphaf_, Rc_, a_, Rs_,
  f_, thetam_, lambda_, tauL_] := Module[
  {alphar = alphaf / a,
Deff = 0.01 * tauL,
   dcoef = 0.001,
   thetaim = 0.2,
   n = 0.05,
   beta0, X, Y},
  beta0 = dcoef / (0.28 * (f / 2 n) ^2);
  omega = f thetaim / thetam;
solution = NDSolve[
    {∂t u[t, x] == -∂x u[t, x] + Deff ∂x,x u[t, x] +
       -beta0 * (omega * u[t, x] - w[t, x])  - alphaf Rc
        u[t, x] + alphar Rc v[t, x] - lambda u[t, x],
     Rc ∂t v[t, x] == -∂x v[t, x] + Deff ∂x,x v[t, x] +
       alphaf Rc u[t, x] -
       alphar Rc v[t, x] - lambda Rc v[t, x] ,
     Rs ∂t w[t, x] == beta0 * (omega * u[t, x] - w[t, x]) -
       lambda Rs w[t, x],
     u[0, x] == 0.,
     u[t, 0] == Min[t / 10., 1],
     u[t, 5 * tauL ] == 0,
```

```
      v[0, x] == 0.,
      v[t, 5 * tauL] == 0.,
      v[t, 0] == 0.,
      w[0, x] == 0.},
     {u, v, w}, {t, 0, 20000}, {x, 0, 5 * tauL},
     StartingStepSize → {1, 5 * tauL / 5000},
     AccuracyGoal → 6,
     MaxStepSize → {10, 5 tauL / 500},
     MaxSteps → {10000, 10000}];
{X, Y} = {u, v} /. First[solution]
 ]
```

```
General::spell1 :
 Possible spelling error: new symbol name "alphar" is similar to existing symbol "alphaf". More…

General::spell1 :
 Possible spelling error: new symbol name "thetaim" is similar to existing symbol "thetam". More…
```

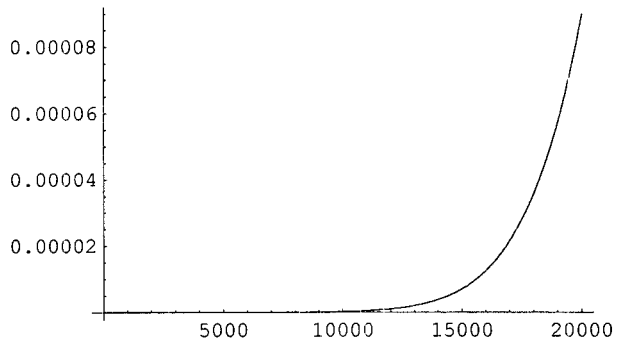
Reference Case Parameters

```
alphaf = 0.1;
Rc = 20;
a = 5;
lambda = 2.88 10^-5;
f = 0.032;
Rs = 1800;
thetam = 0.0032;
```

```
{X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, 100]
```

```
{InterpolatingFunction[{{0., 20000.}, {0., 500.}}, <>],
 InterpolatingFunction[{{0., 20000.}, {0., 500.}}, <>]}
```

```
result = Table[ X1[t, 100] + Y1[t, 100] , {t, 0, 20000, 100}];
time = (Range[201] - 1) * 100;
ListPlot[ Transpose[{time, result}], PlotJoined → True, PlotRange → All]
```



- Graphics -

Central Streamtube: a=5

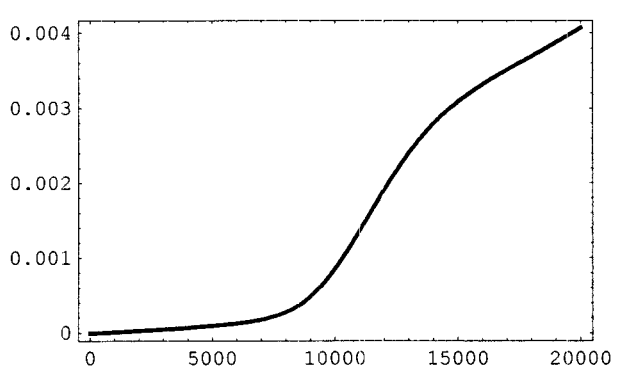
300 realizations. Sample travel time, porosity, retardation R, and colloid retardation Rc.

```
Q = 252;
nreal = 300;
ldist = LogNormalDistribution[9.45708, 1.10];
Rdist = Interpolation[ Transpose[ {{0, 0.15, 0.25, 0.5, 0.8, 1.0},
     {Log[6], Log[6], Log[10.23], Log[26.], Log[59.98], Log[800]}} ]];
bigresult = Table[
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   thetam = 10^Random[Real, {-3, -2}];
   f = 10^Random[Real, {-2, -1}];
   velocity = darcyv / thetam;
   tauL = (tddist) / velocity;
   Rc = Exp[ Rdist[Random[]]];
   Rs = Random[ ldist];
{X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, tauL];
Table[ X1[t, tauL] + Y1[t, tauL] , {t, 0, 20000, 100}],
   {i, 1, nreal}];

time = (Range[201] - 1) * 100;
```

```
meancurve5 = Apply[Plus, bigresult] / Length[bigresult]
```

{0., 2.93635×10^{-11}, 2.65281×10^{-7}, 1.39972×10^{-6}, 2.83023×10^{-6}, 4.31537×10^{-6},
5.82039×10^{-6}, 7.34243×10^{-6}, 8.88086×10^{-6}, 0.0000104357, 0.0000120073, 0.0000135958,
0.0000152015, 0.0000168249, 0.0000184661, 0.0000201256, 0.0000218038, 0.0000235011,
0.0000252179, 0.0000269548, 0.0000287122, 0.0000304906, 0.0000322907, 0.0000341129,
0.000035958, 0.0000378265, 0.0000397192, 0.0000416368, 0.0000435799, 0.0000455494,
0.0000475461, 0.0000495707, 0.0000516243, 0.0000537076, 0.0000558216, 0.0000579673,
0.0000601457, 0.0000623577, 0.0000646045, 0.0000668871, 0.0000692067, 0.0000715645,
0.0000739617, 0.0000763997, 0.0000788798, 0.0000814035, 0.0000839725, 0.0000865884,
0.0000892533, 0.0000919693, 0.0000947392, 0.0000975657, 0.000100452, 0.000103403,
0.000106423, 0.000109518, 0.000112694, 0.00011596, 0.000119325, 0.0001228, 0.0001264,
0.000130138, 0.000134033, 0.000138107, 0.000142381, 0.000146884, 0.000151646,
0.000156701, 0.000162086, 0.000167844, 0.00017402, 0.000180664, 0.00018783,
0.000195576, 0.000203963, 0.000213055, 0.000222919, 0.000233625, 0.000245245,
0.000257851, 0.000271518, 0.000286318, 0.000302323, 0.000319605, 0.000338232,
0.000358267, 0.000379771, 0.000402799, 0.000427399, 0.000453614, 0.000481479,
0.00051102, 0.000542254, 0.000575192, 0.000609831, 0.000646163, 0.000684166,
0.000723812, 0.000765059, 0.00080786, 0.000852156, 0.000897879, 0.000944956,
0.000993301, 0.00104283, 0.00109344, 0.00114503, 0.0011975, 0.00125073, 0.00130462,
0.00135905, 0.00141391, 0.00146908, 0.00152444, 0.00157989, 0.0016353, 0.00169059,
0.00174564, 0.00180034, 0.00185462, 0.00190838, 0.00196154, 0.00201401, 0.00206574,
0.00211665, 0.00216669, 0.0022158, 0.00226395, 0.00231109, 0.0023572, 0.00240225,
0.00244621, 0.00248909, 0.00253087, 0.00257154, 0.00261112, 0.00264961, 0.00268702,
0.00272337, 0.00275867, 0.00279296, 0.00282625, 0.00285857, 0.00288996, 0.00292045,
0.00295006, 0.00297884, 0.00300681, 0.00303402, 0.0030605, 0.0030863, 0.00311143,
0.00313595, 0.00315989, 0.00318327, 0.00320615, 0.00322855, 0.0032505, 0.00327205,
0.00329321, 0.00331401, 0.0033345, 0.00335469, 0.00337461, 0.00339428, 0.00341373,
0.00343299, 0.00345207, 0.003471, 0.00348979, 0.00350845, 0.00352702, 0.0035455,
0.00356391, 0.00358226, 0.00360057, 0.00361884, 0.00363709, 0.00365533, 0.00367356,
0.0036918, 0.00371006, 0.00372834, 0.00374664, 0.00376498, 0.00378336, 0.00380178,
0.00382025, 0.00383878, 0.00385737, 0.00387601, 0.00389472, 0.00391349, 0.00393234,
0.00395125, 0.00397024, 0.00398931, 0.00400845, 0.00402767, 0.00404696, 0.00406634}

```
ListPlot[Transpose[{time, meancurve5}],
 PlotJoined → True, PlotStyle → AbsoluteThickness[2], Frame → True]
```



- Graphics -

Central Streamtube: a=50

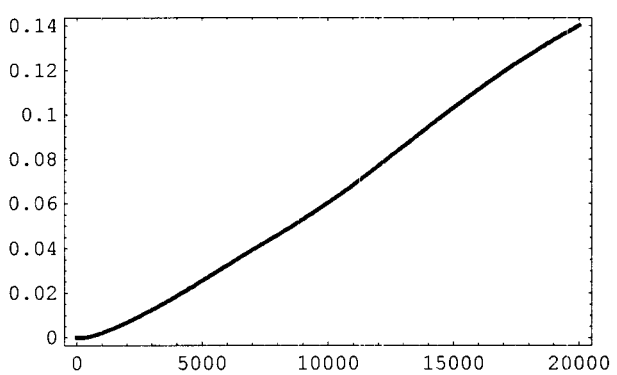
```
Q = 252;
nreal = 300;
a = 50;
ldist = LogNormalDistribution[9.45708, 1.10];
Rdist = Interpolation[ Transpose[ {{0, 0.15, 0.25, 0.5, 0.8, 1.0},
     {Log[6], Log[6], Log[10.23], Log[26.], Log[59.98], Log[800]}} ]];
bigresult = Table[
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   thetam = 10^Random[Real, {-3, -2}];
   f = 10^Random[Real, {-2, -1}];
   velocity = darcyv / thetam;
   tauL = (tddist) / velocity;
   Rc = Exp[ Rdist[Random[]]];
   Rs = Random[ ldist];
{X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, tauL];
Table[ X1[t, tauL] + Y1[t, tauL] , {t, 0, 20000, 100}],
   {i, 1, nreal}];

time = (Range[201] - 1) * 100;
```

ECP 1/18/2010

```
meancurve50 = Apply[Plus, bigresult] / Length[bigresult]
```

{0., 6.48762×10^{-11}, 5.31012×10^{-6}, 0.0000505193, 0.000148621, 0.000310591, 0.000551916, 0.000857271, 0.00120056, 0.00157215, 0.00196785, 0.00238323, 0.00281488, 0.00326058, 0.003719, 0.00418958, 0.00467223, 0.00516688, 0.00567333, 0.00619114, 0.00671973, 0.00725847, 0.00780672, 0.00836386, 0.00892935, 0.00950263, 0.0100832, 0.0106704, 0.0112638, 0.0118629, 0.012467, 0.0130756, 0.0136884, 0.0143048, 0.0149247, 0.0155477, 0.0161739, 0.0168031, 0.0174356, 0.0180717, 0.0187115, 0.0193557, 0.0200046, 0.0206586, 0.0213181, 0.0219835, 0.022655, 0.0233326, 0.0240163, 0.0247058, 0.0254008, 0.0261008, 0.0268051, 0.027513, 0.0282236, 0.0289361, 0.0296495, 0.0303631, 0.0310759, 0.0317873, 0.0324964, 0.0332026, 0.0339056, 0.0346048, 0.0353001, 0.0359914, 0.0366784, 0.0373615, 0.0380406, 0.0387162, 0.0393884, 0.0400578, 0.0407248, 0.04139, 0.0420537, 0.0427167, 0.0433794, 0.0440424, 0.0447063, 0.0453718, 0.0460392, 0.0467091, 0.047382, 0.0480584, 0.0487387, 0.0494234, 0.0501127, 0.050807, 0.0515066, 0.0522117, 0.0529226, 0.0536396, 0.0543626, 0.0550919, 0.0558275, 0.0565695, 0.0573179, 0.0580728, 0.0588341, 0.0596017, 0.0603756, 0.0611558, 0.061942, 0.0627343, 0.0635323, 0.0643361, 0.0651454, 0.06596, 0.0667797, 0.0676045, 0.068434, 0.069268, 0.0701064, 0.0709489, 0.0717954, 0.0726455, 0.0734991, 0.0743559, 0.0752158, 0.0760785, 0.0769437, 0.0778113, 0.0786811, 0.0795528, 0.0804262, 0.0813011, 0.0821773, 0.0830546, 0.0839328, 0.0848116, 0.0856909, 0.0865705, 0.0874501, 0.0883296, 0.0892089, 0.0900875, 0.0909656, 0.0918427, 0.0927187, 0.0935936, 0.094467, 0.0953388, 0.0962089, 0.0970771, 0.0979431, 0.098807, 0.0996684, 0.100527, 0.101383, 0.102237, 0.103087, 0.103934, 0.104778, 0.105619, 0.106456, 0.10729, 0.108119, 0.108945, 0.109767, 0.110585, 0.111399, 0.112209, 0.113014, 0.113815, 0.114612, 0.115404, 0.116191, 0.116974, 0.117752, 0.118525, 0.119294, 0.120058, 0.120816, 0.12157, 0.122319, 0.123064, 0.123803, 0.124537, 0.125266, 0.12599, 0.126709, 0.127423, 0.128132, 0.128836, 0.129535, 0.130229, 0.130918, 0.131602, 0.132281, 0.132955, 0.133625, 0.134289, 0.134948, 0.135603, 0.136252, 0.136897, 0.137537, 0.138173, 0.138803, 0.139429, 0.140051}

```
ListPlot[ Transpose[{time, meancurve50}],
 PlotJoined → True, PlotStyle → AbsoluteThickness[2] , Frame → True ]
```



- Graphics -

Central Streamtube: a=500

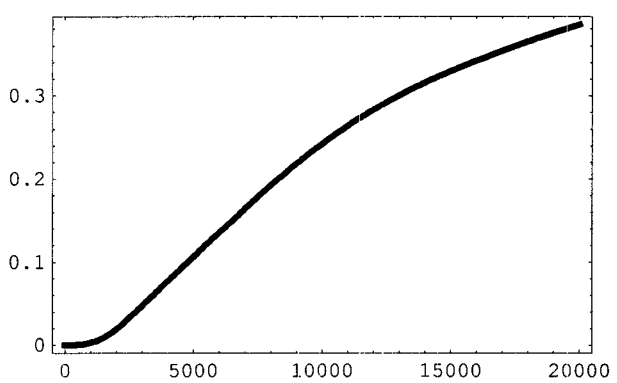
```
Q = 252;
nreal = 300;
a = 500;
ldist = LogNormalDistribution[9.45708, 1.10];
Rdist = Interpolation[ Transpose[ {{0, 0.15, 0.25, 0.5, 0.8, 1.0},
     {Log[6], Log[6], Log[10.23], Log[26.], Log[59.98], Log[800]}} ]];
bigresult = Table[
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   thetam = 10^Random[Real, {-3, -2}];
   f = 10^Random[Real, {-2, -1}];
   velocity = darcyv / thetam;
   tauL = (tddist) / velocity;
   Rc = Exp[ Rdist[Random[]]];
   Rs = Random[ ldist];
{X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, tauL];
Table[ X1[t, tauL] + Y1[t, tauL] , {t, 0, 20000, 100}],
   {i, 1, nreal}];

time = (Range[201] - 1) * 100;
```

RCP 1/18/2010

```
meancurve500 = Apply[Plus, bigresult] / Length[bigresult]
```

{0., 1.48455×10^{-11}, 1.00878×10^{-6}, 0.0000195531, 0.0000848927, 0.000255477, 0.000532184,
0.000921716, 0.00143652, 0.00207356, 0.00283671, 0.00373203, 0.0047742, 0.00598367,
0.00737043, 0.00892972, 0.010652, 0.0125328, 0.014573, 0.0167747, 0.0191346,
0.0216409, 0.0242739, 0.0270088, 0.0298199, 0.0326844, 0.0355839, 0.0385052,
0.0414392, 0.0443802, 0.0473248, 0.0502712, 0.0532191, 0.0561687, 0.0591209,
0.0620768, 0.0650371, 0.0680022, 0.0709718, 0.0739451, 0.0769204, 0.0798958,
0.0828689, 0.0858373, 0.0887988, 0.0917515, 0.0946939, 0.097625, 0.100544,
0.103452, 0.106348, 0.109235, 0.112112, 0.114982, 0.117846, 0.120706, 0.123563,
0.126419, 0.129275, 0.132132, 0.134991, 0.137853, 0.140716, 0.143582, 0.14645,
0.149319, 0.152188, 0.155057, 0.157925, 0.160789, 0.163649, 0.166503, 0.16935,
0.172188, 0.175017, 0.177834, 0.180638, 0.183428, 0.186204, 0.188963, 0.191705,
0.194428, 0.197133, 0.199818, 0.202482, 0.205125, 0.207746, 0.210346, 0.212922,
0.215476, 0.218007, 0.220514, 0.222998, 0.225458, 0.227895, 0.230307, 0.232696,
0.235061, 0.237402, 0.239718, 0.242011, 0.24428, 0.246526, 0.248747, 0.250945,
0.253119, 0.255269, 0.257396, 0.2595, 0.26158, 0.263637, 0.265672, 0.267683,
0.269671, 0.271637, 0.27358, 0.275501, 0.277399, 0.279276, 0.281131, 0.282964,
0.284776, 0.286567, 0.288337, 0.290086, 0.291815, 0.293525, 0.295214, 0.296885,
0.298536, 0.300169, 0.301783, 0.303379, 0.304958, 0.30652, 0.308064, 0.309593,
0.311105, 0.312601, 0.314082, 0.315548, 0.316999, 0.318436, 0.319859, 0.321269,
0.322665, 0.324048, 0.325419, 0.326778, 0.328124, 0.329459, 0.330783, 0.332096,
0.333397, 0.334689, 0.33597, 0.337241, 0.338502, 0.339753, 0.340995, 0.342228,
0.343452, 0.344667, 0.345873, 0.347071, 0.34826, 0.349441, 0.350614, 0.351779,
0.352935, 0.354084, 0.355225, 0.356358, 0.357483, 0.358601, 0.359711, 0.360813,
0.361908, 0.362996, 0.364075, 0.365148, 0.366212, 0.36727, 0.368319, 0.369362,
0.370396, 0.371423, 0.372443, 0.373455, 0.374459, 0.375456, 0.376445, 0.377426,
0.378399, 0.379365, 0.380323, 0.381274, 0.382216, 0.383151, 0.384078, 0.384997}

```
ListPlot[ Transpose[{time, meancurve500}],
 PlotJoined → True, PlotStyle → AbsoluteThickness[3] , Frame → True ]
```



- Graphics -

Central Streamtube: Rc variability and a variability

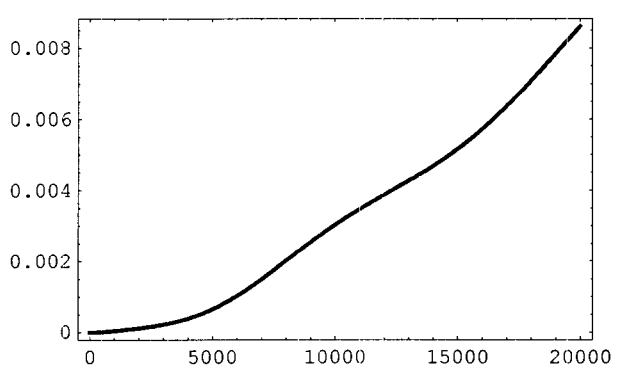
```
Q = 252;
nreal = 300;
Rdist = Interpolation[ Transpose[ {{0, 0.15, 0.25, 0.5, 0.8, 1.0},
     {Log[6], Log[6], Log[10.23], Log[26.], Log[59.98], Log[800]}} ]];
adist = ExponentialDistribution[1. / 4.];
bigresult = Table[
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   thetam = 10^Random[Real, {-3, -2}];
   f = 10^Random[Real, {-2, -1}];
   velocity = darcyv / thetam;
   tauL = (tddist) / velocity;
   Rc = Exp[ Rdist[Random[]]];
   a = Random[ adist] + 1;
   Rs = Random[ ldist];
{X1, Y1} = SolveTuff[alphaf, Rc, a, Rs, f, thetam, lambda, tauL];
Table[ X1[t, tauL] + Y1[t, tauL] , {t, 0, 20000, 100}],
   {i, 1, nreal}];

time = (Range[201] - 1) * 100;
```

RCS 1/18/2010

```
meancurveA = Apply[Plus, bigresult] / Length[bigresult]
```

{0., 6.67582×10^{-12}, 3.52571×10^{-7}, 2.50425×10^{-6}, 6.05946×10^{-6}, 0.0000108586, 0.0000160426,
0.0000214451, 0.0000271854, 0.000033316, 0.0000397355, 0.0000463377, 0.0000530791,
0.0000599532, 0.0000669658, 0.000074127, 0.0000814496, 0.0000889492, 0.0000966437,
0.000104554, 0.000112702, 0.000121115, 0.00012982, 0.000138848, 0.000148231,
0.000158004, 0.000168204, 0.000178869, 0.00019004, 0.000201757, 0.000214065,
0.000227004, 0.00024062, 0.000254955, 0.000270052, 0.000285955, 0.000302705,
0.000320342, 0.000338904, 0.00035843, 0.000378953, 0.000400506, 0.000423118,
0.000446815, 0.000471621, 0.000497555, 0.000524635, 0.000552873, 0.000582278,
0.000612855, 0.000644606, 0.000677528, 0.000711616, 0.000746858, 0.000783241,
0.000820748, 0.000859358, 0.000899045, 0.000939783, 0.00098154, 0.00102428,
0.00106797, 0.00111258, 0.00115805, 0.00120434, 0.00125141, 0.00129922, 0.00134771,
0.00139684, 0.00144655, 0.0014968, 0.00154752, 0.00159868, 0.00165022, 0.00170208,
0.00175422, 0.00180658, 0.00185912, 0.00191178, 0.00196452, 0.0020173, 0.00207006,
0.00212277, 0.00217538, 0.00222786, 0.00228016, 0.00233224, 0.00238409, 0.00243565,
0.00248691, 0.00253784, 0.0025884, 0.00263859, 0.00268837, 0.00273773, 0.00278666,
0.00283513, 0.00288315, 0.00293069, 0.00297776, 0.00302435, 0.00307046, 0.00311608,
0.00316123, 0.00320589, 0.0032501, 0.00329384, 0.00333713, 0.00337999, 0.00342242,
0.00346446, 0.0035061, 0.00354738, 0.00358832, 0.00362893, 0.00366925, 0.00370929,
0.0037491, 0.00378869, 0.0038281, 0.00386736, 0.0039065, 0.00394555, 0.00398455,
0.00402353, 0.00406254, 0.0041016, 0.00414075, 0.00418003, 0.00421948, 0.00425913,
0.00429902, 0.00433919, 0.00437968, 0.00442053, 0.00446176, 0.00450342, 0.00454555,
0.00458817, 0.00463132, 0.00467504, 0.00471935, 0.00476429, 0.00480989, 0.00485617,
0.00490316, 0.0049509, 0.00499939, 0.00504867, 0.00509875, 0.00514965, 0.0052014,
0.00525399, 0.00530746, 0.0053618, 0.00541703, 0.00547315, 0.00553017, 0.00558809,
0.00564691, 0.00570663, 0.00576725, 0.00582875, 0.00589114, 0.0059544, 0.00601852,
0.00608349, 0.00614929, 0.00621591, 0.00628332, 0.00635151, 0.00642045, 0.00649012,
0.00656049, 0.00663154, 0.00670324, 0.00677556, 0.00684848, 0.00692195, 0.00699595,
0.00707045, 0.00714541, 0.0072208, 0.00729659, 0.00737274, 0.00744921, 0.00752598,
0.007603, 0.00768024, 0.00775766, 0.00783524, 0.00791293, 0.00799071, 0.00806854,
0.00814638, 0.0082242, 0.00830197, 0.00837967, 0.00845725, 0.0085347, 0.00861197}

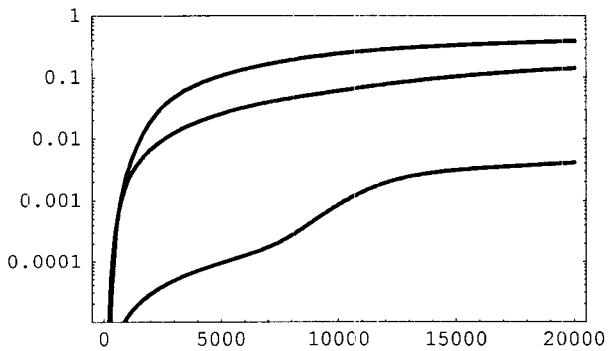
```
ListPlot[Transpose[{time, meancurveA}],
 PlotJoined → True, PlotStyle → AbsoluteThickness[2], Frame → True]
```



- Graphics -

Plot Together

```
DisplayTogether [
 LogListPlot[ Transpose[{time, meancurve5}],
  PlotJoined → True, PlotStyle → AbsoluteThickness[2] ],
 LogListPlot[ Transpose[{time, meancurve50}], PlotJoined → True,
  PlotStyle → AbsoluteThickness[2] ],
 LogListPlot[ Transpose[{time, meancurve500}], PlotJoined → True,
  PlotStyle → AbsoluteThickness[2] ] , Frame → True, PlotRange → {-5, 0} ]
```



- Graphics -

RS 1/18/2010

Tau distribution in Alluvium, Tuff and Total using TPA (V5.0) input

Scott Painter
6.02.02

Preliminaries

```
<< Statistics`DescriptiveStatistics`
```

```
<< Graphics`Graphics`
```

■ Central Streamtube

```
centralst = Partition[{0.0, 1500., 0.,
1.0, 1500., 1350.,
3.0, 450., 4350.,
9.5, 250., 10850.,
13.0, 400., 14350.,
13.5 , 375., 14850.,
15.0, 325., 16350.,
18.0 , 225., 19350.}, 3] ;
centralst = Transpose[centralst];
centralst1 = Transpose[{centralst[[1]], centralst[[3]]}]
centralst2 = Transpose[{centralst[[3]], centralst[[2]]}]
```

```
{{0., 0.}, {1., 1350.}, {3., 4350.}, {9.5, 10850.},
 {13., 14350.}, {13.5, 14850.}, {15., 16350.}, {18., 19350.}}
```

```
{{0., 1500.}, {1350., 1500.}, {4350., 450.}, {10850., 250.},
 {14350., 400.}, {14850., 375.}, {16350., 325.}, {19350., 225.}}
```

```
cstfunc1 = Interpolation[centralst1]
cstfunc2 = Interpolation[centralst2]
```

```
InterpolatingFunction[{{0., 18.}}, <>]
```

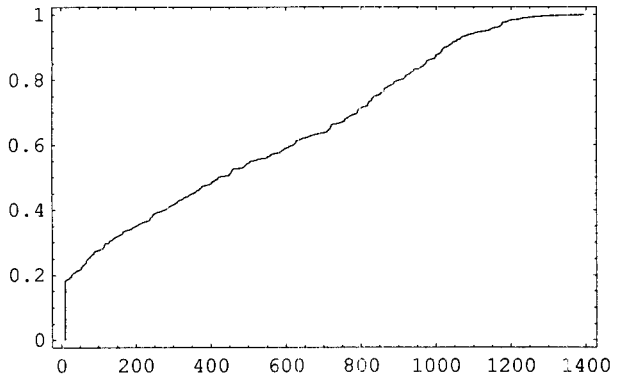
```
InterpolatingFunction[{{0., 19350.}}, <>]
```

Alluvium and Tuff with Proper Correlations

■ central streamtube

```
Q = 252;

taudist = Table[
   sldist = Random[Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, tddist, 18000},
      MinRecursion → 5, MaxRecursion → 12] / (18000 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tauA = (18000 - tddist) / velocity;

   avgwidth = NIntegrate[
      sstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   porosity = 10^Random[Real, {-3, -2}];
   velocity = darcyv / porosity;
   tauT = (tddist) / velocity;

   {Max[tauA, 10.], Max[tauT, 10.]}, {300}];

Min[taudist]

10.

Max[taudist]

1392.96

Median[taudist]

422.112

InterpolatedQuantile[taudist, 0.1]

10.

InterpolatedQuantile[taudist, 0.01]

10.
```

```
cdc = Transpose[ { Sort[taudist], (Range[Length[taudist]] - 0.5) / Length[taudist]} ];
ListPlot[ cdc, Frame → True, PlotRange → All, PlotJoined → True]
```



- Graphics -

TUFF

south streamtube

```
Q = 263;

taudist = Table[
   sldist = Random[ Real, {10, 18.}];
   tddist = sstfunc1[sldist];
avgwidth = NIntegrate[
      sstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   porosity = 10^Random[Real, {-3, -2}];
   velocity = darcyv / porosity;
   tau = (tddist) / velocity;
   Max[tau, 10.], {300}];

Min[taudist]
```

25.0862

```
Max[taudist]
```

344.954

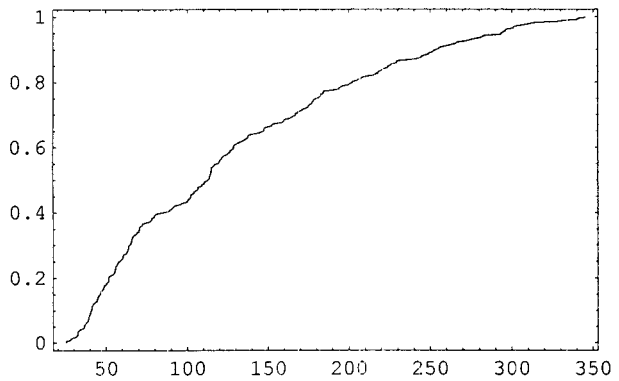
```
Median[taudist]
```

112.641

```
InterpolatedQuantile[taudist, 0.01]
```

27.9515

```
cds1 = Transpose[ { Sort[taudist], (Range[Length[taudist]] - 0.5) / Length[taudist]} ];
ListPlot[ cds1, Frame → True, PlotRange → All, PlotJoined → True, Axes → False]
```



- Graphics -

■ central streamtube

```
Q = 252;

taudist = Table[
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   porosity = 10^Random[Real, {-3, -2}];
   velocity = darcyv / porosity;
   tau = (tddist) / velocity;
   Max[tau, 10.], {300}];

Min[taudist]
```

28.0078

```
Max[taudist]
```

366.099

```
Median[taudist]
```

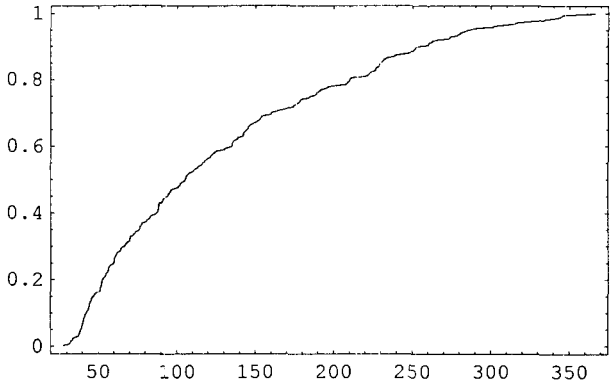
105.544

```
InterpolatedQuantile[taudist, 0.01]
```

32.1215

```
cdc1 = Transpose[ { Sort[taudist], (Range[Length[taudist]] - 0.5) / Length[taudist]} ];
ListPlot[ cdc1, Frame → True, PlotRange → All, PlotJoined → True, Axes → False]
```

```
General::spell1 :
 Possible spelling error: new symbol name "cdc1" is similar to existing symbol "cds1". More...
```



- Graphics -

north streamtube

```
Q = 155;

taudist = Table[
   sldist = Random[ Real, {10, 18.}];
   tddist = nstfunc1[sldist];
avgwidth = NIntegrate[
      nstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   porosity = 10^Random[Real, {-3, -2}];
   velocity = darcyv / porosity;
   tau = (tddist) / velocity;
   Max[tau, 10.], {300}];

Min[taudist]
```

```
48.4992
```

```
Max[taudist]
```

```
537.487
```

```
Median[taudist]
```

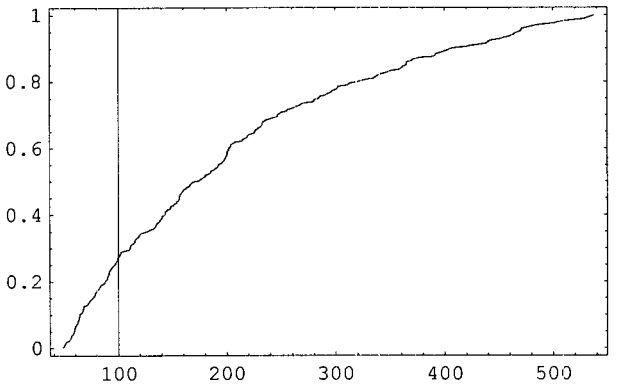
```
172.351
```

```
InterpolatedQuantile[taudist, 0.01]
```

```
50.3168
```

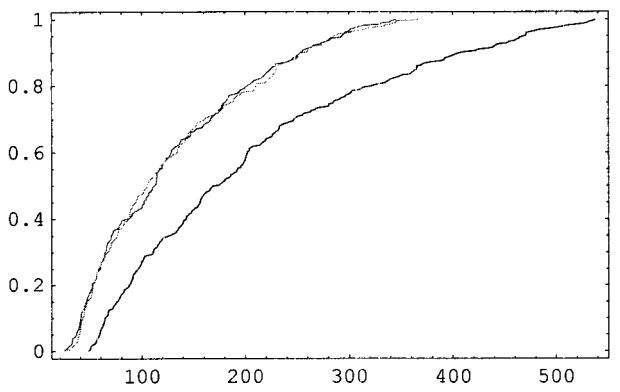
```
cdn1 = Transpose[ { Sort[taudist], (Range[Length[taudist]] - 0.5) / Length[taudist]} ];
ListPlot[ cdn1, Frame → True, PlotRange → All, PlotJoined → True]
```

```
General::spell :
 Possible spelling error: new symbol name "cdn1" is similar to existing symbols {cdc1, cds1}. More...
```



- Graphics -

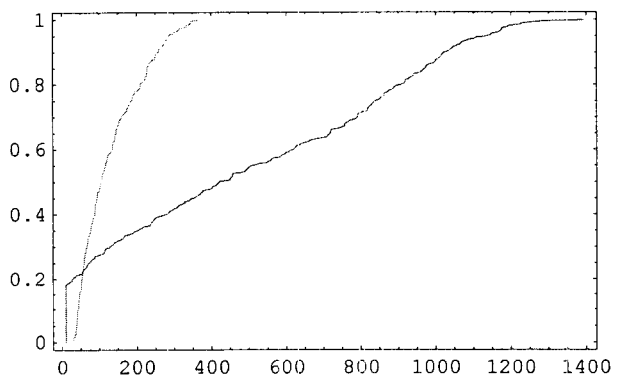
```
DisplayTogether[
 ListPlot[ cds1, Frame → True, PlotRange → All,
  PlotJoined → True, PlotStyle → RGBColor[1, 0, 0] ],
 ListPlot[ cdc1, Frame → True, PlotRange → All,
  PlotJoined → True, PlotStyle → RGBColor[0, 1, 0]],
 ListPlot[ cdn1, Frame → True, PlotRange → All, PlotJoined → True], Axes → False
]
```



- Graphics -

Display Together

```
DisplayTogether[
 ListPlot[ cdc, Frame → True, PlotRange → All,
  PlotJoined → True, PlotStyle → RGBColor[1, 0, 0] ],
 ListPlot[ cdc1, Frame → True, PlotRange → All,
  PlotJoined → True, PlotStyle → RGBColor[0, 1, 0]],
 Axes → False
]
```



- Graphics -

RS 1/18/2010

Calculating tau distribution using TPA (V5.0) input

Scott Painter
5.27.04

Preliminaries

```
<< Statistics`DescriptiveStatistics`

<< Graphics`Graphics`
```

South Streamtube

```
southst = Partition[
   {0.0 , 1400., 0.,
1.5, 1500., 1500.,
2.5, 450., 2500.,
4.0, 300., 4000.,
6.5, 225., 6500.,
9.0 , 350., 9000.,
12.5, 550., 12500.,
13, 450., 13000.,
15 , 375., 15000.,
18.0 , 220., 18000.}, 3];
southst = Transpose[southst];
southst1 = Transpose[{southst[[1]], southst[[3]]}]
southst2 = Transpose[{southst[[3]], southst[[2]]}]
```

```
{{0., 0.}, {1.5, 1500.}, {2.5, 2500.}, {4., 4000.}, {6.5, 6500.},
 {9., 9000.}, {12.5, 12500.}, {13, 13000.}, {15, 15000.}, {18., 18000.}}

{{0., 1400.}, {1500., 1500.}, {2500., 450.}, {4000., 300.}, {6500., 225.},
 {9000., 350.}, {12500., 550.}, {13000., 450.}, {15000., 375.}, {18000., 220.}}
```

```
sstfunc1 = Interpolation[southst1]
sstfunc2 = Interpolation[southst2]
```

```
InterpolatingFunction[{{0., 18.}}, <>]

InterpolatingFunction[{{0., 18000.}}, <>]
```

■ Central Streamtube

```
centralst = Partition[{0.0, 1500., 0.,
1.0, 1500., 1350.,
3.0, 450., 4350.,
9.5, 250., 10850.,
13.0, 400., 14350.,
13.5 , 375., 14850.,
15.0, 325., 16350.,
18.0 , 225., 19350.}, 3] ;
centralst = Transpose[centralst];
centralst1 = Transpose[{centralst[[1]], centralst[[3]]}]
centralst2 = Transpose[{centralst[[3]], centralst[[2]]}]
```

```
{{0., 0.}, {1., 1350.}, {3., 4350.}, {9.5, 10850.},
 {13., 14350.}, {13.5, 14850.}, {15., 16350.}, {18., 19350.}}

{{0., 1500.}, {1350., 1500.}, {4350., 450.}, {10850., 250.},
 {14350., 400.}, {14850., 375.}, {16350., 325.}, {19350., 225.}}
```

```
cstfunc1 = Interpolation[centralst1]
cstfunc2 = Interpolation[centralst2]
```

```
General::spell1 :
 Possible spelling error: new symbol name "cstfunc1" is similar to existing symbol "sstfunc1". More...

InterpolatingFunction[{{0., 18.}}, <>]

General::spell1 :
 Possible spelling error: new symbol name "cstfunc2" is similar to existing symbol "sstfunc2". More...

InterpolatingFunction[{{0., 19350.}}, <>]
```

RCP 1/18/2010

■ North Streamtube

```
northst = Partition[{0.0 , 1300., 0.,
1.5, 1200., 2200.,
2.0, 500., 4600.,
3.0 , 300., 6100.,
7.0, 225., 10600.,
10.0, 175., 13900.,
13.0, 250., 16900.,
13.5 , 225., 17400.,
15.0 , 175., 18900.,
18.0, 150., 21900.}, 3];
northst = Transpose[northst];
northst1 = Transpose[{northst[[1]], northst[[3]]}]
northst2 = Transpose[{northst[[3]], northst[[2]]}]
```

```
{{0., 0.}, {1.5, 2200.}, {2., 4600.}, {3., 6100.}, {7., 10600.},
 {10., 13900.}, {13., 16900.}, {13.5, 17400.}, {15., 18900.}, {18., 21900.}}

{{0., 1300.}, {2200., 1200.}, {4600., 500.}, {6100., 300.}, {10600., 225.},
 {13900., 175.}, {16900., 250.}, {17400., 225.}, {18900., 175.}, {21900., 150.}}
```

```
nstfunc1 = Interpolation[northst1]
nstfunc2 = Interpolation[northst2]
```

```
General::spell : Possible spelling error: new symbol
   name "nstfunc1" is similar to existing symbols {cstfunc1, sstfunc1}. More...

InterpolatingFunction[{{0., 18.}}, <>]

General::spell : Possible spelling error: new symbol
   name "nstfunc2" is similar to existing symbols {cstfunc2, sstfunc2}. More...

InterpolatingFunction[{{0., 21900.}}, <>]
```

Alluvium

■ south streamtube

```
Q = 263;
```

```
taudist = Table[
   sldist = Random[Real, {10, 18.}];
   tddist = sstfunc1[sldist];
avgwidth = NIntegrate[
      sstfunc2[td], {td, tddist, 19350},
      MinRecursion → 5, MaxRecursion → 12] / (19350 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tau = (19350 - tddist) / velocity;
   Max[tau, 10.], {300}];
```

```
Min[taudist]
```

```
10.
```

```
Max[taudist]
```

```
2179.26
```

```
Median[taudist]
```

```
670.799
```

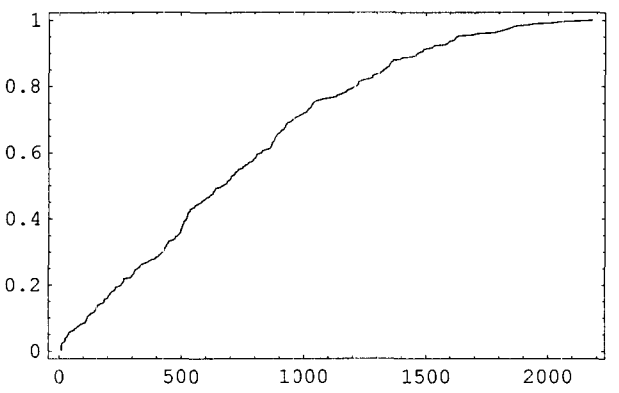
```
InterpolatedQuantile[taudist, 0.1]
```

```
117.556
```

```
InterpolatedQuantile[taudist, 0.01]
```

```
10.
```

```
cds = Transpose[ { Sort[taudist], (Range[Length[taudist]] - 0.5) / Length[taudist]} ];
ListPlot[ cds, Frame → True, PlotRange → All, PlotJoined → True]
```




```
- Graphics -
```

■ central streamtube

```
Q = 252;
```

```
taudist = Table[
   sldist = Random[Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, tddist, 18000},
      MinRecursion → 5, MaxRecursion → 12] / (18000 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tau = (18000 - tddist) / velocity;
   Max[tau, 10.], {300}];
```

```
Min[taudist]
```

```
10.
```

```
Max[taudist]
```

```
1392.96
```

```
Median[taudist]
```

```
422.112
```

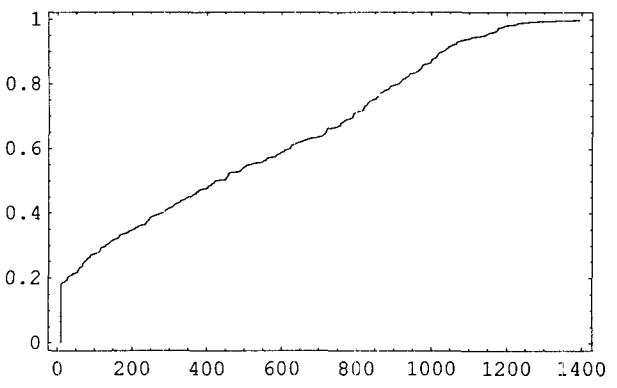
```
InterpolatedQuantile[taudist, 0.1]
```

```
10.
```

```
InterpolatedQuantile[taudist, 0.01]
```

```
10.
```

```
cdc = Transpose[ { Sort[taudist], (Range[Length[taudist]] - 0.5) / Length[taudist]} ];
ListPlot[ cdc, Frame → True, PlotRange → All, PlotJoined → True]
```




```
- Graphics -
```

north streamtube

```
Q = 155.;
```

```
taudist = Table[
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, tddist, 21900},
      MinRecursion → 5, MaxRecursion → 12] / (21900 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tau = (21900 - tddist) / velocity;
   Max[tau, 10.], {300}];
```

```
Min[taudist]
```

```
218.058
```

```
Max[taudist]
```

```
2932.72
```

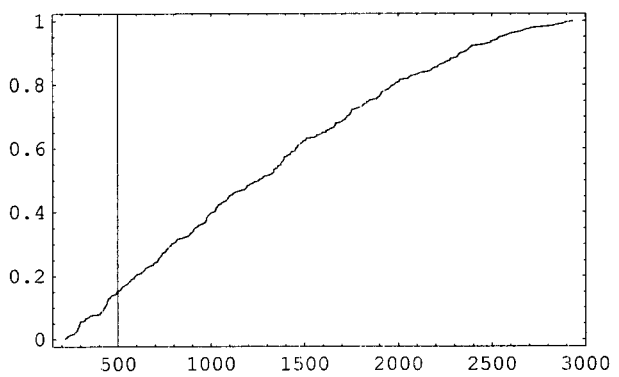
```
Median[taudist]
```

```
1255.94
```

```
InterpolatedQuantile[taudist, 0.01]
```

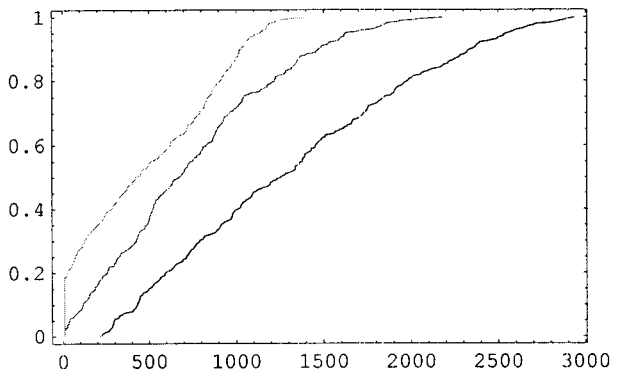
```
236.747
```

```
cdn = Transpose[ { Sort[taudist], (Range[Length[taudist]] - 0.5) / Length[taudist]} ];
ListPlot[ cdn, Frame → True, PlotRange → All, PlotJoined → True]
```



- Graphics -

```
DisplayTogether[
 ListPlot[ cds, Frame → True, PlotRange → All,
  PlotJoined → True, PlotStyle → RGBColor[1, 0, 0] ],
 ListPlot[ cdc, Frame → True, PlotRange → All,
  PlotJoined → True, PlotStyle → RGBColor[0, 1, 0]],
 ListPlot[ cdn, Frame → True, PlotRange → All, PlotJoined → True]
]
```



- Graphics -

TUFF

south streamtube

```
Q = 263;

taudist = Table[
   sldist = Random[ Real, {10, 18.}];
   tddist = sstfunc1[sldist];
avgwidth = NIntegrate[
      sstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   porosity = 10^Random[Real, {-3, -2}];
   velocity = darcyv / porosity;
   tau = (tddist) / velocity;
   Max[tau, 10.], {300}];

Min[taudist]
```

25.0862

```
Max[taudist]
```

344.954

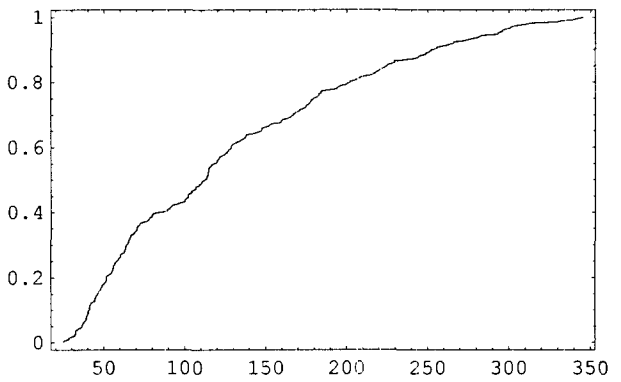
```
Median[taudist]
```

112.641

```
InterpolatedQuantile[taudist, 0.01]
```

27.9515

```
cds1 = Transpose[ { Sort[taudist], (Range[Length[taudist]] - 0.5) / Length[taudist]} ];
ListPlot[ cds1, Frame → True, PlotRange → All, PlotJoined → True, Axes → False]
```



- Graphics -

central streamtube

```
Q = 252;

taudist = Table[
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   porosity = 10^Random[Real, {-3, -2}];
   velocity = darcyv / porosity;
   tau = (tddist) / velocity;
   Max[tau, 10.], {300}];

Min[taudist]
```

28.0078

```
Max[taudist]
```

366.099

```
Median[taudist]
```

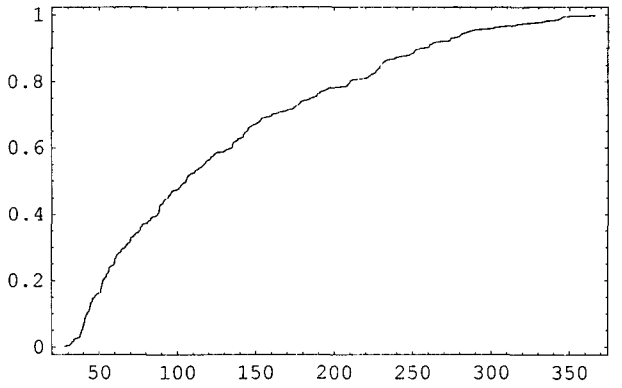
105.544

```
InterpolatedQuantile[taudist, 0.01]
```

32.1215

```
cdc1 = Transpose[ { Sort[taudist], (Range[Length[taudist]] - 0.5) / Length[taudist]} ];
ListPlot[ cdc1, Frame → True, PlotRange → All, PlotJoined → True, Axes → False]
```

General::spell1 :
Possible spelling error: new symbol name "cdc1" is similar to existing symbol "cds1". More...



- Graphics -

north streamtube

```
Q = 155;

taudist = Table[
   sldist = Random[ Real, {10, 18.}];
   tddist = nstfunc1[sldist];
avgwidth = NIntegrate[
      nstfunc2[td], {td, 0, tddist}, MinRecursion → 5, MaxRecursion → 12] / (tddist);
   darcyv = Q / avgwidth;
   porosity = 10^Random[Real, {-3, -2}];
   velocity = darcyv / porosity;
   tau = (tddist) / velocity;
   Max[tau, 10.], {300}];

Min[taudist]
```

48.4992

```
Max[taudist]
```

537.487

```
Median[taudist]
```

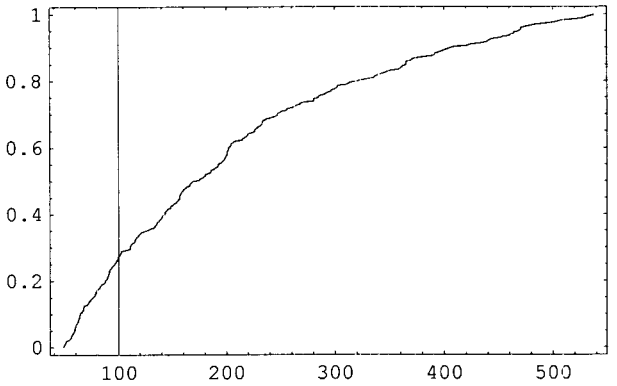
172.351

```
InterpolatedQuantile[taudist, 0.01]
```

50.3168

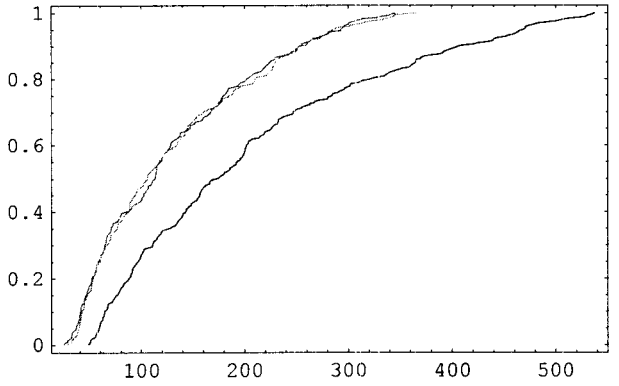
```
cdn1 = Transpose[ { Sort[taudist], (Range[Length[taudist]] - 0.5) / Length[taudist]} ];
ListPlot[ cdn1, Frame → True, PlotRange → All, PlotJoined → True]
```

General::spell :
Possible spelling error: new symbol name "cdn1" is similar to existing symbols {cdc1, cds1}. More...



- Graphics -

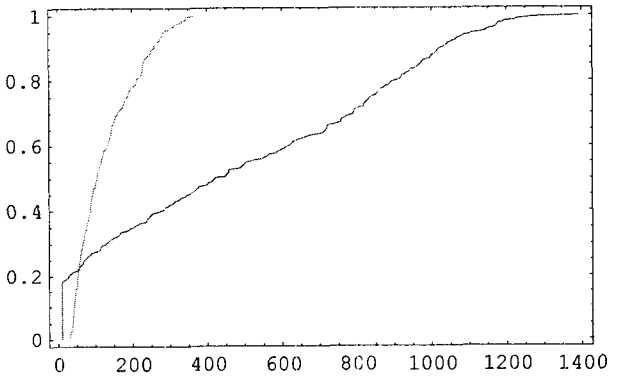
```
DisplayTogether[
 ListPlot[ cds1, Frame → True, PlotRange → All,
  PlotJoined → True, PlotStyle → RGBColor[1, 0, 0] ],
 ListPlot[ cdc1, Frame → True, PlotRange → All,
  PlotJoined → True, PlotStyle → RGBColor[0, 1, 0]],
 ListPlot[ cdn1, Frame → True, PlotRange → All, PlotJoined → True], Axes → False
]
```



- Graphics -

Display Together

```
DisplayTogether[
 ListPlot[ cdc, Frame → True, PlotRange → All,
  PlotJoined → True, PlotStyle → RGBColor[1, 0, 0] ],
 ListPlot[ cdc1, Frame → True, PlotRange → All,
  PlotJoined → True, PlotStyle → RGBColor[0, 1, 0]],
 Axes → False
]
```



- Graphics -

20 1/18/2010

SCIENTIFIC NOTEBOOK

by

Scott Painter

Southwest Research Institute
Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

June 1, 2005

Table of Contents

INITIAL ENTRIES....3
Radionuclide Transport Analysis for Yucca Mountain....4
Objective:....4
6.01.05 Reparameterization of colloid-facilitated transport in alluvium and tuff....5
6.06.05 CFT model in dimensionless form....6

INITIAL ENTRIES

Scientific notebook: #318E Vol. 11
Issued to: S. L. Painter
Original Issue Date: 22-March-1999

By agreement with CNWRA QA this notebook will be printed at approximate quarterly intervals. This computerized electronic notebook is intended to address the criteria of CNWRA QAP-001.

Radionuclide Transport Analysis for Yucca Mountain

Account number: 20-06002.01.241 and 242
Description: Radionuclide Transport Analysis.
Collaborators: Drs. D. Pickett, and V. Cvetkovic (consultant)

Objective:
Development and testing of transport analysis methodology for spatially variable systems including demonstrations and applications to the proposed repository at Yucca Mountain.

SP 1/15/2010

SN 318E Vol. 11 Pg. 4, Scott Painter

6.01.05 Reparameterization of colloid-facilitated transport in alluvium and tuff

This entry describes an alternative parameterization for colloid-facilitated transport (CFT). In our previous investigations, we considered first-order kinetic exchanges between solution and colloid-bound states. The forward rate and a "reversibility ratio" equal to the ratio of forward and reverse rates were used as parameters describing the exchange.

Note that the reversibility ratio is simply a dimensionless partitioning coefficient in the situation of fast rates. Following Contardi et al., it is reasonable to relate this partitioning coefficient to the distribution for solutes on the fixed mineral substrate, the reasoning being that colloids are representative of mineralogy in the aquifer. With this assumption, we write the "reversibility ratio" as

$$\frac{\alpha_f}{\alpha_r} \equiv K_0 = C_c K_d F$$

where Cc is colloid concentration (mass per volume water), Kd is the usual distribution coefficient (mass solute/ mass minerals) and F is the ratio of colloid surface area to porous medium surface area = 590 (Contardi et al.).

Now, the distribution coefficient Kd is related to retardation factor R in the usual way

$$R = 1 + \frac{1-\theta}{\theta} \rho K_d$$

or

$$K_0 = (R-1)\frac{\theta}{1-\theta}\frac{C_c}{\rho} F = 0.059\,(R-1)\,C_c$$

where a porosity θ=0.2, mineral density ρ=2500, and F=590 are assumed. Typically, Cc is in the range 10^{-4} to 10^{-2} kg/m^3.

Now consider the forward rate. Cvetkvoic et al. 2004 write

$\alpha_f = \alpha_0 C_c$ with α_0 in the range 16 – 876 m^3 / kg yr

6.06.05 CFT model in dimensionless form

The colloid facilitated transport model in alluvium can be made dimensionless as follows. Define a dimensionless colloid concentration as $\frac{\theta}{1-\theta}\frac{C_c}{\rho}F$ and a dimensionless intrinsic forward rate as $\frac{1-\theta}{\theta}\frac{\rho}{F}\alpha_0\tau_L$. Then multiply all terms by τ_L the total travel time in the streamtube. The CFT model then becomes

$$R\frac{\partial C}{\partial t}+TC=-\alpha_0 C_c R_c C+\frac{\alpha_0}{(R-1)}R_c S+\frac{\alpha_0}{(R-1)}S^{**}-\lambda RC$$

$$R_c\frac{\partial S}{\partial t}+TS=\alpha_0 C_c R_c C-\frac{\alpha_0}{(R-1)}R_c S-\varepsilon S-\lambda R_c S$$

$$\frac{\partial S^{**}}{\partial t}=\varepsilon S-\frac{\alpha_0}{(R-1)}S^{**}-\lambda S^{**}$$

where $T\equiv\frac{\partial}{\partial\tau}-f\frac{\partial^2}{\partial\tau^2}$ is the transport operator. Note that all rates in this equation have been normalized by the travel time τ_L as have all times, but the same symbols are used for clarity. In addition, C_c and α_0 are dimensionless as discussed above.

With the dimensional α_0 of approximately 50 m^3 / kg yr (Cvetkovic et al. 2004), the dimensionless version becomes 8.47 10^4 using a travel time of 100 years. Similarly, the dimensionless colloid concentration is in the range of 5.9 10^{-6} to 5.9 10^{-4}.

Entries into Scientific Notebook No. 318E for the period 6.01.05 to 6.06.05 were made by Scott Painter. No original text entered into this scientific notebook has been removed.

Scott Painter
1-18-10

SP 1/18/2010

SCIENTIFIC NOTEBOOK

by

Scott Painter

Southwest Research Institute
Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

December 21, 2005

Table of Contents

INITIAL ENTRIES....3
Radionuclide Transport Analysis for Yucca Mountain....4
Objective:....4
12.21.05 Effect of kinetic limitations on colloid-facilitated radionuclide transport at the field scale....5
12.21.2005 Calculations for previous entry....14
4.25.2006 Irreversible sorption to colloids: abstraction, scoping calculations, and parameter distributions....14

INITIAL ENTRIES

Scientific notebook: #318E Vol. 12
Issued to: S. L. Painter
Original Issue Date: 22-March-1999

By agreement with CNWRA QA this notebook will be printed at approximate quarterly intervals. This computerized electronic notebook is intended to address the criteria of CNWRA QAP-001.

Radionuclide Transport Analysis for Yucca Mountain

Account number: 20-06002.01.241 and 242
Description: Radionuclide Transport Analysis.
Collaborators: Drs. D. Pickett, and V. Cvetkovic (consultant)

Objective:
Development and testing of transport analysis methodology for spatially variable systems including demonstrations and applications to the proposed repository at Yucca Mountain.

SP 1/18/2010

SN 318E Vol. 12 Pg. 4, Scott Painter

12.21.05 Effect of kinetic limitations on colloid-facilitated radionuclide transport at the field scale

The following was prepared for submission to the international high level radioactive waste management conference, 2006.

I. INTRODUCTION

Radionuclides that interact strongly with minerals in the subsurface also tend to have a strong affinity for naturally occurring inorganic colloidal particles that are ubiquitous in the subsurface. Once attached to colloids, radionuclides have reduced interaction with the subsurface formations, potentially enhancing the transport of strongly sorbing radionuclides that would otherwise be relatively immobile. This process of colloid-facilitated transport has long been identified as a possible factor affecting the performance of high-level waste repositories and radionuclide migration in general. Indeed, colloid-facilitated transport of radionuclides has been clearly demonstrated in column experiments (e.g. [1–4]). Attachment to naturally occurring inorganic colloids is also suspected to be the transport mechanism responsible for observed long-distance migration of plutonium and other radionuclides at the Nevada Test Site [5] and other sites [6,7].

Despite clear indications of colloid-facilitated transport in the field and the laboratory, the overall impact of colloid-facilitated transport on the performance of potential repositories for radioactive waste remains uncertain. If sorption on colloids is reversible and rapid relative to the transport time scale, then the effect of colloid-facilitated transport is likely to be minor, given the relatively low colloid concentrations in most aquifers. Thus, the observation of enhanced transport in laboratory experiments with short transport times does not necessarily imply enhanced transport at the much longer time scales relevant for waste repository performance. In studies that indicate possible field-scale colloid-facilitated transport of radionuclides, the migrating radionuclides were detected in low concentrations.

Further complicating and adding uncertainty to analyses of potential repository impacts is the fact that irreversible or slowly reversible sorption is difficult to verify and quantify in the laboratory because of the long observation times required. Uncertainty in data interpretation was explored previously by Painter et al. [8]. They showed that data [9] on plutonium sorption on inorganic colloids could be interpreted with single-rate or two-rate models with a range of kinetic rate constants.

Most previous modeling of colloid-facilitated radionuclide transport focused on the relatively short time scales of laboratory experiments, where kinetic limitations on the attachment and detachment of colloids are clearly important. The relatively few studies that have addressed colloid-facilitated transport on the field scale have assumed equilibrium partitioning among solution, colloid-bound, and sorbed states (e.g. [10,11]). This study addresses the effect of kinetically limited sorption/desorption on field-scale transport of radionuclides. The analysis is intended to identify the regions of the parameter space where kinetic limitations are important. The focus is on natural colloids that are typically observed in low concentrations in groundwater and may be modeled as constant in space and time. Anthropogenic colloids, which may have

highly transient populations as well as different behavior in the subsurface, are not addressed here.

II. TRANSPORT MODEL

The transport model is an extension to the one presented in Cvetkovic et al. [12]. The basic configuration is a single transport pathway connecting a radionuclide source location to a monitoring boundary. The movement of colloids and dissolved radionuclide is described with the advection-dispersion equation with specified retardation constants. Colloids are considered to be subject to both irreversible removal with a specified removal rate constant and reversible attachment and detachment. Radionuclides are subject to fast equilibrium sorption on the porous matrix and slow kinetically controlled sorption on colloids. The quantity of interest in this work is the transient breakthrough at the monitoring boundary due to a step-function input at the source location. A similar transport scenario was considered by Cvetkovic et al. [12], but transients were neglected in that work.

II.A. Conservation Equations

The conservation equations for radionuclides are written as

$$\frac{\partial C}{\partial t} + \mathcal{T}C = -\psi^{C\to S}(C,S) - \psi^{C\to S^*}(C,S^*) - \psi^{C\to S^{**}}(C,S^{**}) - \psi^{C\to C^*}(C,C^*) - \lambda C \tag{1a}$$

$$\frac{\partial S}{\partial t} + \mathcal{T}S = \psi^{C\to S}(C,S) - \psi^{S\to S^*}(S,S^*) - \psi^{S\to S^{**}}(S,S^{**}) - \lambda S \tag{1b}$$

$$\frac{\partial C^*}{\partial t} = \psi^{C\to C^*}(C,C^*) - \lambda C^* \tag{1c}$$

$$\frac{\partial S^*}{\partial t} = \psi^{S\to S^*}(S,S^*) + \psi^{C\to S^*}(C,S^*) - \lambda S^* \tag{1d}$$

$$\frac{\partial S^{**}}{\partial t} = \psi^{S\to S^{**}}(S,S^{**}) + \psi^{C\to S^{**}}(C,S^{**}) - \lambda S^{**} \tag{1e}$$

where C is the concentration of radionuclides in solution, C^* is the concentration of immobile radionuclides, S is the concentration of radionuclides attached to mobile colloids, S^* is the concentration of radionuclides attached to temporarily immobilized colloids, and S^{**} is the concentration of radionuclides attached to permanently immobilized colloids. All concentrations are defined on a bulk aquifer volume basis. The ψ terms represent the various exchanges. The differential operator $\mathcal{T}$ is the transport operator representing advection and dispersion processes.

Taking into account that radionuclides and temporarily immobilized colloids attach and detach quickly from the porous matrix and can be modeled with an equilibrium model parameterized with partitioning coefficients, the conservation equations for radionuclides can be written as

$$R\frac{\partial C}{\partial t} + \mathcal{T}C = -\psi^{C\to S}(C,S) - \psi^{C\to S^*}(C,K_C S) - \psi^{C\to S^{**}}(C,S^{**}) - \lambda RC \tag{2a}$$

$$R_C \frac{\partial S}{\partial t} + \mathcal{T} S = \psi^{C \to S}(C,S) + \psi^{C \to S^*}(C, K_C S) - \psi^{S \to S^{**}}(S, S^{**}) - \lambda R_C S \quad (2b)$$

$$\frac{\partial S^{**}}{\partial t} = \psi^{S \to S^{**}}(S, S^{**}) + \psi^{C \to S^{**}}(C, S^{**}) - \lambda S^{**} \quad (2c)$$

where R_c is the colloid retardation factor, and R is the radionuclide retardation factor.

II.B. Transport Operator in Lagrangian Form

On a one-dimensional streamtube, $\mathcal{T} = v \frac{\partial}{\partial x} - D \frac{\partial^2}{\partial x^2}$ (3)

where $D = a|v|$ is the longitudinal dispersion coefficient, v is velocity, a is the dispersivity, and x is position along the streamtube. Writing the dispersivity as a fixed fraction f of the travel distance L,

$$\mathcal{T} = v \frac{\partial}{\partial x} - f\, L\, |v| \frac{\partial^2}{\partial x^2}. \quad (4)$$

Note that both v and L may be uncertain. For the potential repository at Yucca Mountain, for example, uncertainty in L comes from uncertainty in the tuff-alluvium contact. Uncertainty in v comes from uncertainty in porosity and possibly total flow rate in the streamtube. These two uncertain parameters can be collapsed to a single uncertain parameter by re-writing the transport operator in a Lagrangian form. In Lagrangian form,

$$\mathcal{T} = \frac{\partial}{\partial \tau} - f\, \tau_L \frac{\partial^2}{\partial \tau^2} \quad (5)$$

where τ is the groundwater travel time, and τ_L is the global travel time. With this form of the transport operator, the concentrations are functions of τ and time t, and uncertainties in travel distance and velocity are manifest through the uncertainty in the total travel time τ_L.

II.C. Exchange Model

The various exchange terms above can be linear or non-linear, equilibrium or kinetically controlled. Linear models are generally applicable when the sorption capacity is large relative to the local aqueous concentration. Cvetkovic et al. [12] evaluated linear and nonlinear models for sorption on colloids, and concluded that conditions expected in the Yucca Mountain saturated zone are well within the range of validity for linear models. Assuming linear exchanges,

$$\psi^{C \to S}(C,S) = \alpha_f C - \alpha_r S \quad (6)$$

$$\psi^{C \to S^*}(C,S^*) = \alpha_f^* C - \alpha_r^* S^* \quad (7)$$

$$\psi^{C \to S^{**}}(C,S^{**}) = -\alpha_r^{**} S^{**} \quad (8)$$

$$\psi^{S \to S^{**}}(S,S^{**}) = \varepsilon S \quad (9)$$

All of these exchanges are bidirectional except for the last two, which are unidirectional (irreversible). The S to S^{**} exchange is unidirectional by definition because it applies to permanent removal. No forward rate is included in the C to S^{**} exchange because a permanently removed colloid is indistinguishable from the porous matrix, and that exchange is presumably already counted in the any measured rates for C to C^{*} exchange.

Six rate coefficients appear in the above set of exchange equations. However, these rate coefficients are not independent, and two simplifying assumptions can reduce the number of parameters. First, the rate coefficients for desorption from colloids are assumed to be identical irrespective of whether the colloid is mobile or immobile: $\alpha_r = \alpha_r^* = \alpha_r^{**}$. Second, the forward rate for sorption on colloids is generally assumed to be proportional to the concentration of sorption sites, and thus proportional to colloid concentration. This assumption implies $\alpha_f = \alpha_0 C_c$ and $\alpha_f^* = K_C \alpha_f$ where C_c is colloid concentration, and α_0 is an intrinsic forward rate.

Because natural colloids typically reflect the mineralogy of the aquifer, at least in a gross sense, radionuclides that sorb strongly on the porous matrix also tend to have a strong affinity for colloids. The ratio of forward to reverse rate coefficients for sorption on colloids should be related to the distribution coefficient K_d to reflect this. Using an equilibrium sorption model, Contardi et al. [11] define a dimensionless distribution coefficient for solutes on colloids as $F\,C_c\,K_d$, where F is the ratio of colloid specific surface area to porous matrix specific surface area. Adopting similar reasoning, we write

$$\frac{\alpha_f}{\alpha_r} = F C_c K_d = C_c' K_d' \qquad (10)$$

where

$$C_c' \equiv \frac{F C_c \theta}{\rho(1-\theta)} \qquad (11)$$

is a dimensionless colloid concentration and K_d' is a dimensionless distribution coefficient.

II.D. Transport Equations in Dimensionless Form

Using the above exchange model, normalizing times and rates by the travel time τ_L, and normalizing concentrations to the input value, the transport equations can be put in a convenient dimensionless form. For simplicity, we retain the same symbols for the normalized concentrations

$$R\frac{\partial C}{\partial t} + \tau C = -\alpha R_C C + \frac{\alpha}{C_c' K_d'} R_C S + \frac{\alpha}{C_c' K_d'} S^{**} - \lambda' RC \qquad (12a)$$

$$R_C \frac{\partial S}{\partial t} + \mathcal{T} S = \alpha R_C C - \frac{\alpha}{C_c' K_d'} R_C S - \varepsilon' S - \lambda' R_C S \qquad (12b)$$

$$\frac{\partial S^{**}}{\partial t} = \varepsilon' S - \frac{\alpha}{C_c' K_d'} S^{**} - \lambda' S^{**} \qquad (12c)$$

where $\alpha \equiv \alpha_f \tau_L$, $\lambda' \equiv \lambda \tau_L$, and $\varepsilon' \equiv \varepsilon \tau_L$ are normalized rate constants, and the transport operator now becomes

$$\mathcal{T} = \frac{\partial}{\partial \tau} - f \frac{\partial^2}{\partial \tau^2}. \qquad (12d)$$

III. GENERIC ANALYSIS OF EFFECT OF KINETIC LIMITATIONS

The sensitivity to the dimensionless rate constant α is examined in Fig. 1. For this sensitivity set of calculations, the other parameters are fixed at the values given in Table 1.

TABLE I. Parameter values for the generic analysis.

Parameter	Description	Value
R	Retardation factor	2000
R_c	Colloid retardation factor	20
λ'	Decay constant normalized by travel time	$2.9\ 10^{-3}$
C_c	Dimensionless colloid concentration	$2.0\ 10^{-4}$
ε'	Removal rate constant normalized by travel time	0

Figure 1 shows a set of breakthrough curves for different values of the dimensionless rate constant α: 0, 0.002, 0.02, 0.2, 2 and 20. The situation $\alpha = 0$ is equivalent to the colloid-free case because radionuclides are not transferred to colloids in this situation.

Figure 1 indicates that colloids affect the breakthrough curve in two ways. First, colloids increase the total "throughput" (value after steady-state is reached). For the combination of parameters in Figure 1 (see Table 1), this enhancement is modest – roughly a factor of three. Second, colloids greatly decrease the time of early arrival if the kinetic limitations are important. In Figure 1, for example, colloids decrease the time at which the breakthrough curve reaches 1% of its maximum value by more than a factor of 10.

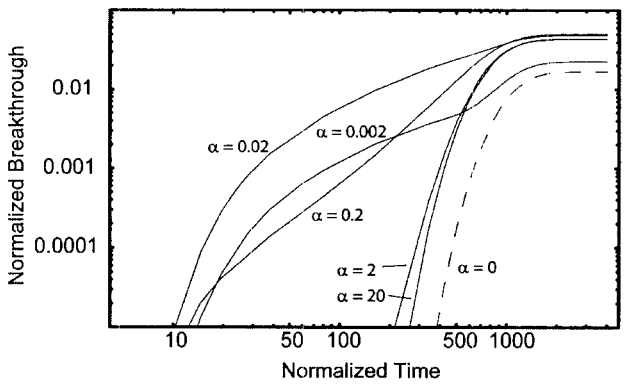


Fig. 1. Normalized breakthrough curves for various values of the normalized rate constant α.

The effects of kinetic limitations on sorption/desorption can be understood by comparing the six normalized breakthrough curves in Figure 1. If the detachment rate is fast compared with time spent by the colloid in the transport path then the radionuclide may sorb and desorb multiple times while traveling through the transport path. In this case, transport may be approximated using the equilibrium model. The situation $\alpha = 2$ and $\alpha = 20$ closely approximate the equilibrium situation. At the other extreme, if α is very small then radionuclides traveling at the retarded velocity do not have enough time to transfer to colloids and colloids have little or no effect (i.e. $\alpha = 0$ curve in Figure 1). The situation of $\alpha = 0.2$, 0.02 and 0.002 are intermediate between these two extremes and result in the greatest amount of transport.

Exploring this further, we define a dimensionless Damkohler number as the product of the detachment rate times the average time spent by a colloid in the transport path,

$$Da = \frac{\alpha R_c}{C_c' K_d'}. \qquad (13)$$

Kinetic effects become important if the Damkohler number is less than a critical value. In Figure 1, Damkohler number for the five curves with colloids ranges from 0.1 to 1000. Kinetic effects are not important for $Da = 100$, but are important for $Da = 10$ ($\alpha = 0.2$). From this and similar families of breakthrough curves (not shown), it appears that critical Damkohler number is approximately 10. Thus, kinetic limitations are important and can enhance transport if $Da = \frac{\alpha R_c}{C_c' K_d'} < Da^{crit} \approx 10$.

The critical regime for colloid-facilitated transport is

$$\frac{1}{R} < \alpha < 10 \frac{C_c' K_d'}{R_c}. \qquad (14)$$

If the dimensionless rate is greater than this range, then colloids can still have an effect on transport. However, the system is in the equilibrium range in this situation, and for typical colloid concentrations, the colloid-facilitated enhancement is modest. If the dimensionless rate constant is smaller than the range Eq. 14, then radionuclides do not have time to transfer to colloids as

they travel through the system.

IV. POTENTIAL EFFECTS OF KINETICS ON COLLOID FACILITATED TRANSPORT IN THE ALLUVIAL AQUIFER NEAR YUCCA MOUNTAIN, NEVADA

The transport of plutonium in the alluvial aquifer near Yucca Mountain, Nevada is used in this section to illustrate the potential field-scale effects of kinetically controlled attachment/detachment to colloids. In this sensitivity analysis, the alluvial aquifer is isolated from the rest of the system. The aquifer is initially free of radionuclide mass. At $t = 0$, a constant input of mass is introduced into the system. The intent is to investigate the potential sensitivities by comparing breakthrough curves for different assumptions about kinetically controlled attachment to colloids.

Painter et al. [8] analyzed plutonium sorption experiments of Lu et al. [9] and identified sorption kinetics governed by two rates. For the slow sorption, Painter et al. [8] estimated values for an intrinsic rate constant $\alpha_0 = \alpha_f / C_c$ in the range 10 – 100 m^3/kg-yr.

Transport parameters for the Yucca Mountain alluvium are uncertain, and probabilistic modeling is often used in performance assessment studies to quantify the effect of this uncertainty on transport (e.g. [14]). Taking median values from these studies as typical values for the parameters (R_c=34; C_c=10^{-4} kg/m^3; R=3000; τ= 100 years), the dimensionless Damkohler number is estimated to be 190 – 1900. This range is greater than the critical value of 10, indicating the equilibrium approximation is adequate. However, if values closer to the tails of the distribution are used instead of median values, then the conclusion is different. Using a travel time of 10 years and a colloid retardation factor of 10, both of which are near the 20th percentile in their respective distributions, the Damkohler number is estimated to be 5 – 50, indicating possible effects of kinetic limitations.

This scoping analysis suggests that part of the distribution is in the kinetic range while the bulk of the distribution is in the equilibrium range. To further assess the potential effects of kinetically controlled attachment/detachment to colloids, the kinetic process model was run in a Monte Carlo mode with all uncertain parameters sampled from probability distributions. The assumed probability distributions for C_c, R, R_c, and τ are shown in Figures 2 – 4. The R and τ distributions are the same as those in [15], and are consistent with those from [14]. The C_c distributions are those of [11]. The R_c distributions are those of [11] after removing an assumed truncation at R_c=8.

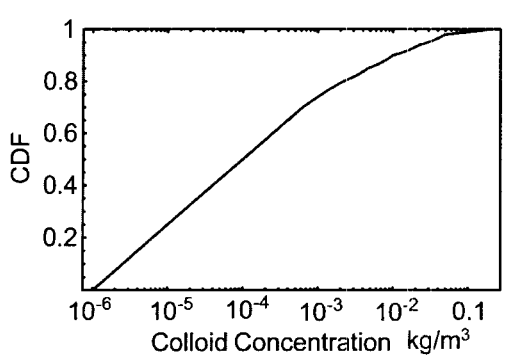


Fig. 2. Cumulative distribution function (CDF) for the uncertain parameter C_c, colloid

concentration.

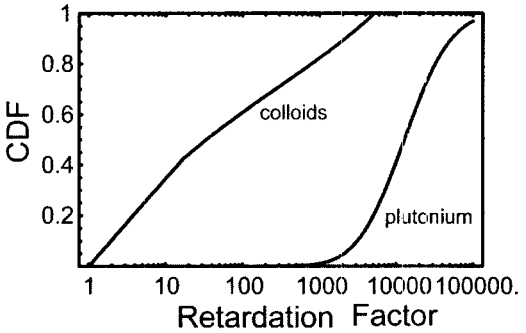


Fig. 3. Cumulative distribution function (CDF) for the uncertain parameters R_c and R, retardation factors for colloids and plutonium.

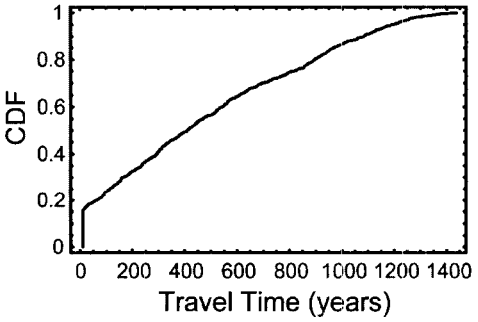


Fig. 4. Cumulative distribution function (CDF) for the uncertain parameter τ_L, groundwater travel time.

Breakthrough curves after averaging over 500 realizations are shown in Figure 5. The four expected breakthrough curves represent the no-colloids case, the case with equilibrium partitioning (the usual performance assessment model), and two kinetic models with (α_0=10 and 100 m^3/kg-yr). In the equilibrium partitioning case (fast and reversible sorption to colloids), the enhancement represents at most a factor of 3 shift to earlier times. The enhancement is much larger for the two kinetic models. It is emphasized that the kinetic rates used in constructing Fig. 5 are consistent with the previous analysis [8] of the laboratory data of Lu et al. [9].

Although kinetic limitations greatly enhance the early breakthrough in Fig. 5, the amount of mass in the leading tail of the breakthrough curve represents a small fraction of the total. The bulk of the breakthrough curve in Fig. 5 is not enhanced significantly by colloids. However, the breakthrough curves in Figure 5 are based on a constant input of radionuclides into the streamtube. More realistic scenarios involving a highly transient source would likely show significant enhancement of the peak dose due to kinetic limitations.

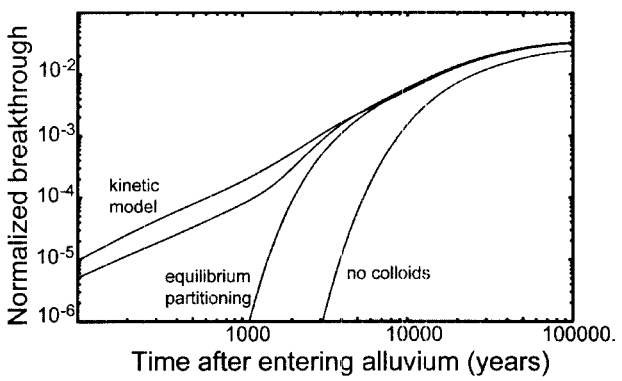


Fig. 5. Breakthrough curves for various assumptions about radionuclide attachment to groundwater colloids in the Yucca Mountain alluvial aquifer. The curves are expected values based on 500 realizations.

V. CONCLUSIONS

Kinetic effects are typically neglected in studies of colloid-facilitated radionuclide transport at the field scale because of the long transport times involved. Equation 14, which was deduced from detailed process-level simulations, provides a simple screening criterion for evaluating the potential effect of kinetic limitations. If the normalized rate constant α is in the critical regime defined by Equation 14, kinetic limitations on sorption/desorption may affect radionuclide migration. The main effect in this situation is to enhance transport for a small fraction of the radionuclide mass, leading to early breakthrough.

Parameters governing transport in the alluvial aquifer near Yucca Mountain, Nevada are uncertain. Applying Eq. 14 with representative (median) values for the key uncertain parameters suggests that equilibrium partitioning assumption is adequate for modeling colloid-facilitated radionuclide transport. However, a small portion of the parameter space overlaps with the kinetic region. Monte Carlo simulations suggest this small kinetically controlled region of the parameter space may have the potential to cause early breakthrough for a small fraction of migrating plutonium. Further analyses using more realistic models for plutonium release into the alluvial aquifer are needed to understand the risk significance of this potential early breakthrough.

ACKNOWLEDGMENTS

The authors are grateful to Alexander Sun for reviewing this manuscript and to David Pickett for useful discussions on colloid transport.

This paper was prepared to document work performed by the Center for Nuclear Waste Regulatory Analyses (CNWRA) and its contractors for the U.S. Nuclear Regulatory Commission (NRC) under Contract No. NRC–02–02–012. The activities reported here were performed on behalf of the NRC Office of Nuclear Material Safety and Safeguards, Division of High-Level Waste Repository Safety. This abstract is an independent product of the CNWRA and does not necessarily reflect the view or regulatory position of NRC.

REFERENCES

1. J. SAIERS, and G.M. HORNBERGER, "Water Resources Research 32", 33–41 (1996).
2. H. VAN DE WEERD, and A. LEIJNSE, "Journal of Contaminant Hydrology 26", 245–256 (1997).
3. A. NOELL, J. THOMPSON, M. CORAPCIOGLU and I. TRIAY, "Journal of Contaminant Hydrology 31", 23–56 (1998).
4. L. LUHRMANN and U. NOSECK, "Water Resources Research 34", 421–426 (1998).

5. A. KERSTING, D. EFURED, D. FINNEGAN, D. ROKOP, D. SMITH, and J. THOMPSON, "Nature 397", 56–59 (1999).
6. J. MCCARTHY, and J. ZACHARA, "Environmental Science and Technology 23", 496–502 (1989).
7. W. PENROSE, W. PLOZER, E. ESSINGTON. D. NELSON, and K. ORLANDINI, "Environmental Science and Technology 36", 5369–5375 (1990).
8. S. PAINTER, V. CVETKOVIC, D. TURNER, and D. PICKETT, "Environmental Science and Technology 36", 5369–5375 (2002).
9. N. LU, J. CONCA, G. PARKER, P. LEONARD, B. MOORE, B. STREITELMEIER, and I. TRIAY, "Adsorption of actinides onto colloids as a function of time, temperature, ionic strength, and colloid concentration", LA-UR-00-5121, Los Alamos Natl. Lab. (2000).
10. D. A. PICKETT, and W. L. DAM, "Nuclear Science and Engineering 151", 1–7 (2005).
11. BECHTEL SAIC COMPANY LLC, "Saturated Zone Colloid Transport", ANL-NBS-HS-000031 Rev. 01, Bechtel SAIC Company (2003).
12. V. CVETKOVIC, S. PAINTER, D. TURNER, D. PICKETT, and P. BERTETTI, Water Resources Research, 40, doi:10.1029/2004WR003048 (2004).
13. J. CONTARDI, D. TURNER, and T. AHN, "Journal of Contaminant Hydrology 47", 323–333 (2001).
14. S. MOHANTY, T. J. MCCARTIN, and D. W. ESH, "Total-system performance assessment (TPA) Version 4.0 Code: Module Descriptions and User's Guide", Center for Nuclear Waste Regulatory Analyses (2002).
15. S. PAINTER and V. CVETKOVIC, "Colloid-facilitated Transport in the Saturated Zone: Process-level Sensitivity Analysis", Center for Nuclear Waste Regulatory Analyses (2004).

12.21.2005 Calculations for previous entry

Calculations for the previous entry are included in the attached Mathematica notebooks:

Show Distributions.nb Figures 2—4.

CFT in Alluvium 5.nb Figure 1

CFT in Alluvium Sampled 7.nb Figure 5

4.25.2006 Irreversible sorption to colloids: abstraction, scoping calculations, and parameter distributions

This entry describes a new abstraction for radionuclide release from waste packages due to irreversible attachment to colloids. The abstraction is for irreversible sorption of radionuclides to corrosion product colloids. Other colloids are not considered in this abstraction. Note that this abstraction is documented here, but is work performed for 20.06002.01.222 and 20.06002.01.354.

This abstraction was implemented in TPA5.0.2 and was previously documented in SCR611. This

entry also covers scoping calculations and basis for parameter distributions.

Overview of the abstraction:

Consider iron-oxide type corrosion products as dominant in waste package. Focus on release abstraction – transport model remains unchanged.

Physical picture is fast irreversible sorption with competition until all sorption sites on colloids are filled.

Pu, Am, Th, and Cm compete for available sorption sites on colloids. U also competes, but is not specifically tracked as a colloid.

Assign radionuclide mass to colloids until *finite sorption capacity* is reached.

Remainder of radionuclide mass is dissolved (limited by solubility or dissolution rate).

Abstraction is conservative. Additional competition with stationary corrosion products may reduce releases, but this will require more detailed model and technical basis is more difficult to develop.

Dataflow:

(1) Before calling RELEASET, sample sorption capacity (colloid concentration * specific surface area * number sites per unit area). This is a new parameter S_X.

(2) For each of the five elements, sample *relative affinity* γ for corrosion product colloid.

(3) Sample solubility limit C_S for each radioelement.

(4) Calculate an *effective solubility limit* by assuming that Pu, Am, U are solubility limited and that sorption is described by a competitive Langmuir-like sorption model. Effective solubility limit is then

$$C_{Seff,Pu} = C_{S,Pu} + S_x \frac{\gamma_{Pu} C_{S,Pu}}{\gamma_{Pu} C_{S,Pu} + \gamma_{Am} C_{S,Am} + \gamma_U C_{S,U}}$$

and similar for Am. Solubility for Th and Cm are unchanged.

(5) Call RELEASET with the effective solubility limits.

(6) Using concentrations C_j as calculated by RELEASET, set the "J" species concentrations:

$C_{JPu} = S_x \frac{\gamma_{Pu} C_{Pu}}{\sum_j \gamma_j C_j}$ where sum is over Pu, Am, Th, Cm, and U.

(7) Dissolved concentration in water leaving waste package is then $C_{Pu} - C_{JP}$.

Final dissolved concentrations may be slightly above true solubility limit, but this is relatively unimportant.

At late times, when releases from waste package are limited by dissolution rate, colloids may take all released Pu.

Scoping calculations

A set of scoping calculations were carried out to assess the possible effects of this abstraction. Note that the previous abstraction for release due to irreversible sorption to colloids used a constant enhancement factor on the solubility limit. This was used as a baseline for comparing the new abstraction.

Calculations are Monte Carlo with uncertainty distributions for all parameters. See attached mathematica notebook "Scoping Calculations for Irreversible Sorption to Colloids.nb" for the calculations.

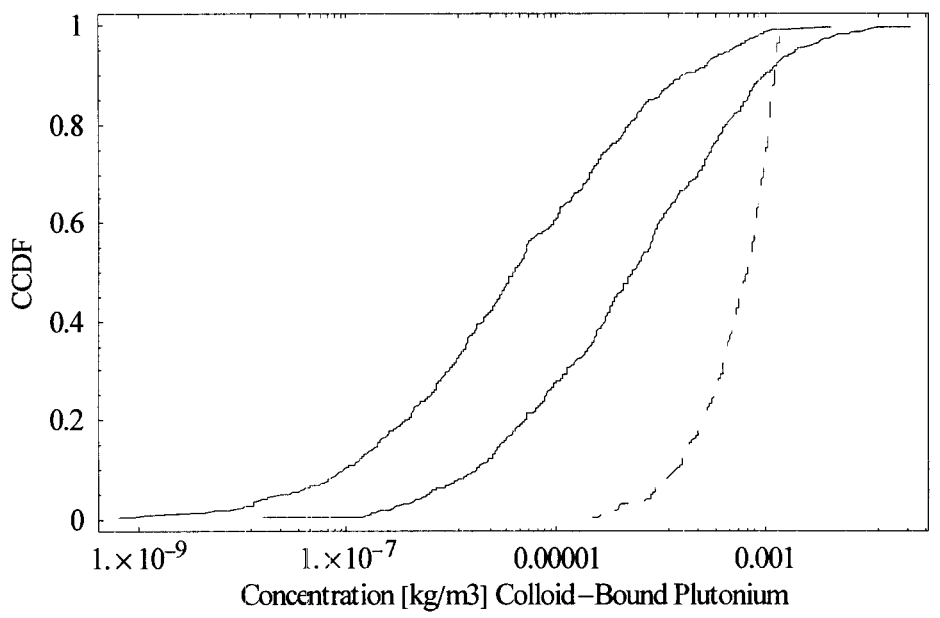
Parameter distributions and source are summarized in the following table.

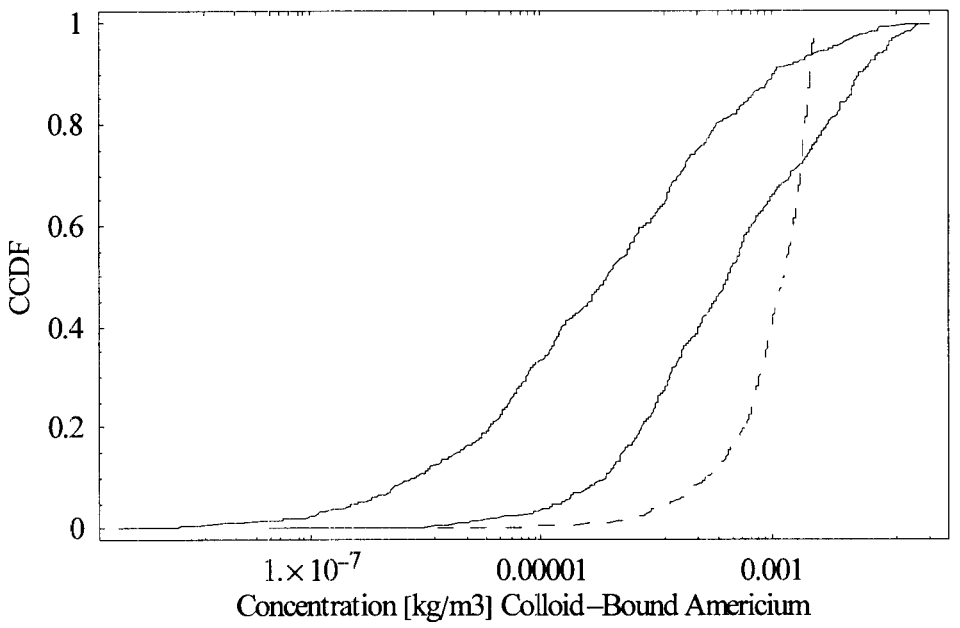
Parameter	Distribution	Source
Solubility Limited Concentration Pu [kg/m^3]	Uniform: 2.4 10^{-6} to 2.4 10^{-4}	Current TPA distribution
Solubility Limited Concentration Am [kg/m^3]	Uniform: 2.4 10^{-8} to 2.4 10^{-4}	Current TPA distribution
Solubility Limited Concentration U [kg/m^3]	Log Triangular: 2.4 10^{-6} to 2.4 mode = 7.6 10^{-3}	Current TPA distribution
Kd for Pu on Hematite Colloids [m^3/kg]	Piecewise Empirical: 0 10 0.15 50 0.35 100 0.85 500 1.0 1000	DOE distributions for Kd of Pu on hematite colloids
Kd for Am on	Piecewise Empirical:	DOE distributions for Kd of Am on

Hematite Colloids [m^3/kg]	0 100 0.15 500 0.35 1000 0.90 5000 1.0 10000	hematite colloids
Kd for U on Hematite Colloids [m^3/kg]	Log Uniform 0.1 to 50.	DOE distributions for Kd of U on stationary corrosion products after adjusting for increased surface area of colloids
Sorption Capacity [moles/m^3]	Calculated.	Calculated for this scoping exercise based on DOE values for specific surface area and colloid concentration Also assumes 2.3 sites/nm^2 . See next subsection.

The Kd for Am, U, Pu are used for the relative affinity in this scoping calculations. Affinities calculated/assumed differently in the TPA implementation. See following subsection for details on parameters.

Results are shown in the following two figures. Each figure is the CDF of J-species released from waste package. J-species are radionuclides irreversibly attached to colloids. For each figure, three simulations were performed with 500 realizations each. The solid curve on the left is with U allowed to compete for sites, the solid curve on the right is without U. The dashed curve in each is the current abstraction. Note that by enforcing a limit on sorption capacity, the proposed abstraction reduces the median release by a large amount. Releases at upper end of distribution are reduced by a smaller amount and may actually increase at extreme end.



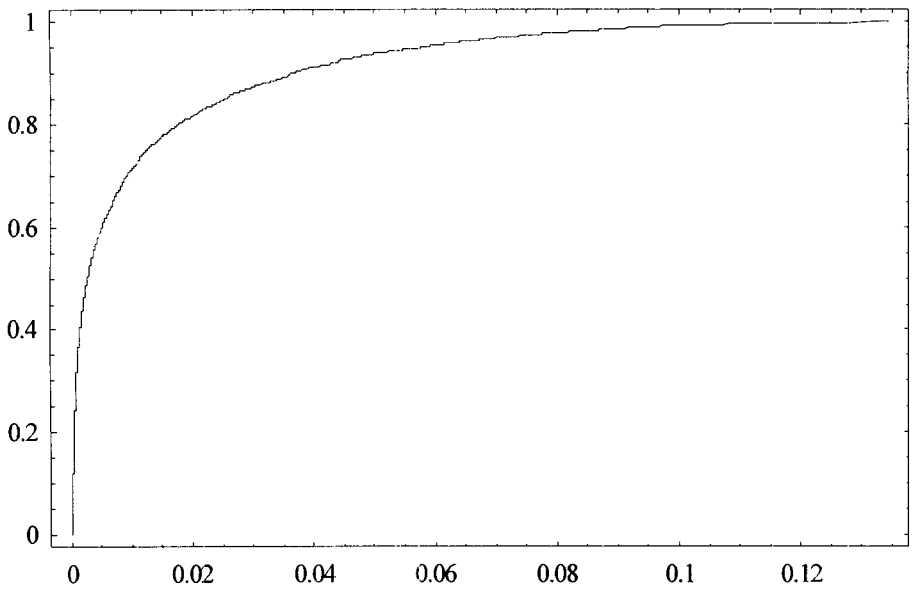


Parameter distributions.

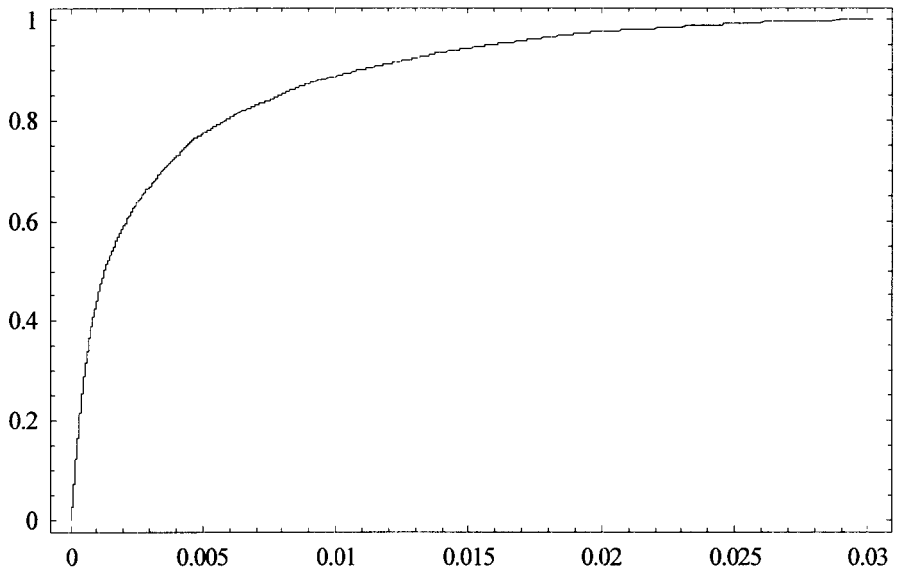
Main parameter is colloid sorption capacity [moles/m3]. It is based on colloid concentration and specific surface area for colloids.

Distribution of colloid sorption capacity is calculated in the attached mathematica notebook Colloid Sorption Capacity.nb . Two cases are considered. In the first case, the full range of specific surface area as given by DOE in EBS Radionuclide Transport Abstraction ANL-WIS-PA-000001 REV 01 and Waste Form and Indrift Colloids-Associated Radionuclide Concentrations: Abstraction and Summary MDL-EBS-PA-000004 REV 00 is used. In the second, the specific surface area is truncated at 170 m2/g, which is at the upper end for goethite, as documented in the aforementioned documents. In both cases, colloid concentrations span a range from 0.05 to 50 mg/liter as given by DOE in the aforementioned documents. Results are shown in the next two figures. First is the full distribution, second is the truncated distribution.

Relative affinity factors have relatively little information. Within the irreversible sorption model, the affinity factors are ratios of rates constant for sorption on colloids. These are defined relative to one radionuclide, chosen arbitrarily. Plutonium is used to define the relative affinities. Thus, the relative affinity for americium is the forward rate constant for americium sorption on colloids divided by the forward rate constant for plutonium sorption on colloids. Note that the abstraction is much more sensitive to the sorption capacity as compared with the relative affinities.



Plot is CDF for colloid sorption capacity [moles/m3]. This is the untruncated case.



Plot is CDF for colloid sorption capacity [moles/m3]. This is the truncated case.

Entries into Scientific Notebook No. 318E for the period 12.21.05 to 4.25.06 were made by Scott Painter. No original text entered into this scientific notebook has been removed.

Scott Painter
1-18-10

RCS 1/18/2010

Colloid Facilitated Transport in Alluvium: Transient calculations for IHLRWMC 2006

This notebook includes the calculations for Figure 1 of the International High Level Radioactive Waste Management Conference Paper.
A few other sensitivity calculations are also included here.
Scott Painter
12.21.2005

Preliminaries

```
<< Statistics`DescriptiveStatistics`
<< Graphics`Graphics`
<< Statistics`ContinuousDistributions`
$HistoryLength = 5
```

```
5
```

Define the breakthrough solution

rates are in units of 1/year

```
SolveIt[alphaf_, R_, Rc_,
  eps_, a_, lambda_, tauL_] := Module[
  {alphar = alphaf / a,
Deff = 0.1 * tauL,
   a11, a12, a13, a21, a22, a23, a31, a32, a33,
   solution, X, Y},
  a11 = -(alphaf Rc + lambda R);
a12 = Rc alphar;
a13 = alphar;
a21 = Rc alphaf;
```

```
a22 = - (Rc alphar + eps + lambda Rc);
a23 = 0.;
a31 = 0.;
a32 = eps;
a33 = - (lambda + alphar);
solution = NDSolve[
    {R ∂t u[t, x] == -∂x u[t, x] + Deff ∂x,x u[t, x] +
       a11 u[t, x] + a12 v[t, x] + a13 w[t, x],
     Rc ∂t v[t, x] == -∂x v[t, x] + Deff ∂x,x v[t, x] +
       a22 v[t, x] + a21 u[t, x],
     ∂t w[t, x] == a32 v[t, x] + a33 w[t, x],
     u[0, x] == 0.,
     u[t, 0] == Min[t / 2., 1],
     u[t, tauL * 3. ] == 0,
     v[0, x] == 0.,
     v[t, tauL * 3.] == 0.,
     v[t, 0] == 0.,
     w[0, x] == 0.},
    {u, v, w}, {t, 0, tauL * R * 2}, {x, 0, tauL * 3.},
    StartingStepSize → {tauL * R / 4000., tauL / 2000.},
    MaxStepSize → {tauL * R * 2 / 500., tauL / 500.},
    MaxSteps → {20000, 20000}];
{X, Y} = {u, v} /. First[solution] ;
  Clear[solution];
  {X, Y}
 ]
```

```
General::spell1 :
 Possible spelling error: new symbol name "alphar" is similar to existing symbol "alphaf". More…
```

Sensitivity to Forward Rate: Figure 1

```
Rc = 20;
eps = 0.0;
R = 2000.;
lambda = 2.88 10^-5 * 100.;
cc = 2 10^-4;
a = cc * (R - 1);
alpha0 = 10^5;
alphaf = alpha0 * cc;
```

```
tauL = 1.;
{X0, Y0} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

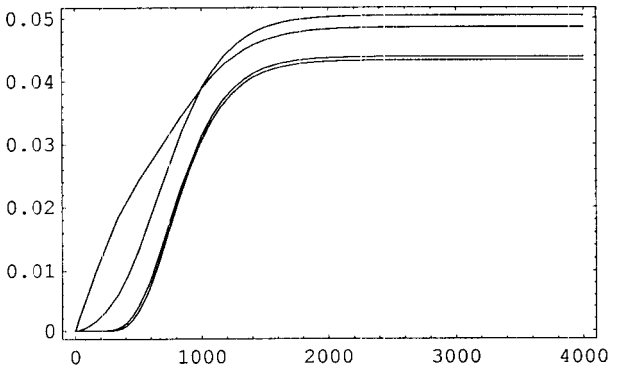
```
alpha0 = 10^4;
alphaf = alpha0 * cc;
tauL = 1.;
{X1, Y1} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alpha0 = 10^3;
alphaf = alpha0 * cc;
tauL = 1.;
{X2, Y2} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alpha0 = 10^2;
alphaf = alpha0 * cc;
tauL = 1.;
{X3, Y3} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alpha0 = 10^1;
alphaf = alpha0 * cc;
tauL = 1.;
{X4, Y4} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

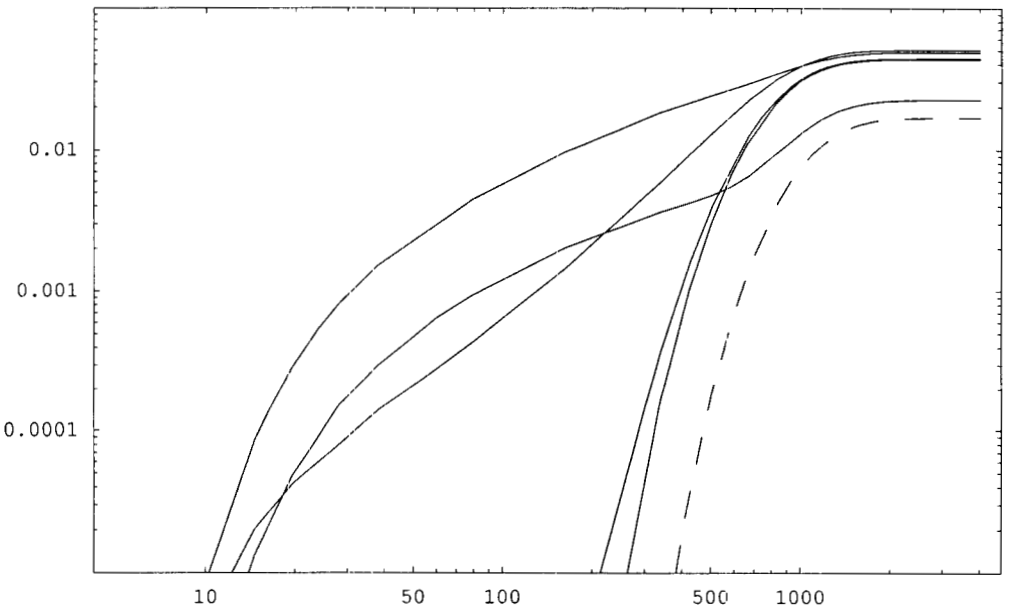
```
alphaf = 0.0;
tauL = 1.;
{X00, Y00} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];

DisplayTogether[

 Plot[X3[t, tauL] + Y3[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X2[t, tauL] + Y2[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X1[t, tauL] + Y1[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X0[t, tauL] + Y0[t, tauL] , {t, 0, tauL*R*2}] ,
 PlotRange → All, Frame → True, Axes → False]
```



- Graphics -

RCP 1/18/2010

```
DisplayTogether[
 LogLogPlot[ X00[t, tauL] + Y00[t, tauL] ,
  {t, 0, tauL*R*2}, PlotStyle → Dashing[{0.03, 0.03} ] ] ,
 LogLogPlot[ X4[t, tauL] + Y4[t, tauL] , {t, 0, tauL*R*2}] ,
 LogLogPlot[ X3[t, tauL] + Y3[t, tauL] , {t, 0, tauL*R*2}] ,
 LogLogPlot[ X2[t, tauL] + Y2[t, tauL] , {t, 0, tauL*R*2}] ,
 LogLogPlot[ X1[t, tauL] + Y1[t, tauL] , {t, 0, tauL*R*2}] ,
 LogLogPlot[ X0[t, tauL] + Y0[t, tauL] , {t, 0, tauL*R*2}] ,
 PlotRange → {-5, -1}, Frame → True, Axes → False]
```



- Graphics -

Sensitivity to Forward Rate: R=3000, Lower cc

This figure not shown in paper, but results were quoted.

```
Rc = 20;
eps = 0.0;
R = 3000.;
lambda = 2.88 10^-5 * 100.;
cc = 10^-4;
a = cc * (R - 1);
alpha0 = 10^5;
alphaf = alpha0 * cc;
```

```
tauL = 1.;
{X0, Y0} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alpha0 = 10^4;
alphaf = alpha0 * cc;
tauL = 1.;
{X1, Y1} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

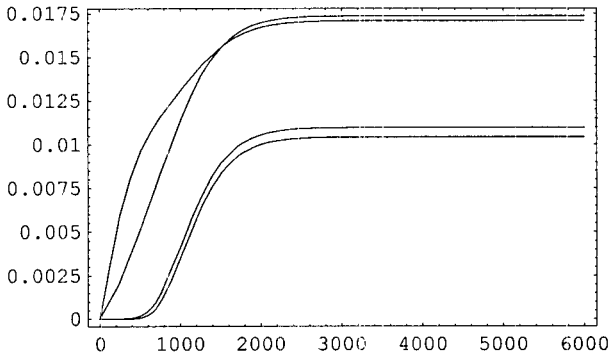
```
alpha0 = 10^3;
alphaf = alpha0 * cc;
tauL = 1.;
{X2, Y2} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alpha0 = 10^2;
alphaf = alpha0 * cc;
tauL = 1.;
{X3, Y3} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alpha0 = 10^1;
alphaf = alpha0 * cc;
tauL = 1.;
{X4, Y4} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alphaf = 0.0;
tauL = 1.;
{X00, Y00} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

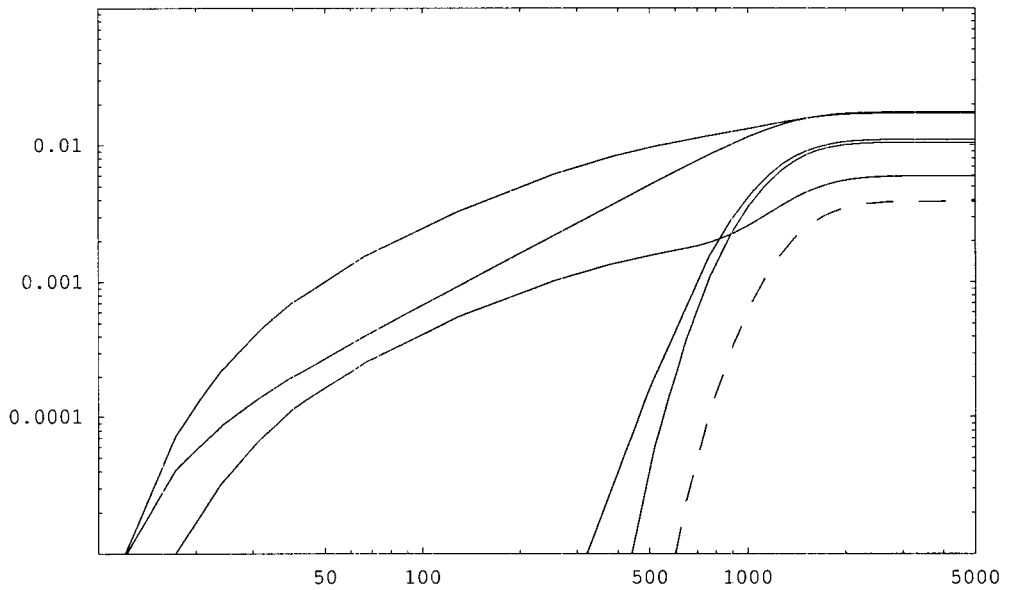
```
DisplayTogether[

 Plot[X3[t, tauL] + Y3[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X2[t, tauL] + Y2[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X1[t, tauL] + Y1[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X0[t, tauL] + Y0[t, tauL] , {t, 0, tauL*R*2}] ,
 PlotRange → All, Frame → True, Axes → False]
```



- Graphics -

RD 1/18/2010

```
DisplayTogether[
 LogLogPlot[ X00[t, tauL] + Y00[t, tauL] ,
  {t, 10, tauL*R*2}, PlotStyle → Dashing[{0.03, 0.03} ] ] ,
 LogLogPlot[ X4[t, tauL] + Y4[t, tauL] , {t, 10, tauL*R*2}] ,
 LogLogPlot[ X3[t, tauL] + Y3[t, tauL] , {t, 10, tauL*R*2}] ,
 LogLogPlot[ X2[t, tauL] + Y2[t, tauL] , {t, 10, tauL*R*2}] ,
 LogLogPlot[ X1[t, tauL] + Y1[t, tauL] , {t, 10, tauL*R*2}] ,
 LogLogPlot[ X0[t, tauL] + Y0[t, tauL] , {t, 10, tauL*R*2}] ,
 PlotRange → {{1, 3.7}, {-5, -1}}, Frame → True, Axes → False]
```



- Graphics -

Sensitivity to Forward Rate: Same, as previous with some filtration

normalized filtration rate is 1 in this example

```
Rc = 20;
eps = 1;
R = 3000.;
lambda = 2.88 10^-5 * 100.;
cc = 10^-4;
a = cc * (R - 1);
alpha0 = 10^5;
alphaf = alpha0 * cc;
```

```
tauL = 1.;
{X0, Y0} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alpha0 = 10^4;
alphaf = alpha0 * cc;
tauL = 1.;
{X1, Y1} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

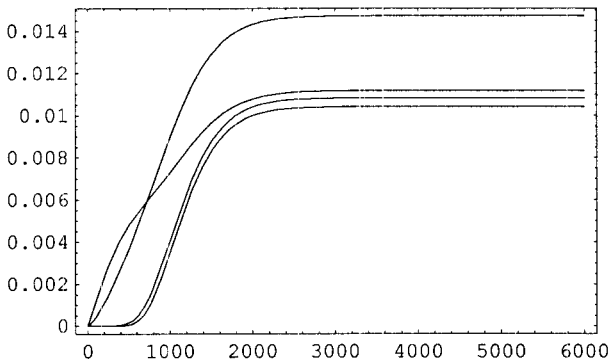
```
alpha0 = 10^3;
alphaf = alpha0 * cc;
tauL = 1.;
{X2, Y2} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alpha0 = 10^2;
alphaf = alpha0 * cc;
tauL = 1.;
{X3, Y3} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alpha0 = 10^1;
alphaf = alpha0 * cc;
tauL = 1.;
{X4, Y4} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alphaf = 0.0;
tauL = 1.;
{X00, Y00} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

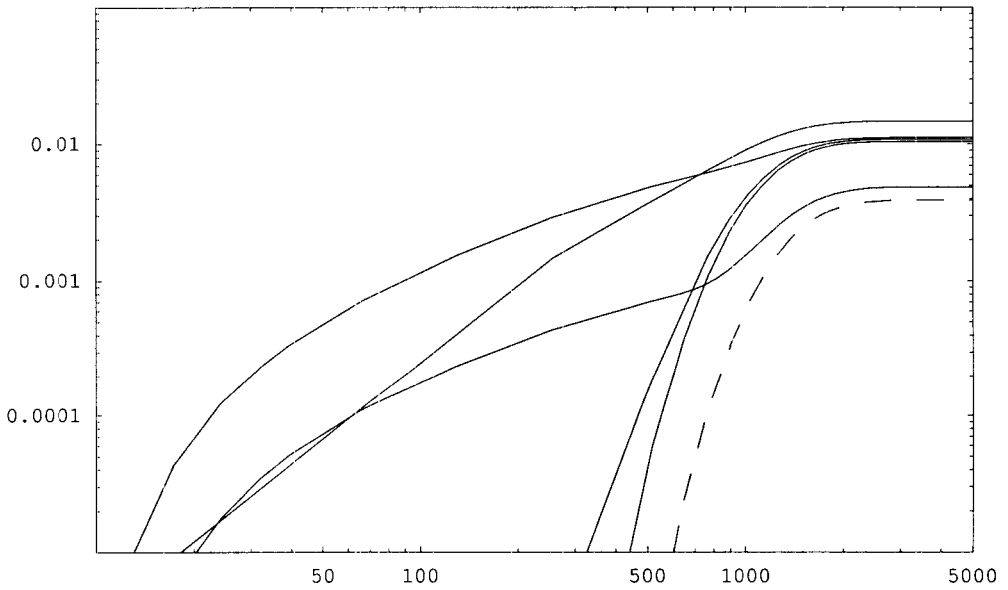
```
DisplayTogether[

 Plot[X3[t, tauL] + Y3[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X2[t, tauL] + Y2[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X1[t, tauL] + Y1[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X0[t, tauL] + Y0[t, tauL] , {t, 0, tauL*R*2}] ,
 PlotRange → All, Frame → True, Axes → False]
```



- Graphics -

28 1/18/2010

```
DisplayTogether[
 LogLogPlot[ X00[t, tauL] + Y00[t, tauL] ,
  {t, 10, tauL*R*2}, PlotStyle → Dashing[{0.03, 0.03} ] ] ,
 LogLogPlot[ X4[t, tauL] + Y4[t, tauL] , {t, 10, tauL*R*2}] ,
 LogLogPlot[ X3[t, tauL] + Y3[t, tauL] , {t, 10, tauL*R*2}] ,
 LogLogPlot[ X2[t, tauL] + Y2[t, tauL] , {t, 10, tauL*R*2}] ,
 LogLogPlot[ X1[t, tauL] + Y1[t, tauL] , {t, 10, tauL*R*2}] ,
 LogLogPlot[ X0[t, tauL] + Y0[t, tauL] , {t, 10, tauL*R*2}] ,
 PlotRange → {{1, 3.7}, {-5, -1}}, Frame → True, Axes → False]
```



- Graphics -

Sensitivity to Forward Rate: Same as previous, even stronger filtration

normalize filtration rate is 10

```
Rc = 20;
eps = 10;
R = 3000.;
lambda = 2.88 10^-5 * 100.;
cc = 10^-4;
a = cc * (R - 1);
alpha0 = 10^5;
alphaf = alpha0 * cc;
```

```
tauL = 1.;
{X0, Y0} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alpha0 = 10^4;
alphaf = alpha0 * cc;
tauL = 1.;
{X1, Y1} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

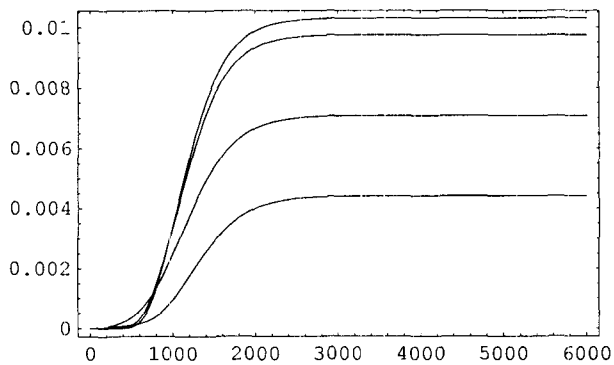
```
alpha0 = 10^3;
alphaf = alpha0 * cc;
tauL = 1.;
{X2, Y2} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alpha0 = 10^2;
alphaf = alpha0 * cc;
tauL = 1.;
{X3, Y3} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alpha0 = 10^1;
alphaf = alpha0 * cc;
tauL = 1.;
{X4, Y4} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alphaf = 0.0;
tauL = 1.;
{X00, Y00} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

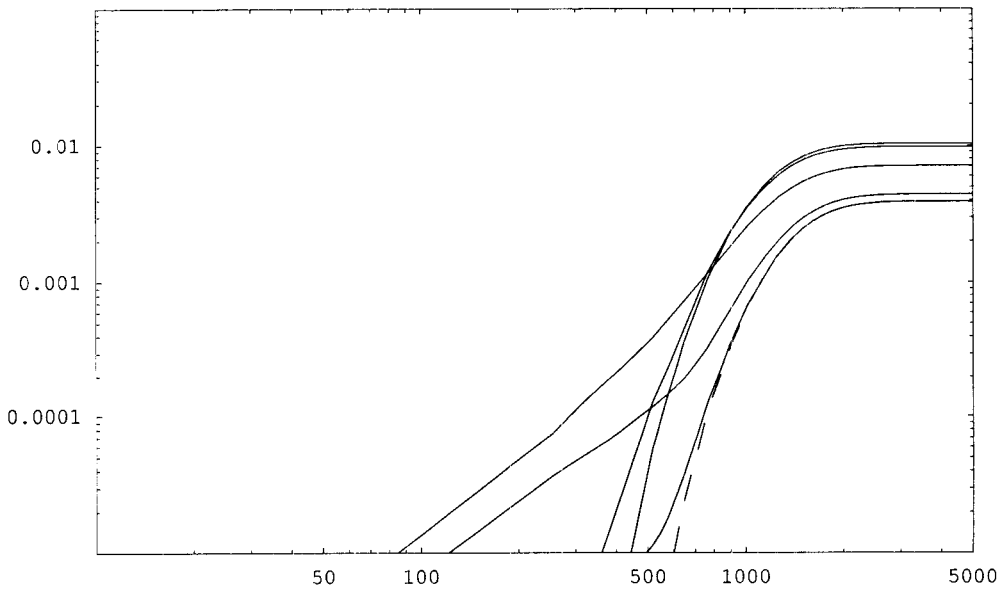
```
DisplayTogether[

 Plot[X3[t, tauL] + Y3[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X2[t, tauL] + Y2[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X1[t, tauL] + Y1[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X0[t, tauL] + Y0[t, tauL] , {t, 0, tauL*R*2}] ,
 PlotRange → All, Frame → True, Axes → False]
```



- Graphics -

ECB 1/18/2010

```
DisplayTogether[
 LogLogPlot[ X00[t, tauL] + Y00[t, tauL] ,
  {t, 10, tauL*R*2}, PlotStyle → Dashing[{0.03, 0.03} ] ] ,
 LogLogPlot[ X4[t, tauL] + Y4[t, tauL] , {t, 10, tauL*R*2}] ,
 LogLogPlot[ X3[t, tauL] + Y3[t, tauL] , {t, 10, tauL*R*2}] ,
 LogLogPlot[ X2[t, tauL] + Y2[t, tauL] , {t, 10, tauL*R*2}] ,
 LogLogPlot[ X1[t, tauL] + Y1[t, tauL] , {t, 10, tauL*R*2}] ,
 LogLogPlot[ X0[t, tauL] + Y0[t, tauL] , {t, 10, tauL*R*2}] ,
 PlotRange → {{1, 3.7}, {-5, -1}}, Frame → True, Axes → False]
```



- Graphics -

ED 1/18/2010

Parameter sensitivity with relatively fast rates

■ Colloid concentration

```
Rc = 20;
eps = 0.0;
R = 2000.;
lambda = 2.88 10^-5 * 100.;
cc = 10^-4;
a = cc * (R - 1);
alpha0 = 10^5;
alphaf = alpha0 * cc;
```

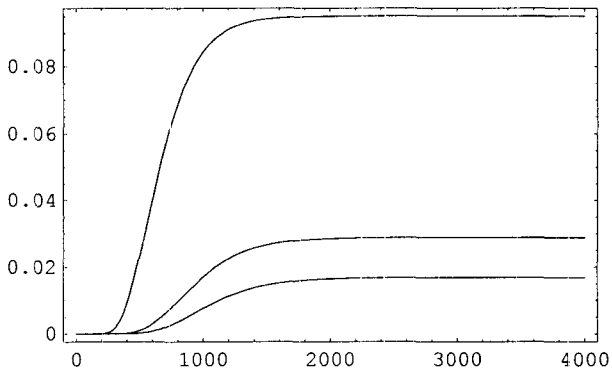
```
tauL = 1.;
{X0, Y0} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
cc = 5 10^-4;
a = cc * (R - 1);
alpha0 = 10^5;
alphaf = alpha0 * cc;
tauL = 1.;
{X1, Y1} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
cc = 10^-5;
a = cc * (R - 1);
alpha0 = 10^5;
alphaf = alpha0 * cc;
tauL = 1.;
{X2, Y2} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
cc = 10^-6;
a = cc * (R - 1);
alpha0 = 10^5;
alphaf = alpha0 * cc;
tauL = 1.;
{X3, Y3} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
DisplayTogether[
 Plot[X3[t, tauL] + Y3[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X1[t, tauL] + Y1[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X0[t, tauL] + Y0[t, tauL] , {t, 0, tauL*R*2}] ,
 PlotRange → All, Frame → True, Axes → False]
```



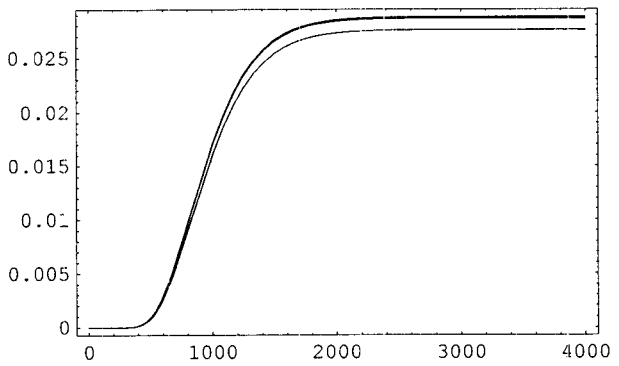
- Graphics -

Epsilon

```
Rc = 20;
eps = 0.0;
R = 2000.;
lambda = 2.88 10^-5 * 100.;
cc = 10^-4;
a = cc* (R - 1);
alpha0 = 10^5;
alphaf = alpha0 * cc;
```

```
tauL = 1.;
{X0, Y0} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];

eps = 1;
{X1, Y1} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];

eps = 10;
{X3, Y3} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];

eps = 100;
{X4, Y4} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
DisplayTogether[
 Plot[X4[t, tauL] + Y4[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X3[t, tauL] + Y3[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X1[t, tauL] + Y1[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X0[t, tauL] + Y0[t, tauL] , {t, 0, tauL*R*2}] ,
 PlotRange → All, Frame → True, Axes → False]
```



- Graphics -

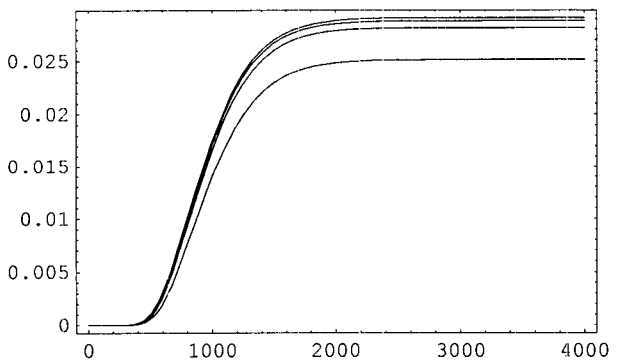
Colloid retardation

```
Rc = 20;
eps = 0.0;
R = 2000.;
lambda = 2.88 10^-5 * 100.;
cc = 10^-4;
a = cc*(R - 1);
alpha0 = 10^5;
alphaf = alpha0*cc;
```

```
tauL = 1.;
{X0, Y0} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
tauL = 1.;
Rc = 4;
{X1, Y1} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
tauL = 1.;
Rc = 100;
{X2, Y2} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];


tauL = 1.;
Rc = 500;
{X3, Y3} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];

DisplayTogether[
 Plot[X3[t, tauL] + Y3[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X2[t, tauL] + Y2[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X1[t, tauL] + Y1[t, tauL] , {t, 0, tauL*R*2}] ,
 Plot[X0[t, tauL] + Y0[t, tauL] , {t, 0, tauL*R*2}] ,
 PlotRange → All, Frame → True, Axes → False]
```



- Graphics -

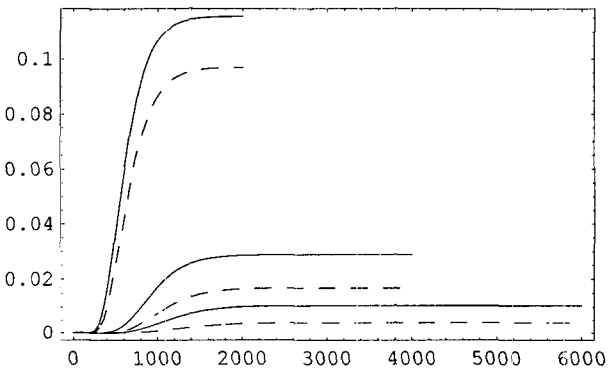
■ retardation

```
Rc = 20;
eps = 0.0;
R = 2000.;
lambda = 2.88 10^-5 * 100.;
cc = 10^-4;
a = cc*(R - 1);
alpha0 = 10^5;
alphaf = alpha0*cc;

tauL = 1.;
{X0, Y0} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
alphaf = 0.0;
{X00, Y00} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];


tauL = 1.;
R = 1000;
a = cc * (R - 1);
alphaf = alpha0 * cc;
{X1, Y1} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];

tauL = 1.;
R = 1000;
alphaf = 0.0;
{X10, Y10} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];

tauL = 1.;
R = 3000;
a = cc * (R - 1);
alphaf = alpha0 * cc;
{X2, Y2} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];

tauL = 1.;
R = 3000;
alphaf = 0.;
{X20, Y20} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

SP 1/18/2010

```
DisplayTogether[
 Plot[X2[t, tauL] + Y2[t, tauL] , {t, 0, tauL * 3000 * 2}] ,
 Plot[X20[t, tauL] + Y20[t, tauL] ,
  {t, 0, tauL * 3000 * 2}, PlotStyle → {Dashing[{0.03, 0.03}]}] ,
 Plot[X1[t, tauL] + Y1[t, tauL] , {t, 0, tauL * 1000 * 2}] ,
 Plot[X10[t, tauL] + Y10[t, tauL] ,
  {t, 0, tauL * 1000 * 2}, PlotStyle → {Dashing[{0.03, 0.03}]}] ,
 Plot[X0[t, tauL] + Y0[t, tauL] , {t, 0, tauL * 2000 * 2}] ,
 Plot[X00[t, tauL] + Y00[t, tauL] ,
  {t, 0, tauL * 2000 * 2}, PlotStyle → {Dashing[{0.03, 0.03}]}] ,
 PlotRange → All, Frame → True, Axes → False]
```



- Graphics -

RCP 1/18/2010

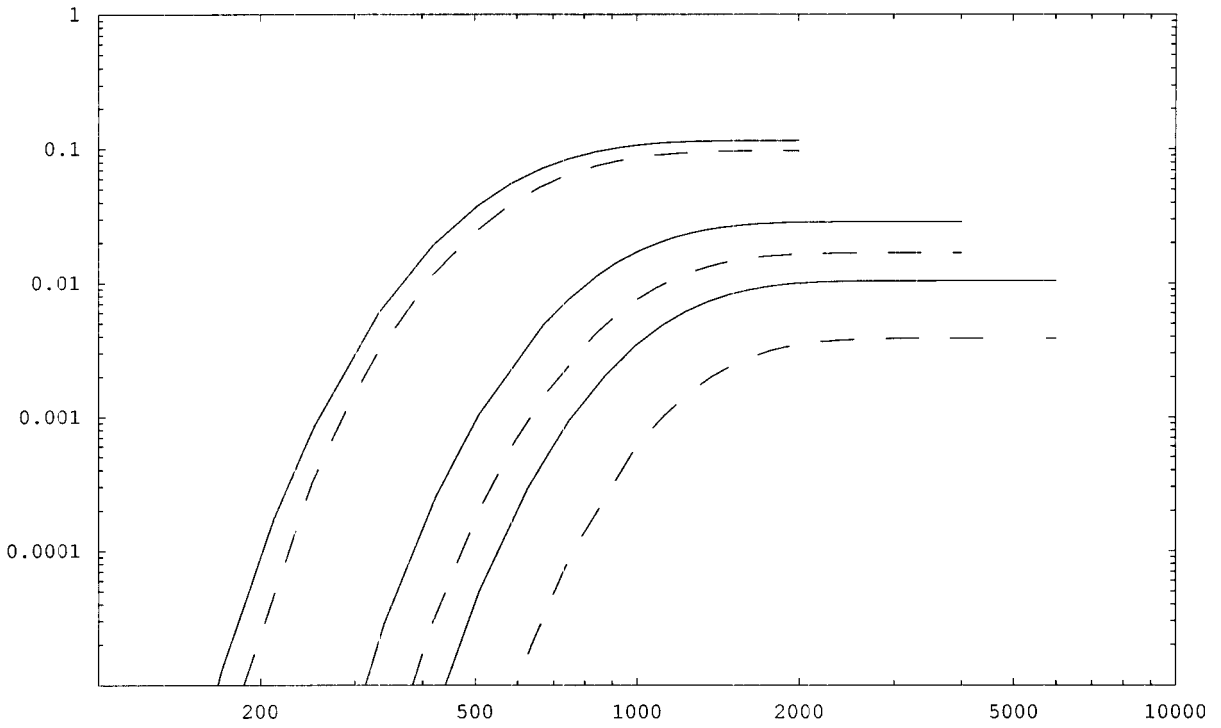
```
DisplayTogether[
 LogLogPlot[ X2[t, tauL] + Y2[t, tauL] , {t, 0, tauL * 3000 * 2}] ,
 LogLogPlot[ X20[t, tauL] + Y20[t, tauL] ,
  {t, 0, tauL * 3000 * 2}, PlotStyle → {Dashing[{0.03, 0.03}]}] ,
 LogLogPlot[ X1[t, tauL] + Y1[t, tauL] , {t, 0, tauL * 1000 * 2}] ,
 LogLogPlot[ X10[t, tauL] + Y10[t, tauL] ,
  {t, 0, tauL * 1000 * 2}, PlotStyle → {Dashing[{0.03, 0.03}]}] ,
 LogLogPlot[ X0[t, tauL] + Y0[t, tauL] , {t, 0, tauL * 2000 * 2}] ,
 LogLogPlot[ X00[t, tauL] + Y00[t, tauL] ,
  {t, 0, tauL * 2000 * 2}, PlotStyle → {Dashing[{0.03, 0.03}]}] ,
 PlotRange → {{2, 4}, {-5, 0}}, Frame → True, Axes → False]
```

ParametricPlot::pptr : $\left\{\frac{\text{Log}[t]}{\text{Log}[10]}, \frac{\text{Log}[X2[t, tauL] + Y2[t, tauL]]}{\text{Log}[10]}\right\}$
does not evaluate to a pair of real numbers at t = 0.00025`. More...

ParametricPlot::pptr : $\left\{\frac{\text{Log}[t]}{\text{Log}[10]}, \frac{\text{Log}[X20[t, tauL] + Y20[t, tauL]]}{\text{Log}[10]}\right\}$
does not evaluate to a pair of real numbers at t = 0.00025`. More...

ParametricPlot::pptr : $\left\{\frac{\text{Log}[t]}{\text{Log}[10]}, \frac{\text{Log}[X1[t, tauL] + Y1[t, tauL]]}{\text{Log}[10]}\right\}$
does not evaluate to a pair of real numbers at t = 0.00008333333333333333`. More...

General::stop : Further output of ParametricPlot::pptr will be suppressed during this calculation. More...



- Graphics -

■ lambda (decay rate)

```
Rc = 20;
eps = 0.0;
R = 2000.;
lambda = 2.88 10^-5 * 100.;
cc = 10^-4;
a = cc * (R - 1);
alpha0 = 10^5;
alphaf = alpha0 * cc;
```

```
tauL = 1.;
{X0, Y0} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];

alphaf = 0.0;
{X00, Y00} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

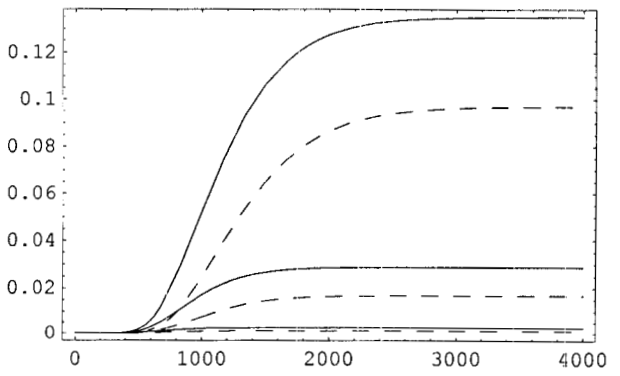
```
tauL = 1.;
lambda = lambda / 2.;
alphaf = alpha0 * cc;
{X1, Y1} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
tauL = 1.;
alphaf = 0.0;
{X10, Y10} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
tauL = 1.;
lambda = lambda * 4.;
alphaf = alpha0 * cc;
{X2, Y2} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

```
tauL = 1.;
alphaf = 0.;
{X20, Y20} = SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
```

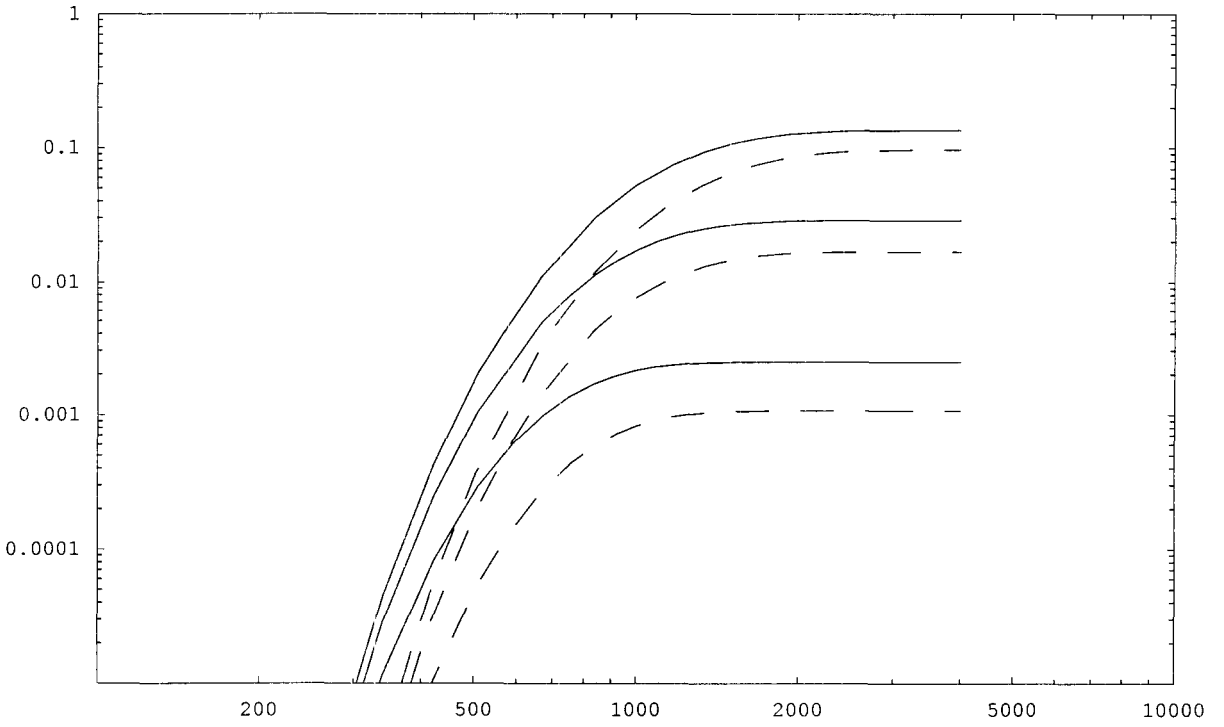
```
DisplayTogether[
 Plot[ X2[t, tauL] + Y2[t, tauL] , {t, 0, tauL * 2000 * 2}] ,
 Plot[ X20[t, tauL] + Y20[t, tauL] ,
  {t, 0, tauL * 2000 * 2}, PlotStyle → {Dashing[{0.03, 0.03}]}] ,
 Plot[ X1[t, tauL] + Y1[t, tauL] , {t, 0, tauL * 2000 * 2}] ,
 Plot[ X10[t, tauL] + Y10[t, tauL] ,
  {t, 0, tauL * 2000 * 2}, PlotStyle → {Dashing[{0.03, 0.03}]}] ,
 Plot[ X0[t, tauL] + Y0[t, tauL] , {t, 0, tauL * 2000 * 2}] ,
 Plot[ X00[t, tauL] + Y00[t, tauL] ,
  {t, 0, tauL * 2000 * 2}, PlotStyle → {Dashing[{0.03, 0.03}]}] ,
 PlotRange → All, Frame → True, Axes → False]
```



- Graphics -

RCB 1/18/2010

```
DisplayTogether[
 LogLogPlot[ X2[t, tauL] + Y2[t, tauL] , {t, 0, tauL*2000*2}] ,
 LogLogPlot[ X20[t, tauL] + Y20[t, tauL] ,
  {t, 0, tauL*2000*2}, PlotStyle → {Dashing[{0.03, 0.03}]}] ,
 LogLogPlot[ X1[t, tauL] + Y1[t, tauL] , {t, 0, tauL*2000*2}] ,
 LogLogPlot[ X10[t, tauL] + Y10[t, tauL] ,
  {t, 0, tauL*2000*2}, PlotStyle → {Dashing[{0.03, 0.03}]}] ,
 LogLogPlot[ X0[t, tauL] + Y0[t, tauL] , {t, 0, tauL*2000*2}] ,
 LogLogPlot[ X00[t, tauL] + Y00[t, tauL] ,
  {t, 0, tauL*2000*2}, PlotStyle → {Dashing[{0.03, 0.03}]}] ,
 PlotRange → {{2, 4}, {-5, 0}}, Frame → True, Axes → False]
```



- Graphics -

Colloid Facilitated Transport in Alluvium: Figure 5 of IHLRWMC 2006

This is a Monte Carlo calculation based on sampling uncertain parameters.
The colloid retardation factor is not trnucated at 8 in this example.

Scott Painter
12.21.2005

Preliminaries

```
<< Statistics`DescriptiveStatistics`
<< Graphics`Graphics`
<< Statistics`ContinuousDistributions`
$HistoryLength = 5
```

5

Define the breakthrough solution

rates are in units of 1/year

```
SolveIt[alphaf_, R_, Rc_,
  eps_, a_, lambda_, tauL_] := Module[
  {alphar = alphaf / a,
Deff = 0.1 * tauL,
   a11, a12, a13, a21, a22, a23, a31, a32, a33,
   solution, X, Y},
  a11 = -(alphaf Rc + lambda R);
a12 = Rc alphar;
a13 = alphar;
a21 = Rc alphaf;
```

```
a22 = - (Rc alphar + eps + lambda Rc);
a23 = 0.;
a31 = 0.;
a32 = eps;
a33 = - (lambda + alphar);
solution = NDSolve[
    {R ∂t u[t, x] == -∂x u[t, x] + Deff ∂x,x u[t, x] +
       a11 u[t, x] + a12 v[t, x] + a13 w[t, x],
     Rc ∂t v[t, x] == -∂x v[t, x] + Deff ∂x,x v[t, x] +
       a22 v[t, x] + a21 u[t, x],
     ∂t w[t, x] == a32 v[t, x] + a33 w[t, x],
     u[0, x] == 0.,
     u[t, 0] == Min[t / 2., 1],
     u[t, tauL * 3.] == 0,
     v[0, x] == 0.,
     v[t, tauL * 3.] == 0.,
     v[t, 0] == 0.,
     w[0, x] == 0.},
    {u, v, w}, {t, 0, 300000.}, {x, 0, tauL * 3.},
    StartingStepSize → {300000., tauL / 2000.},
    MaxStepSize → {300000., tauL / 500.},
    MaxSteps → {20000, 20000}];
{X, Y} = {u, v} /. First[solution];
  Clear[solution];
  {X, Y}
 ]
```

Presample

These are based on TPA 5.0 input

```
nreal = 500;
```

■ Tau on Central Streamtube

```
centralst = Partition[{0.0, 1500., 0.,
1.0, 1500., 1350.,
3.0, 450., 4350.,
9.5, 250., 10850.,
13.0, 400., 14350.,
13.5 , 375., 14850.,
15.0, 325., 16350.,
18.0 , 225., 19350.}, 3] ;
centralst = Transpose[centralst];
centralst1 = Transpose[{centralst[[1]], centralst[[3]]}]
centralst2 = Transpose[{centralst[[3]], centralst[[2]]}]
```

```
{{0., 0.}, {1., 1350.}, {3., 4350.}, {9.5, 10850.},
 {13., 14350.}, {13.5, 14850.}, {15., 16350.}, {18., 19350.}}
```

```
{{0., 1500.}, {1350., 1500.}, {4350., 450.}, {10850., 250.},
 {14350., 400.}, {14850., 375.}, {16350., 325.}, {19350., 225.}}
```

```
cstfunc1 = Interpolation[centralst1]
cstfunc2 = Interpolation[centralst2]
```

```
InterpolatingFunction[{{0., 18.}}, <>]
```

```
InterpolatingFunction[{{0., 19350.}}, <>]
```

```
Q = 252;
taudist = Table[
   sldist = Random[ Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, tddist, 18000},
      MinRecursion → 5, MaxRecursion → 12] / (18000 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tau = (18000 - tddist) / velocity;
   Max[tau, 10.], {nreal}];
```

■ Colloid retardation remove truncation at 8

```
Rdist = Interpolation[
   Transpose[ {{0, 0.331, 0.5, 1.0}, {0.0, Log[8], Log[33.96], Log[5188.]}} ]];
Rctable = Table[ Exp[ Rdist[Random[]]], {nreal}];
```

■ Retardation

```
ldist = LogNormalDistribution[9.45708, 1.10];
Rtable = Table[ Random[ ldist], {nreal}];
```

```
General::spell1 :
 Possible spelling error: new symbol name "ldist" is similar to existing symbol "sldist". More…

General::spell1 :
 Possible spelling error: new symbol name "Rtable" is similar to existing symbol "Rctable". More…
```

■ Colloid Concentration

```
ccfunc = Interpolation[ Transpose[
    {{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.75, 0.8, 0.85, 0.9, 0.94, 0.98, 0.99, 1.0},
     {10^-6, 2.51 10^-6, 6.31 10^-6, 1.58 10^-5, 3.98 10^-5, 1.0 10^-4,
      2.51 10^-4, 6.31 10^-4, 1.0 10^-3, 2.15 10^-3, 4.64 10^-3,
      1.0 10^-2, 2.24 10^-2, 0.05, 0.1, 0.2}} ], InterpolationOrder → 1];

cctable = Table[ ccfunc[ Random[]], {nreal}];
```

Sample, setting alpha0=10

```
alpha0 = 10
```

```
10
```

RCS 1/18/2010

```
bigresult = Table[
eps = 0.0;
tauL = taudist[[i]];
   R = Rtable[[i]];
   Rc = Rctable[[i]];
   lambda = 2.88 10^-5 ;
   cc = cctable[[i]];
   fac = 590 / 2500 * 0.12 / (1 - 0.12);
   cc = cc * fac;
   a = cc * (R - 1);
   alphaf = alpha0 * cc / fac ;

   {X1, Y1} =
    SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
   Table[ X1[t, tauL] + Y1[t, tauL] ,
    {t, 0, 100000, 100}],
   {i, 1, nreal}];
```

```
time = (Range[1001] - 1) * 100000. / 1000.;
```

```
meancurve10 = Apply[Plus, bigresult] / Length[bigresult]
```

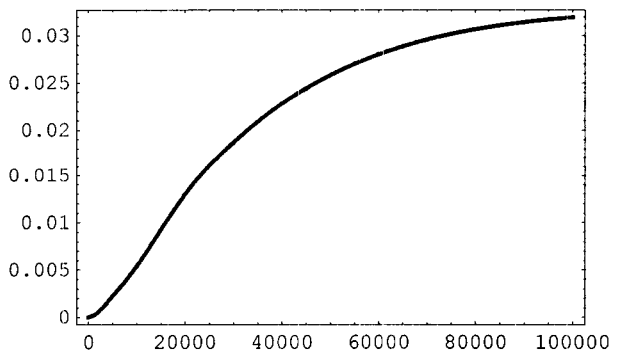
{0., 9.69867×10^{-6}, 0.0000255948, 0.0000423455, 0.0000598622, 0.0000781408, 0.0000972679,
0.000117372, 0.000138593, 0.000161078, 0.000184968, 0.000210399, 0.000237495,
0.000266361, 0.000297092, 0.000329759, 0.000364412, 0.000401079, 0.000439757,
0.000480442, 0.000523094, 0.000567647, 0.000614044, 0.000662183, 0.000711966,
0.000763275, 0.000816003, 0.000869997, 0.000925144, 0.000981291, 0.00103832,
0.00109611, 0.00115453, 0.00121348, 0.00127285, 0.00133256, 0.00139249, 0.0014526,
0.00151279, 0.00157301, 0.00163321, 0.00169336, 0.0017534, 0.00181331, 0.00187309,
0.00193271, 0.00199216, 0.00205146, 0.00211059, 0.00216955, 0.00222837, 0.00228708,
0.00234567, 0.00240415, 0.00246256, 0.00252094, 0.0025793, 0.00263764, 0.002696,
0.00275443, 0.00281295, 0.00287155, 0.00293026, 0.00298914, 0.0030482, 0.00310746,
0.00316692, 0.00322662, 0.00328658, 0.00334682, 0.00340734, 0.00346816, 0.00352932,
0.00359083, 0.00365268, 0.0037149, 0.00377748, 0.00384045, 0.00390382, 0.0039676,
0.00403178, 0.00409637, 0.00416137, 0.00422679, 0.00429264, 0.00435891, 0.0044256,
0.00449272, 0.00456025, 0.0046282, 0.00469656, 0.00476534, 0.00483454, 0.00490413,
0.00497412, 0.00504451, 0.00511528, 0.00518644, 0.00525797, 0.00532987, 0.00540213,
0.00547473, 0.00554768, 0.00562096, 0.00569458, 0.00576851, 0.00584274, 0.00591728,
0.00599209, 0.00606718, 0.00614254, 0.00621816, 0.00629403, 0.00637014, 0.00644648,
0.00652303, 0.00659978, 0.00667673, 0.00675386, 0.00683117, 0.00690865, 0.00698628,
0.00706405, 0.00714196, 0.00721999, 0.00729813, 0.00737639, 0.00745474, 0.00753319,

0.0076117, 0.00769028, 0.0077689, 0.00784757, 0.00792628, 0.00800502, 0.00808379,
0.00816258, 0.00824136, 0.00832014, 0.0083989, 0.00847764, 0.00855634, 0.008635,
0.00871362, 0.00879219, 0.0088707, 0.00894914, 0.00902751, 0.0091058, 0.009184,
0.0092621, 0.00934011, 0.009418, 0.00949579, 0.00957345, 0.00965099, 0.0097284,
0.00980567, 0.00988281, 0.00995981, 0.0100367, 0.0101134, 0.0101899, 0.0102662,
0.0103424, 0.0104185, 0.0104943, 0.01057, 0.0106455, 0.0107208, 0.0107959, 0.0108708,
0.0109455, 0.0110201, 0.0110944, 0.0111685, 0.0112424, 0.0113161, 0.0113896,
0.0114628, 0.0115359, 0.0116087, 0.0116814, 0.0117537, 0.0118259, 0.0118979,
0.0119696, 0.012041, 0.0121123, 0.0121833, 0.0122541, 0.0123246, 0.0123949, 0.012465,
0.0125348, 0.0126044, 0.0126738, 0.0127429, 0.0128118, 0.0128804, 0.0129488,
0.0130169, 0.0130848, 0.0131525, 0.0132199, 0.0132871, 0.013354, 0.0134207, 0.0134872,
0.0135534, 0.0136193, 0.0136851, 0.0137505, 0.0138158, 0.0138808, 0.0139455,
0.0140101, 0.0140743, 0.0141384, 0.0142022, 0.0142657, 0.0143291, 0.0143922,
0.014455, 0.0145176, 0.01458, 0.0146422, 0.0147041, 0.0147658, 0.0148272, 0.0148885,
0.0149495, 0.0150102, 0.0150708, 0.0151311, 0.0151912, 0.015251, 0.0153107,
0.0153701, 0.0154293, 0.0154883, 0.015547, 0.0156055, 0.0156639, 0.015722, 0.0157798,
0.0158375, 0.015895, 0.0159522, 0.0160093, 0.0160661, 0.0161227, 0.0161791,
0.0162353, 0.0162913, 0.0163471, 0.0164027, 0.0164581, 0.0165132, 0.0165682,
0.016623, 0.0166776, 0.016732, 0.0167862, 0.0168402, 0.016894, 0.0169476, 0.017001,
0.0170543, 0.0171073, 0.0171602, 0.0172128, 0.0172653, 0.0173176, 0.0173697,
0.0174217, 0.0174734, 0.017525, 0.0175764, 0.0176276, 0.0176787, 0.0177296,
0.0177802, 0.0178308, 0.0178811, 0.0179313, 0.0179812, 0.0180311, 0.0180807,
0.0181302, 0.0181795, 0.0182287, 0.0182777, 0.0183265, 0.0183752, 0.0184237,
0.0184721, 0.0185202, 0.0185683, 0.0186161, 0.0186639, 0.0187114, 0.0187588,
0.018806, 0.0188531, 0.0189, 0.0189468, 0.0189934, 0.0190399, 0.0190862, 0.0191324,
0.0191784, 0.0192243, 0.01927, 0.0193156, 0.0193611, 0.0194064, 0.0194515, 0.0194965,
0.0195414, 0.0195861, 0.0196307, 0.0196752, 0.0197195, 0.0197636, 0.0198077,
0.0198516, 0.0198953, 0.0199389, 0.0199824, 0.0200257, 0.0200689, 0.020112, 0.020155,
0.0201978, 0.0202404, 0.020283, 0.0203254, 0.0203677, 0.0204099, 0.0204519,
0.0204938, 0.0205356, 0.0205772, 0.0206187, 0.0206601, 0.0207014, 0.0207426,
0.0207836, 0.0208245, 0.0208653, 0.0209059, 0.0209465, 0.0209869, 0.0210272,
0.0210673, 0.0211074, 0.0211473, 0.0211871, 0.0212268, 0.0212664, 0.0213058,
0.0213452, 0.0213844, 0.0214235, 0.0214625, 0.0215013, 0.0215401, 0.0215787,
0.0216173, 0.0216557, 0.021694, 0.0217322, 0.0217703, 0.0218083, 0.0218461,
0.0218839, 0.0219215, 0.021959, 0.0219964, 0.0220337, 0.0220709, 0.022108, 0.0221449,
0.0221818, 0.0222186, 0.0222552, 0.0222917, 0.0223281, 0.0223645, 0.0224007,
0.0224368, 0.0224728, 0.0225087, 0.0225445, 0.0225802, 0.0226158, 0.0226512,
0.0226866, 0.0227219, 0.0227571, 0.0227921, 0.0228271, 0.022862, 0.0228967,
0.0229314, 0.0229659, 0.0230004, 0.0230347, 0.023069, 0.0231031, 0.0231372,
0.0231711, 0.023205, 0.0232387, 0.0232724, 0.0233059, 0.0233394, 0.0233727,
0.023406, 0.0234391, 0.0234722, 0.0235051, 0.023538, 0.0235707, 0.0236034, 0.023636,
0.0236684, 0.0237008, 0.0237331, 0.0237653, 0.0237974, 0.0238293, 0.0238612,
0.023893, 0.0239247, 0.0239564, 0.0239879, 0.0240193, 0.0240506, 0.0240819, 0.024113,
0.0241441, 0.024175, 0.0242059, 0.0242367, 0.0242673, 0.0242979, 0.0243284,
0.0243588, 0.0243891, 0.0244194, 0.0244495, 0.0244795, 0.0245095, 0.0245393,
0.0245691, 0.0245987, 0.0246283, 0.0246578, 0.0246872, 0.0247165, 0.0247457,
0.0247749, 0.0248039, 0.0248329, 0.0248617, 0.0248905, 0.0249192, 0.0249478,
0.0249764, 0.0250048, 0.0250331, 0.0250614, 0.0250896, 0.0251177, 0.0251457,
0.0251736, 0.0252014, 0.0252292, 0.0252568, 0.0252844, 0.0253119, 0.0253393,
0.0253666, 0.0253938, 0.0254209, 0.025448, 0.025475, 0.0255019, 0.0255287,
0.0255554, 0.025582, 0.0256086, 0.025635, 0.0256614, 0.0256877, 0.0257139,
0.0257401, 0.0257661, 0.0257921, 0.025818, 0.0258438, 0.0258696, 0.0258952,
0.0259208, 0.0259463, 0.0259717, 0.025997, 0.0260223, 0.0260474, 0.0260725,
0.0260975, 0.0261225, 0.0261473, 0.0261721, 0.0261968, 0.0262214, 0.026246,

```
0.0262704, 0.0262948, 0.0263191, 0.0263433, 0.0263674, 0.0263915, 0.0264155,
0.0264394, 0.0264632, 0.026487, 0.0265106, 0.0265342, 0.0265578, 0.0265812,
0.0266046, 0.0266279, 0.0266511, 0.0266742, 0.0266973, 0.0267203, 0.0267432,
0.0267661, 0.0267888, 0.0268115, 0.0268342, 0.0268567, 0.0268792, 0.0269016,
0.0269239, 0.0269462, 0.0269684, 0.0269905, 0.0270125, 0.0270345, 0.0270564,
0.0270782, 0.0270999, 0.0271216, 0.0271432, 0.0271647, 0.0271862, 0.0272076,
0.0272289, 0.0272501, 0.0272713, 0.0272924, 0.0273134, 0.0273344, 0.0273553,
0.0273761, 0.0273969, 0.0274175, 0.0274382, 0.0274587, 0.0274792, 0.0274996,
0.0275199, 0.0275402, 0.0275604, 0.0275806, 0.0276006, 0.0276206, 0.0276406,
0.0276604, 0.0276802, 0.0277, 0.0277196, 0.0277392, 0.0277587, 0.0277782, 0.0277976,
0.0278169, 0.0278362, 0.0278554, 0.0278745, 0.0278936, 0.0279126, 0.0279315,
0.0279504, 0.0279692, 0.027988, 0.0280066, 0.0280253, 0.0280438, 0.0280623,
0.0280807, 0.0280991, 0.0281174, 0.0281356, 0.0281538, 0.0281719, 0.0281899,
0.0282079, 0.0282259, 0.0282437, 0.0282615, 0.0282793, 0.0282969, 0.0283146,
0.0283321, 0.0283496, 0.028367, 0.0283844, 0.0284017, 0.028419, 0.0284362,
0.0284533, 0.0284704, 0.0284874, 0.0285043, 0.0285212, 0.028538, 0.0285548,
0.0285715, 0.0285882, 0.0286048, 0.0286213, 0.0286378, 0.0286542, 0.0286706,
0.0286869, 0.0287032, 0.0287193, 0.0287355, 0.0287516, 0.0287676, 0.0287836,
0.0287995, 0.0288153, 0.0288311, 0.0288469, 0.0288625, 0.0288782, 0.0288937,
0.0289093, 0.0289247, 0.0289401, 0.0289555, 0.0289708, 0.028986, 0.0290012,
0.0290164, 0.0290314, 0.0290465, 0.0290615, 0.0290764, 0.0290912, 0.0291061,
0.0291208, 0.0291355, 0.0291502, 0.0291648, 0.0291793, 0.0291938, 0.0292083,
0.0292227, 0.029237, 0.0292513, 0.0292656, 0.0292798, 0.0292939, 0.029308,
0.029322, 0.029336, 0.02935, 0.0293638, 0.0293777, 0.0293915, 0.0294052, 0.0294189,
0.0294325, 0.0294461, 0.0294597, 0.0294732, 0.0294866, 0.0295, 0.0295134, 0.0295267,
0.0295399, 0.0295531, 0.0295663, 0.0295794, 0.0295924, 0.0296054, 0.0296184,
0.0296313, 0.0296442, 0.029657, 0.0296698, 0.0296825, 0.0296952, 0.0297079,
0.0297205, 0.029733, 0.0297455, 0.029758, 0.0297704, 0.0297827, 0.029795, 0.0298073,
0.0298195, 0.0298317, 0.0298439, 0.029856, 0.029868, 0.02988, 0.029892, 0.0299039,
0.0299158, 0.0299276, 0.0299394, 0.0299511, 0.0299628, 0.0299745, 0.0299861,
0.0299977, 0.0300092, 0.0300207, 0.0300322, 0.0300436, 0.0300549, 0.0300663,
0.0300776, 0.0300888, 0.0301, 0.0301111, 0.0301223, 0.0301333, 0.0301444, 0.0301554,
0.0301663, 0.0301772, 0.0301881, 0.0301989, 0.0302097, 0.0302205, 0.0302312,
0.0302419, 0.0302525, 0.0302631, 0.0302737, 0.0302842, 0.0302947, 0.0303051,
0.0303155, 0.0303259, 0.0303362, 0.0303465, 0.0303567, 0.030367, 0.0303771,
0.0303873, 0.0303974, 0.0304074, 0.0304174, 0.0304274, 0.0304374, 0.0304473,
0.0304572, 0.030467, 0.0304768, 0.0304866, 0.0304963, 0.030506, 0.0305157,
0.0305253, 0.0305349, 0.0305444, 0.0305539, 0.0305634, 0.0305729, 0.0305823,
0.0305916, 0.030601, 0.0306103, 0.0306196, 0.0306288, 0.030638, 0.0306472,
0.0306563, 0.0306654, 0.0306745, 0.0306835, 0.0306925, 0.0307014, 0.0307104,
0.0307193, 0.0307281, 0.030737, 0.0307458, 0.0307545, 0.0307633, 0.030772,
0.0307806, 0.0307893, 0.0307979, 0.0308064, 0.030815, 0.0308235, 0.030832,
0.0308404, 0.0308488, 0.0308572, 0.0308656, 0.0308739, 0.0308822, 0.0308904,
0.0308987, 0.0309069, 0.030915, 0.0309232, 0.0309313, 0.0309393, 0.0309474,
0.0309554, 0.0309634, 0.0309713, 0.0309792, 0.0309871, 0.030995, 0.0310028,
0.0310106, 0.0310184, 0.0310262, 0.0310339, 0.0310416, 0.0310492, 0.0310569,
0.0310645, 0.031072, 0.0310796, 0.0310871, 0.0310946, 0.0311021, 0.0311095,
0.0311169, 0.0311243, 0.0311316, 0.031139, 0.0311462, 0.0311535, 0.0311608,
0.031168, 0.0311752, 0.0311823, 0.0311894, 0.0311966, 0.0312036, 0.0312107,
0.0312177, 0.0312247, 0.0312317, 0.0312386, 0.0312456, 0.0312525, 0.0312593,
0.0312662, 0.031273, 0.0312798, 0.0312865, 0.0312933, 0.0313, 0.0313067,
0.0313134, 0.03132, 0.0313266, 0.0313332, 0.0313398, 0.0313463, 0.0313529,
0.0313593, 0.0313658, 0.0313723, 0.0313787, 0.0313851, 0.0313915, 0.0313978,
0.0314041, 0.0314104, 0.0314167, 0.031423, 0.0314292, 0.0314354, 0.0314416,
```

```
0.0314478, 0.0314539, 0.03146, 0.0314661, 0.0314722, 0.0314782, 0.0314843,
0.0314903, 0.0314962, 0.0315022, 0.0315081, 0.031514, 0.0315199, 0.0315258,
0.0315316, 0.0315375, 0.0315433, 0.031549, 0.0315548, 0.0315605, 0.0315663,
0.0315719, 0.0315776, 0.0315833, 0.0315889, 0.0315945, 0.0316001, 0.0316057,
0.0316112, 0.0316167, 0.0316222, 0.0316277, 0.0316332, 0.0316386, 0.0316441,
0.0316495, 0.0316548, 0.0316602, 0.0316655, 0.0316709, 0.0316762, 0.0316815,
0.0316867, 0.031692, 0.0316972, 0.0317024, 0.0317076, 0.0317127, 0.0317179,
0.031723, 0.0317281, 0.0317332, 0.0317383, 0.0317433, 0.0317484, 0.0317534,
0.0317584, 0.0317634, 0.0317683, 0.0317733, 0.0317782, 0.0317831, 0.031788,
0.0317928, 0.0317977, 0.0318025, 0.0318073, 0.0318121, 0.0318169, 0.0318217,
0.0318264, 0.0318311, 0.0318358, 0.0318405, 0.0318452, 0.0318498, 0.0318545,
0.0318591, 0.0318637, 0.0318683, 0.0318728, 0.0318774, 0.0318819, 0.0318864,
0.0318909, 0.0318954, 0.0318999, 0.0319043, 0.0319088, 0.0319132, 0.0319176,
0.0319219, 0.0319263, 0.0319307, 0.031935, 0.0319393, 0.0319436, 0.0319479}
```

```
ListPlot[ Transpose[{time, meancurve10}], PlotJoined → True,
 PlotStyle → AbsoluteThickness[2], Frame → True, Axes → False ]
```



- Graphics -

Sample, setting alpha0=100

```
alpha0 = 100
```

100

Rd 1/18/2010

```
bigresult = Table[
eps = 0.0;
tauL = taudist[[i]];
   R = Rtable[[i]];
   Rc = Rctable[[i]];
   lambda = 2.88 10^-5 ;
   cc = cctable[[i]];
   fac = 590 / 2500 * 0.12 / (1 - 0.12);
   cc = cc * fac;
   a = cc * (R - 1);
   alphaf = alpha0 * cc / fac ;

   {X1, Y1} =
    SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
   Table[ X1[t, tauL] + Y1[t, tauL] ,
    {t, 0, 100000, 100}],
   {i, 1, nreal}];
```

```
time = (Range[1001] - 1) * 100000. / 1000.;
```

```
meancurve100 = Apply[Plus, bigresult] / Length[bigresult]
```

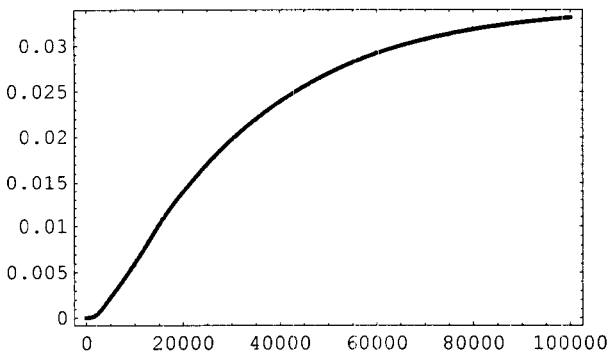
```
{0., 5.07594×10^-6, 0.0000115678, 0.0000184365, 0.0000256823, 0.0000333085, 0.0000413245,
 0.0000497574, 0.0000586665, 0.00006816, 0.0000784127, 0.0000896754, 0.000102264,
 0.000116544, 0.000132905, 0.000151734, 0.000173387, 0.000198165, 0.000226299,
 0.000257947, 0.000293187, 0.000332032, 0.000374411, 0.000420215, 0.00046926,
 0.000521356, 0.000576277, 0.000633781, 0.000693599, 0.000755511, 0.000819225,
 0.000884553, 0.000951227, 0.00101908, 0.0010879, 0.00115753, 0.0012278, 0.00129861,
 0.00136981, 0.00144132, 0.00151304, 0.0015849, 0.00165684, 0.00172883, 0.00180083,
 0.00187279, 0.00194471, 0.00201661, 0.00208843, 0.0021602, 0.00223194, 0.00230363,
 0.00237529, 0.00244696, 0.00251862, 0.00259029, 0.00266201, 0.00273379, 0.00280562,
 0.00287756, 0.00294961, 0.00302177, 0.00309407, 0.00316653, 0.00323916, 0.00331196,
 0.00338495, 0.00345815, 0.00353157, 0.0036052, 0.00367907, 0.00375317, 0.00382752,
 0.00390212, 0.00397698, 0.00405209, 0.00412747, 0.00420311, 0.00427902, 0.00435521,
 0.00443165, 0.00450835, 0.00458532, 0.00466255, 0.00474004, 0.00481779, 0.00489579,
 0.00497402, 0.0050525, 0.00513122, 0.00521017, 0.00528936, 0.00536878, 0.0054484,
 0.00552823, 0.00560825, 0.00568846, 0.00576887, 0.00584947, 0.00593026, 0.0060112,
 0.0060923, 0.00617355, 0.00625493, 0.00633645, 0.00641812, 0.00649991, 0.00658182,
 0.00666383, 0.00674594, 0.00682813, 0.00691041, 0.00699277, 0.0070752, 0.00715769,
 0.00724023, 0.00732282, 0.00740544, 0.0074881, 0.00757079, 0.00765349, 0.00773619,
 0.00781889, 0.00790159, 0.00798428, 0.00806695, 0.00814961, 0.00823224, 0.00831482,
```

0.00839735, 0.00847982, 0.00856224, 0.00864459, 0.00872688, 0.0088091, 0.00889123,
0.00897328, 0.00905523, 0.00913707, 0.00921882, 0.00930046, 0.00938198, 0.00946338,
0.00954465, 0.00962579, 0.0097068, 0.00978767, 0.00986839, 0.00994897, 0.0100294,
0.0101097, 0.0101898, 0.0102697, 0.0103495, 0.0104291, 0.0105085, 0.0105878,
0.0106668, 0.0107457, 0.0108244, 0.0109029, 0.0109813, 0.0110594, 0.0111373, 0.011215,
0.0112925, 0.0113698, 0.0114469, 0.0115237, 0.0116004, 0.0116768, 0.0117531,
0.0118291, 0.0119048, 0.0119804, 0.0120558, 0.0121309, 0.0122058, 0.0122804,
0.0123548, 0.012429, 0.012503, 0.0125767, 0.0126502, 0.0127235, 0.0127965, 0.0128693,
0.0129418, 0.0130141, 0.0130862, 0.013158, 0.0132296, 0.013301, 0.0133721, 0.0134429,
0.0135135, 0.0135839, 0.0136541, 0.0137239, 0.0137936, 0.013863, 0.0139322,
0.0140011, 0.0140698, 0.0141383, 0.0142065, 0.0142744, 0.0143421, 0.0144096,
0.0144768, 0.0145438, 0.0146105, 0.0146771, 0.0147433, 0.0148094, 0.0148752,
0.0149407, 0.015006, 0.0150711, 0.015136, 0.0152006, 0.015265, 0.0153291, 0.015393,
0.0154567, 0.0155202, 0.0155834, 0.0156464, 0.0157091, 0.0157717, 0.015834, 0.015896,
0.0159579, 0.0160195, 0.0160809, 0.0161421, 0.0162031, 0.0162638, 0.0163243,
0.0163846, 0.0164447, 0.0165046, 0.0165642, 0.0166237, 0.0166829, 0.0167419,
0.0168007, 0.0168593, 0.0169177, 0.0169758, 0.0170338, 0.0170915, 0.0171491,
0.0172064, 0.0172636, 0.0173205, 0.0173772, 0.0174338, 0.0174901, 0.0175463,
0.0176022, 0.0176579, 0.0177135, 0.0177689, 0.017824, 0.017879, 0.0179337, 0.0179883,
0.0180427, 0.0180969, 0.0181509, 0.0182048, 0.0182584, 0.0183119, 0.0183651,
0.0184183, 0.0184712, 0.0185239, 0.0185765, 0.0186289, 0.0186811, 0.0187331,
0.0187849, 0.0188366, 0.0188881, 0.0189394, 0.0189905, 0.0190415, 0.0190923,
0.0191429, 0.0191934, 0.0192437, 0.0192939, 0.0193438, 0.0193936, 0.0194433,
0.0194927, 0.0195421, 0.0195912, 0.0196402, 0.019689, 0.0197377, 0.0197862,
0.0198345, 0.0198827, 0.0199308, 0.0199787, 0.0200264, 0.020074, 0.0201214,
0.0201687, 0.0202158, 0.0202627, 0.0203095, 0.0203562, 0.0204027, 0.0204491,
0.0204953, 0.0205414, 0.0205873, 0.0206331, 0.0206787, 0.0207242, 0.0207695,
0.0208147, 0.0208598, 0.0209047, 0.0209495, 0.0209941, 0.0210386, 0.021083,
0.0211272, 0.0211713, 0.0212152, 0.021259, 0.0213027, 0.0213462, 0.0213896,
0.0214329, 0.021476, 0.021519, 0.0215618, 0.0216046, 0.0216471, 0.0216896, 0.0217319,
0.0217741, 0.0218162, 0.0218582, 0.0219, 0.0219417, 0.0219832, 0.0220246, 0.022066,
0.0221071, 0.0221482, 0.0221891, 0.0222299, 0.0222706, 0.0223111, 0.0223516,
0.0223919, 0.022432, 0.0224721, 0.022512, 0.0225518, 0.0225915, 0.0226311, 0.0226706,
0.0227099, 0.0227491, 0.0227882, 0.0228272, 0.022866, 0.0229048, 0.0229434,
0.0229819, 0.0230203, 0.0230586, 0.0230968, 0.0231348, 0.0231727, 0.0232105,
0.0232482, 0.0232858, 0.0233233, 0.0233606, 0.0233979, 0.023435, 0.023472, 0.0235089,
0.0235457, 0.0235824, 0.023619, 0.0236554, 0.0236918, 0.023728, 0.0237642, 0.0238002,
0.0238361, 0.0238719, 0.0239076, 0.0239432, 0.0239787, 0.0240141, 0.0240493,
0.0240845, 0.0241196, 0.0241545, 0.0241893, 0.0242241, 0.0242587, 0.0242932,
0.0243276, 0.0243619, 0.0243961, 0.0244303, 0.0244643, 0.0244982, 0.024532,
0.0245656, 0.0245992, 0.0246327, 0.0246661, 0.0246994, 0.0247326, 0.0247657,
0.0247986, 0.0248315, 0.0248643, 0.024897, 0.0249295, 0.024962, 0.0249944, 0.0250267,
0.0250588, 0.0250909, 0.0251229, 0.0251548, 0.0251866, 0.0252182, 0.0252498,
0.0252813, 0.0253127, 0.025344, 0.0253752, 0.0254063, 0.0254373, 0.0254682, 0.025499,
0.0255297, 0.0255603, 0.0255909, 0.0256213, 0.0256516, 0.0256819, 0.025712,
0.0257421, 0.025772, 0.0258019, 0.0258316, 0.0258613, 0.0258909, 0.0259204,
0.0259498, 0.0259791, 0.0260083, 0.0260374, 0.0260664, 0.0260954, 0.0261242,
0.026153, 0.0261816, 0.0262102, 0.0262387, 0.0262671, 0.0262954, 0.0263236,
0.0263517, 0.0263797, 0.0264077, 0.0264355, 0.0264633, 0.026491, 0.0265185, 0.026546,
0.0265735, 0.0266008, 0.026628, 0.0266552, 0.0266822, 0.0267092, 0.0267361,
0.0267629, 0.0267896, 0.0268162, 0.0268427, 0.0268692, 0.0268956, 0.0269218,
0.026948, 0.0269742, 0.0270002, 0.0270261, 0.027052, 0.0270778, 0.0271035, 0.0271291,
0.0271546, 0.0271801, 0.0272054, 0.0272307, 0.0272559, 0.027281, 0.0273061,
0.027331, 0.0273559, 0.0273807, 0.0274054, 0.02743, 0.0274546, 0.027479, 0.0275034,

0.0275277, 0.0275519, 0.0275761, 0.0276001, 0.0276241, 0.027648, 0.0276718,
0.0276956, 0.0277192, 0.0277428, 0.0277663, 0.0277897, 0.0278131, 0.0278363,
0.0278595, 0.0278827, 0.0279057, 0.0279287, 0.0279515, 0.0279743, 0.0279971,
0.0280197, 0.0280423, 0.0280648, 0.0280873, 0.0281096, 0.0281319, 0.0281541,
0.0281762, 0.0281983, 0.0282203, 0.0282422, 0.028264, 0.0282858, 0.0283075,
0.0283291, 0.0283506, 0.0283721, 0.0283935, 0.0284148, 0.028436, 0.0284572,
0.0284783, 0.0284993, 0.0285203, 0.0285412, 0.028562, 0.0285828, 0.0286034,
0.028624, 0.0286446, 0.028665, 0.0286854, 0.0287057, 0.028726, 0.0287462, 0.0287663,
0.0287863, 0.0288063, 0.0288262, 0.028846, 0.0288658, 0.0288855, 0.0289051,
0.0289246, 0.0289441, 0.0289636, 0.0289829, 0.0290022, 0.0290214, 0.0290406,
0.0290597, 0.0290787, 0.0290977, 0.0291166, 0.0291354, 0.0291542, 0.0291729,
0.0291915, 0.02921, 0.0292286, 0.029247, 0.0292654, 0.0292837, 0.0293019, 0.0293201,
0.0293382, 0.0293563, 0.0293742, 0.0293922, 0.02941, 0.0294278, 0.0294456, 0.0294632,
0.0294809, 0.0294984, 0.0295159, 0.0295333, 0.0295507, 0.029568, 0.0295852,
0.0296024, 0.0296195, 0.0296366, 0.0296535, 0.0296705, 0.0296874, 0.0297042,
0.0297209, 0.0297376, 0.0297543, 0.0297708, 0.0297873, 0.0298038, 0.0298202,
0.0298365, 0.0298528, 0.0298691, 0.0298852, 0.0299013, 0.0299174, 0.0299334,
0.0299493, 0.0299652, 0.029981, 0.0299968, 0.0300125, 0.0300282, 0.0300438,
0.0300593, 0.0300748, 0.0300902, 0.0301056, 0.0301209, 0.0301362, 0.0301514,
0.0301666, 0.0301817, 0.0301967, 0.0302117, 0.0302266, 0.0302415, 0.0302564,
0.0302711, 0.0302859, 0.0303005, 0.0303151, 0.0303297, 0.0303442, 0.0303587,
0.0303731, 0.0303874, 0.0304017, 0.030416, 0.0304302, 0.0304443, 0.0304584,
0.0304724, 0.0304864, 0.0305004, 0.0305143, 0.0305281, 0.0305419, 0.0305556,
0.0305693, 0.0305829, 0.0305965, 0.0306101, 0.0306236, 0.030637, 0.0306504,
0.0306638, 0.0306771, 0.0306903, 0.0307035, 0.0307167, 0.0307298, 0.0307428,
0.0307558, 0.0307688, 0.0307817, 0.0307946, 0.0308074, 0.0308201, 0.0308329,
0.0308455, 0.0308582, 0.0308707, 0.0308833, 0.0308958, 0.0309082, 0.0309206,
0.030933, 0.0309453, 0.0309575, 0.0309697, 0.0309819, 0.030994, 0.0310061, 0.0310181,
0.0310301, 0.0310421, 0.031054, 0.0310658, 0.0310777, 0.0310894, 0.0311012,
0.0311128, 0.0311245, 0.0311361, 0.0311476, 0.0311592, 0.0311706, 0.0311821,
0.0311935, 0.0312048, 0.0312161, 0.0312274, 0.0312386, 0.0312498, 0.0312609,
0.031272, 0.031283, 0.0312941, 0.031305, 0.031316, 0.0313269, 0.0313377, 0.0313485,
0.0313593, 0.03137, 0.0313807, 0.0313914, 0.031402, 0.0314125, 0.0314231, 0.0314336,
0.031444, 0.0314544, 0.0314648, 0.0314751, 0.0314854, 0.0314957, 0.0315059,
0.0315161, 0.0315262, 0.0315363, 0.0315464, 0.0315564, 0.0315664, 0.0315764,
0.0315863, 0.0315962, 0.031606, 0.0316159, 0.0316256, 0.0316354, 0.0316451,
0.0316547, 0.0316644, 0.0316739, 0.0316835, 0.031693, 0.0317025, 0.031712, 0.0317214,
0.0317307, 0.0317401, 0.0317494, 0.0317587, 0.0317679, 0.0317771, 0.0317863,
0.0317954, 0.0318045, 0.0318136, 0.0318226, 0.0318316, 0.0318406, 0.0318495,
0.0318584, 0.0318673, 0.0318761, 0.0318849, 0.0318937, 0.0319024, 0.0319111,
0.0319198, 0.0319284, 0.031937, 0.0319456, 0.0319541, 0.0319627, 0.0319711,
0.0319796, 0.031988, 0.0319964, 0.0320047, 0.032013, 0.0320213, 0.0320296,
0.0320378, 0.032046, 0.0320542, 0.0320623, 0.0320704, 0.0320785, 0.0320865,
0.0320945, 0.0321025, 0.0321104, 0.0321184, 0.0321262, 0.0321341, 0.0321419,
0.0321497, 0.0321575, 0.0321653, 0.032173, 0.0321807, 0.0321883, 0.0321959,
0.0322035, 0.0322111, 0.0322186, 0.0322262, 0.0322336, 0.0322411, 0.0322485,
0.0322559, 0.0322633, 0.0322707, 0.032278, 0.0322853, 0.0322925, 0.0322998,
0.032307, 0.0323142, 0.0323213, 0.0323284, 0.0323355, 0.0323426, 0.0323497,
0.0323567, 0.0323637, 0.0323706, 0.0323776, 0.0323845, 0.0323914, 0.0323983,
0.0324051, 0.0324119, 0.0324187, 0.0324255, 0.0324322, 0.0324389, 0.0324456,
0.0324523, 0.0324589, 0.0324655, 0.0324721, 0.0324787, 0.0324852, 0.0324917,
0.0324982, 0.0325047, 0.0325111, 0.0325175, 0.0325239, 0.0325303, 0.0325366,
0.0325429, 0.0325492, 0.0325555, 0.0325617, 0.032568, 0.0325742, 0.0325803,
0.0325865, 0.0325926, 0.0325987, 0.0326048, 0.0326109, 0.0326169, 0.0326229,

```
0.0326289, 0.0326349, 0.0326409, 0.0326468, 0.0326527, 0.0326586, 0.0326644,
0.0326703, 0.0326761, 0.0326819, 0.0326877, 0.0326934, 0.0326991, 0.0327048,
0.0327105, 0.0327162, 0.0327218, 0.0327275, 0.0327331, 0.0327386, 0.0327442,
0.0327497, 0.0327553, 0.0327608, 0.0327662, 0.0327717, 0.0327771, 0.0327825,
0.0327879, 0.0327933, 0.0327987, 0.032804, 0.0328093, 0.0328146, 0.0328199,
0.0328251, 0.0328304, 0.0328356, 0.0328408, 0.032846, 0.0328511, 0.0328563,
0.0328614, 0.0328665, 0.0328716, 0.0328766, 0.0328817, 0.0328867, 0.0328917,
0.0328967, 0.0329017, 0.0329066, 0.0329115, 0.0329165, 0.0329214, 0.0329262,
0.0329311, 0.0329359, 0.0329407, 0.0329456, 0.0329503, 0.0329551, 0.0329599,
0.0329646, 0.0329693, 0.032974, 0.0329787, 0.0329833, 0.032988, 0.0329926, 0.0329972,
0.0330018, 0.0330064, 0.0330109, 0.0330155, 0.03302, 0.0330245, 0.033029, 0.0330335,
0.0330379, 0.0330424, 0.0330468, 0.0330512, 0.0330556, 0.0330599, 0.0330643,
0.0330686, 0.033073, 0.0330773, 0.0330816, 0.0330858, 0.0330901, 0.0330943}
```

```
ListPlot[ Transpose[{time, meancurve100}], PlotJoined → True,
 PlotStyle → AbsoluteThickness[2], Frame → True, Axes → False ]
```



- Graphics -

Sample, setting alpha0=10000

```
alpha0 = 10000
```

10000

1/18/2010

```
bigresult = Table[
eps = 0.0;
tauL = taudist[[i]];
   R = Rtable[[i]];
   Rc = Rctable[[i]];
   lambda = 2.88 10^-5 ;
   cc = cctable[[i]];
   fac = 590 / 2500 * 0.12 / (1 - 0.12);
   cc = cc * fac;
   a = cc * (R - 1);
   alphaf = alpha0 * cc / fac ;

   {X1, Y1} =
    SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
   Table[ X1[t, tauL] + Y1[t, tauL] ,
    {t, 0, 100000, 100}],
   {i, 1, nreal}];
```

```
time = (Range[1001] - 1) * 100000. / 1000.;
```

```
meancurve1000 = Apply[Plus, bigresult] / Length[bigresult]
```

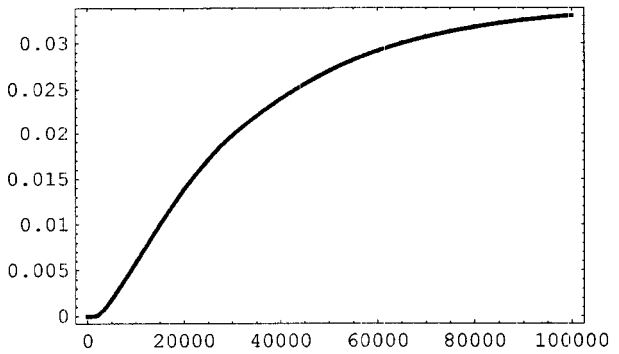
{0., 5.38101×10^{-18}, 1.28351×10^{-17}, 1.89172×10^{-16}, 1.66754×10^{-13}, 1.55799×10^{-11}, 3.60239×10^{-10}, 3.76383×10^{-9}, 2.40163×10^{-8}, 1.07239×10^{-7}, 3.63524×10^{-7}, 9.91203×10^{-7}, 2.29132×10^{-6}, 4.65439×10^{-6}, 8.52391×10^{-6}, 0.0000143611, 0.000022617, 0.0000336822, 0.0000478679, 0.0000654013, 0.0000864282, 0.000111015, 0.000139154, 0.000170794, 0.000205808, 0.000244077, 0.000285413, 0.000329653, 0.000376618, 0.000426118, 0.000477958, 0.000531982, 0.000587991, 0.000645857, 0.000705408, 0.000766527, 0.000829078, 0.000892952, 0.000958042, 0.00102426, 0.00109152, 0.00115973, 0.00122882, 0.00129875, 0.00136942, 0.0014408, 0.00151286, 0.00158551, 0.00165872, 0.0017325, 0.00180675, 0.00188148, 0.00195666, 0.00203224, 0.00210819, 0.00218453, 0.00226122, 0.00233822, 0.00241551, 0.00249311, 0.00257098, 0.0026491, 0.00272747, 0.00280606, 0.00288487, 0.0029639, 0.00304312, 0.00312253, 0.00320213, 0.00328189, 0.00336182, 0.0034419, 0.00352215, 0.00360255, 0.00368309, 0.00376378, 0.0038446, 0.00392554, 0.00400662, 0.00408783, 0.00416917, 0.00425063, 0.00433219, 0.00441387, 0.00449567, 0.00457758, 0.00465959, 0.00474171, 0.00482392, 0.00490623, 0.00498864, 0.00507116, 0.00515377, 0.00523647, 0.00531926, 0.00540211, 0.00548506, 0.00556808, 0.00565119, 0.00573439, 0.00581767, 0.00590101, 0.0059844, 0.00606785, 0.00615135, 0.00623493, 0.00631857, 0.00640227, 0.00648601, 0.00656979, 0.00665361, 0.00673746, 0.00682135, 0.00690526, 0.00698919, 0.00707315, 0.00715713, 0.00724111, 0.00732511, 0.0074091, 0.00749309, 0.00757707, 0.00766103, 0.00774497, 0.0078289, 0.0079128,

0.00799666, 0.00808049, 0.00816429, 0.00824803, 0.00833172, 0.00841534, 0.0084989,
0.0085824, 0.00866583, 0.00874919, 0.00883247, 0.00891567, 0.00899878, 0.00908179,
0.00916469, 0.00924749, 0.00933018, 0.00941277, 0.00949524, 0.00957759, 0.00965983,
0.00974193, 0.0098239, 0.00990572, 0.0099874, 0.0100689, 0.0101503, 0.0102315,
0.0103126, 0.0103935, 0.0104743, 0.0105549, 0.0106353, 0.0107156, 0.0107956,
0.0108755, 0.0109552, 0.0110347, 0.011114, 0.0111931, 0.0112721, 0.0113508,
0.0114293, 0.0115076, 0.0115857, 0.0116636, 0.0117413, 0.0118188, 0.0118961,
0.0119731, 0.0120499, 0.0121265, 0.0122029, 0.012279, 0.012355, 0.0124306, 0.0125061,
0.0125813, 0.0126563, 0.0127311, 0.0128056, 0.0128799, 0.0129539, 0.0130277,
0.0131012, 0.0131746, 0.0132476, 0.0133204, 0.013393, 0.0134653, 0.0135374,
0.0136092, 0.0136808, 0.0137522, 0.0138233, 0.0138941, 0.0139647, 0.014035,
0.0141051, 0.0141749, 0.0142445, 0.0143138, 0.0143829, 0.0144518, 0.0145203,
0.0145887, 0.0146567, 0.0147246, 0.0147921, 0.0148595, 0.0149265, 0.0149933,
0.0150599, 0.0151263, 0.0151923, 0.0152582, 0.0153237, 0.0153891, 0.0154542,
0.015519, 0.0155836, 0.015648, 0.0157121, 0.015776, 0.0158396, 0.015903, 0.0159661,
0.016029, 0.0160917, 0.0161541, 0.0162163, 0.0162783, 0.01634, 0.0164015, 0.0164627,
0.0165237, 0.0165845, 0.016645, 0.0167054, 0.0167655, 0.0168253, 0.016885, 0.0169444,
0.0170035, 0.0170625, 0.0171212, 0.0171798, 0.017238, 0.0172961, 0.017354, 0.0174116,
0.017469, 0.0175262, 0.0175832, 0.0176399, 0.0176965, 0.0177528, 0.017809, 0.0178649,
0.0179206, 0.0179761, 0.0180314, 0.0180864, 0.0181413, 0.018196, 0.0182505,
0.0183048, 0.0183588, 0.0184127, 0.0184664, 0.0185199, 0.0185731, 0.0186262,
0.0186791, 0.0187318, 0.0187843, 0.0188366, 0.0188887, 0.0189406, 0.0189924,
0.0190439, 0.0190953, 0.0191465, 0.0191975, 0.0192483, 0.0192989, 0.0193494,
0.0193996, 0.0194497, 0.0194997, 0.0195494, 0.019599, 0.0196483, 0.0196975,
0.0197466, 0.0197954, 0.0198441, 0.0198927, 0.019941, 0.0199892, 0.0200372, 0.020085,
0.0201327, 0.0201802, 0.0202276, 0.0202748, 0.0203218, 0.0203687, 0.0204154,
0.0204619, 0.0205083, 0.0205545, 0.0206006, 0.0206465, 0.0206922, 0.0207378,
0.0207833, 0.0208286, 0.0208737, 0.0209187, 0.0209635, 0.0210082, 0.0210528,
0.0210971, 0.0211414, 0.0211855, 0.0212294, 0.0212732, 0.0213169, 0.0213604,
0.0214037, 0.0214469, 0.02149, 0.0215329, 0.0215757, 0.0216184, 0.0216609, 0.0217033,
0.0217455, 0.0217876, 0.0218296, 0.0218714, 0.0219131, 0.0219546, 0.0219961,
0.0220374, 0.0220785, 0.0221195, 0.0221604, 0.0222011, 0.0222418, 0.0222822,
0.0223226, 0.0223628, 0.0224029, 0.0224429, 0.0224827, 0.0225224, 0.022562,
0.0226015, 0.0226408, 0.0226801, 0.0227192, 0.0227581, 0.022797, 0.0228357,
0.0228743, 0.0229128, 0.0229511, 0.0229893, 0.0230275, 0.0230654, 0.0231033,
0.0231411, 0.0231787, 0.0232162, 0.0232536, 0.0232909, 0.023328, 0.0233651, 0.023402,
0.0234388, 0.0234755, 0.0235121, 0.0235485, 0.0235849, 0.0236212, 0.0236573,
0.0236933, 0.0237292, 0.023765, 0.0238007, 0.0238363, 0.0238717, 0.0239071,
0.0239423, 0.0239774, 0.0240124, 0.0240473, 0.0240821, 0.0241168, 0.0241514,
0.0241859, 0.0242203, 0.0242545, 0.0242887, 0.0243227, 0.0243567, 0.0243905,
0.0244242, 0.0244579, 0.0244914, 0.0245248, 0.0245581, 0.0245914, 0.0246245,
0.0246575, 0.0246904, 0.0247232, 0.0247559, 0.0247885, 0.024821, 0.0248534,
0.0248857, 0.0249179, 0.02495, 0.024982, 0.0250139, 0.0250457, 0.0250774, 0.025109,
0.0251405, 0.0251719, 0.0252032, 0.0252344, 0.0252655, 0.0252965, 0.0253274,
0.0253582, 0.025389, 0.0254196, 0.0254501, 0.0254806, 0.0255109, 0.0255412,
0.0255713, 0.0256014, 0.0256313, 0.0256612, 0.025691, 0.0257207, 0.0257503,
0.0257798, 0.0258092, 0.0258385, 0.0258677, 0.0258968, 0.0259258, 0.0259548,
0.0259836, 0.0260124, 0.0260411, 0.0260697, 0.0260981, 0.0261265, 0.0261549,
0.0261831, 0.0262112, 0.0262393, 0.0262672, 0.0262951, 0.0263229, 0.0263506,
0.0263782, 0.0264057, 0.0264331, 0.0264605, 0.0264877, 0.0265149, 0.026542, 0.026569,
0.0265959, 0.0266227, 0.0266495, 0.0266761, 0.0267027, 0.0267292, 0.0267555,
0.0267819, 0.0268081, 0.0268342, 0.0268603, 0.0268863, 0.0269122, 0.026938,
0.0269637, 0.0269893, 0.0270149, 0.0270404, 0.0270658, 0.0270911, 0.0271163,
0.0271415, 0.0271666, 0.0271915, 0.0272165, 0.0272413, 0.027266, 0.0272907,

0.0273153, 0.0273398, 0.0273642, 0.0273886, 0.0274129, 0.0274371, 0.0274612,
0.0274852, 0.0275092, 0.027533, 0.0275569, 0.0275806, 0.0276042, 0.0276278,
0.0276513, 0.0276747, 0.027698, 0.0277213, 0.0277445, 0.0277676, 0.0277906,
0.0278136, 0.0278364, 0.0278592, 0.027882, 0.0279046, 0.0279272, 0.0279497,
0.0279721, 0.0279945, 0.0280168, 0.028039, 0.0280611, 0.0280832, 0.0281052,
0.0281271, 0.0281489, 0.0281707, 0.0281924, 0.028214, 0.0282356, 0.0282571,
0.0282785, 0.0282998, 0.0283211, 0.0283423, 0.0283634, 0.0283845, 0.0284055,
0.0284264, 0.0284472, 0.028468, 0.0284887, 0.0285093, 0.0285299, 0.0285504,
0.0285708, 0.0285912, 0.0286115, 0.0286317, 0.0286519, 0.0286719, 0.028692,
0.0287119, 0.0287318, 0.0287516, 0.0287714, 0.028791, 0.0288107, 0.0288302,
0.0288497, 0.0288691, 0.0288885, 0.0289077, 0.028927, 0.0289461, 0.0289652,
0.0289842, 0.0290032, 0.0290221, 0.0290409, 0.0290597, 0.0290784, 0.029097,
0.0291156, 0.0291341, 0.0291525, 0.0291709, 0.0291892, 0.0292075, 0.0292257,
0.0292438, 0.0292619, 0.0292799, 0.0292978, 0.0293157, 0.0293335, 0.0293513,
0.029369, 0.0293866, 0.0294042, 0.0294217, 0.0294391, 0.0294565, 0.0294739,
0.0294911, 0.0295083, 0.0295255, 0.0295426, 0.0295596, 0.0295766, 0.0295935,
0.0296103, 0.0296271, 0.0296438, 0.0296605, 0.0296771, 0.0296937, 0.0297102,
0.0297267, 0.029743, 0.0297594, 0.0297756, 0.0297919, 0.029808, 0.0298241,
0.0298402, 0.0298562, 0.0298721, 0.029888, 0.0299038, 0.0299196, 0.0299353,
0.0299509, 0.0299665, 0.0299821, 0.0299976, 0.030013, 0.0300284, 0.0300437,
0.030059, 0.0300742, 0.0300894, 0.0301045, 0.0301195, 0.0301345, 0.0301495,
0.0301644, 0.0301792, 0.030194, 0.0302088, 0.0302234, 0.0302381, 0.0302527,
0.0302672, 0.0302817, 0.0302961, 0.0303105, 0.0303248, 0.0303391, 0.0303533,
0.0303675, 0.0303816, 0.0303956, 0.0304097, 0.0304236, 0.0304376, 0.0304514,
0.0304653, 0.030479, 0.0304928, 0.0305064, 0.03052, 0.0305336, 0.0305472, 0.0305606,
0.0305741, 0.0305874, 0.0306008, 0.0306141, 0.0306273, 0.0306405, 0.0306536,
0.0306667, 0.0306798, 0.0306928, 0.0307057, 0.0307186, 0.0307315, 0.0307443,
0.0307571, 0.0307698, 0.0307825, 0.0307951, 0.0308077, 0.0308202, 0.0308327,
0.0308452, 0.0308576, 0.0308699, 0.0308822, 0.0308945, 0.0309067, 0.0309189,
0.030931, 0.0309431, 0.0309552, 0.0309672, 0.0309792, 0.0309911, 0.0310029,
0.0310148, 0.0310266, 0.0310383, 0.03105, 0.0310617, 0.0310733, 0.0310849,
0.0310964, 0.0311079, 0.0311194, 0.0311308, 0.0311421, 0.0311535, 0.0311648,
0.031176, 0.0311872, 0.0311984, 0.0312095, 0.0312206, 0.0312316, 0.0312426,
0.0312536, 0.0312645, 0.0312754, 0.0312862, 0.031297, 0.0313078, 0.0313185,
0.0313292, 0.0313398, 0.0313504, 0.031361, 0.0313715, 0.031382, 0.0313924,
0.0314029, 0.0314132, 0.0314236, 0.0314339, 0.0314441, 0.0314543, 0.0314645,
0.0314747, 0.0314848, 0.0314949, 0.0315049, 0.0315149, 0.0315249, 0.0315348,
0.0315447, 0.0315545, 0.0315643, 0.0315741, 0.0315839, 0.0315936, 0.0316033,
0.0316129, 0.0316225, 0.0316321, 0.0316416, 0.0316511, 0.0316606, 0.03167,
0.0316794, 0.0316887, 0.0316981, 0.0317074, 0.0317166, 0.0317258, 0.031735,
0.0317442, 0.0317533, 0.0317624, 0.0317714, 0.0317805, 0.0317894, 0.0317984,
0.0318073, 0.0318162, 0.0318251, 0.0318339, 0.0318427, 0.0318514, 0.0318601,
0.0318688, 0.0318775, 0.0318861, 0.0318947, 0.0319033, 0.0319118, 0.0319203,
0.0319288, 0.0319372, 0.0319456, 0.031954, 0.0319623, 0.0319707, 0.0319789,
0.0319872, 0.0319954, 0.0320036, 0.0320118, 0.0320199, 0.032028, 0.0320361,
0.0320441, 0.0320521, 0.0320601, 0.0320681, 0.032076, 0.0320839, 0.0320917,
0.0320996, 0.0321074, 0.0321152, 0.0321229, 0.0321306, 0.0321383, 0.032146,
0.0321536, 0.0321612, 0.0321688, 0.0321764, 0.0321839, 0.0321914, 0.0321988,
0.0322063, 0.0322137, 0.0322211, 0.0322284, 0.0322357, 0.032243, 0.0322503,
0.0322576, 0.0322648, 0.032272, 0.0322791, 0.0322863, 0.0322934, 0.0323005,
0.0323075, 0.0323146, 0.0323216, 0.0323286, 0.0323355, 0.0323424, 0.0323494,
0.0323562, 0.0323631, 0.0323699, 0.0323767, 0.0323835, 0.0323902, 0.032397,
0.0324037, 0.0324104, 0.032417, 0.0324236, 0.0324302, 0.0324368, 0.0324434,
0.0324499, 0.0324564, 0.0324629, 0.0324694, 0.0324758, 0.0324822, 0.0324886,

```
0.032495, 0.0325013, 0.0325076, 0.0325139, 0.0325202, 0.0325264, 0.0325327,
0.0325389, 0.032545, 0.0325512, 0.0325573, 0.0325634, 0.0325695, 0.0325756,
0.0325816, 0.0325876, 0.0325936, 0.0325996, 0.0326056, 0.0326115, 0.0326174,
0.0326233, 0.0326291, 0.032635, 0.0326408, 0.0326466, 0.0326524, 0.0326581,
0.0326638, 0.0326696, 0.0326752, 0.0326809, 0.0326866, 0.0326922, 0.0326978,
0.0327034, 0.032709, 0.0327145, 0.03272, 0.0327255, 0.032731, 0.0327365, 0.0327419,
0.0327473, 0.0327527, 0.0327581, 0.0327635, 0.0327688, 0.0327741, 0.0327795,
0.0327847, 0.03279, 0.0327952, 0.0328005, 0.0328057, 0.0328109, 0.032816, 0.0328212,
0.0328263, 0.0328314, 0.0328365, 0.0328416, 0.0328467, 0.0328517, 0.0328567,
0.0328617, 0.0328667, 0.0328717, 0.0328766, 0.0328815, 0.0328864, 0.0328913,
0.0328962, 0.032901, 0.0329059, 0.0329107, 0.0329155, 0.0329203, 0.032925,
0.0329298, 0.0329345, 0.0329392, 0.0329439, 0.0329486, 0.0329532, 0.0329579,
0.0329625, 0.0329671, 0.0329717, 0.0329762, 0.0329808, 0.0329853, 0.0329898,
0.0329944, 0.0329988, 0.0330033, 0.0330078, 0.0330122, 0.0330166, 0.033021,
0.0330254, 0.0330298, 0.0330341, 0.0330385, 0.0330428, 0.0330471, 0.0330514}
```

```
ListPlot[ Transpose[{time, meancurve1000}], PlotJoined → True,
 PlotStyle → AbsoluteThickness[2], Frame → True, Axes → False ]
```



- Graphics -

Sample, setting alpha0=0

```
alpha0 = 0
```

0

RW 1/18/2010

```
bigresult = Table[
eps = 0.0;
tauL = taudist[[i]];
   R = Rtable[[i]];
   Rc = Rctable[[i]];
   lambda = 2.88 10^-5 ;
   cc = cctable[[i]];
   fac = 590 / 2500 * 0.12 / (1 - 0.12);
   cc = cc * fac;
   a = cc * (R - 1);
   alphaf = alpha0 * cc / fac ;

   {X1, Y1} =
    SolveIt[alphaf, R, Rc, eps, a, lambda, tauL];
   Table[ X1[t, tauL] + Y1[t, tauL] ,
    {t, 0, 100000, 100}],
   {i, 1, nreal}];
```

```
time = (Range[1001] - 1) * 100000. / 1000.;

meancurve00 = Apply[Plus, bigresult] / Length[bigresult]
```

General::spell1 :
Possible spelling error: new symbol name "meancurve00" is similar to existing symbol "meancurve100". More...

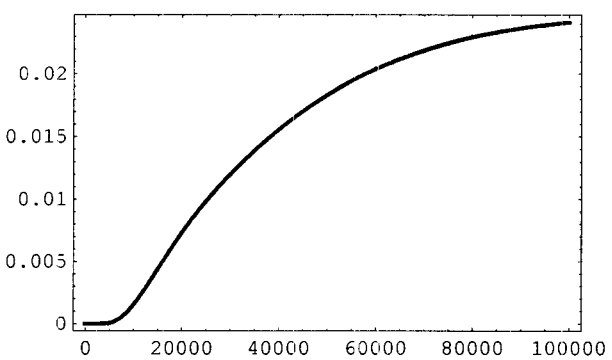
{0., 2.16317×10^{-62}, 3.69151×10^{-42}, 2.86848×10^{-33}, 1.07009×10^{-27}, 5.00366×10^{-24}, 2.44463×10^{-21}, 1.96091×10^{-19}, 6.04065×10^{-18}, 1.21324×10^{-16}, 2.53182×10^{-15}, 3.04709×10^{-14}, 2.33113×10^{-13}, 1.3213×10^{-12}, 6.03161×10^{-12}, 2.57371×10^{-11}, 8.72639×10^{-11}, 2.61102×10^{-10}, 7.06988×10^{-10}, 1.70582×10^{-9}, 3.81686×10^{-9}, 7.98163×10^{-9}, 1.58274×10^{-8}, 2.96133×10^{-8}, 5.26284×10^{-8}, 8.96663×10^{-8}, 1.46547×10^{-7}, 2.31998×10^{-7}, 3.55237×10^{-7}, 5.29247×10^{-7}, 7.68281×10^{-7}, 1.08953×10^{-6}, 1.51242×10^{-6}, 2.05897×10^{-6}, 2.75422×10^{-6}, 3.62313×10^{-6}, 4.69495×10^{-6}, 6.00163×10^{-6}, 7.57214×10^{-6}, 9.44035×10^{-6}, 0.0000116431, 0.0000142122, 0.000017184, 0.000020595, 0.0000244774, 0.0000288702, 0.0000338047, 0.0000393119, 0.0000454286, 0.0000521864, 0.0000596146, 0.0000677402, 0.0000765907, 0.0000861938, 0.0000965747, 0.000107753, 0.000119746, 0.000132575, 0.000146262, 0.000160817, 0.000176254, 0.000192581, 0.000209815, 0.000227966, 0.000247035, 0.00026703, 0.000287959, 0.000309822, 0.000332619, 0.000356345, 0.000381004, 0.000406599, 0.000433124, 0.000460571, 0.000488932, 0.0005182, 0.000548379, 0.000579454, 0.000611414, 0.000644254, 0.000677959, 0.000712526, 0.000747939, 0.000784179, 0.000821237, 0.000859107, 0.000897778, 0.000937232, 0.00097745, 0.00101841, 0.00106011, 0.00110254, 0.00114569, 0.00118954, 0.00123406,

0.00127924, 0.00132505, 0.00137151, 0.0014186, 0.00146629, 0.00151457, 0.00156341,
0.0016128, 0.00166273, 0.00171319, 0.00176417, 0.00181564, 0.00186759, 0.00192,
0.00197286, 0.00202615, 0.00207987, 0.00213399, 0.0021885, 0.00224339, 0.00229864,
0.00235424, 0.00241017, 0.00246643, 0.002523, 0.00257986, 0.00263701, 0.00269442,
0.0027521, 0.00281, 0.00286814, 0.00292651, 0.00298508, 0.00304386, 0.00310283,
0.00316197, 0.00322127, 0.00328071, 0.0033403, 0.00340001, 0.00345985, 0.00351982,
0.00357989, 0.00364006, 0.00370031, 0.00376063, 0.003821, 0.00388143, 0.00394191,
0.00400243, 0.00406299, 0.00412359, 0.0041842, 0.00424482, 0.00430544, 0.00436605,
0.00442665, 0.00448722, 0.00454778, 0.0046083, 0.00466879, 0.00472925, 0.00478965,
0.00485, 0.00491028, 0.00497051, 0.00503066, 0.00509074, 0.00515074, 0.00521066,
0.00527049, 0.00533023, 0.00538987, 0.0054494, 0.00550884, 0.00556816, 0.00562738,
0.00568648, 0.00574547, 0.00580433, 0.00586308, 0.0059217, 0.00598019, 0.00603854,
0.00609676, 0.00615485, 0.00621279, 0.00627059, 0.00632825, 0.00638576, 0.00644312,
0.00650033, 0.0065574, 0.0066143, 0.00667106, 0.00672766, 0.00678411, 0.00684039,
0.00689652, 0.00695248, 0.00700828, 0.00706391, 0.00711938, 0.00717469, 0.00722983,
0.0072848, 0.0073396, 0.00739425, 0.00744872, 0.00750303, 0.00755717, 0.00761113,
0.00766493, 0.00771855, 0.00777201, 0.00782529, 0.00787841, 0.00793136, 0.00798414,
0.00803674, 0.00808918, 0.00814145, 0.00819355, 0.00824548, 0.00829725, 0.00834884,
0.00840026, 0.00845152, 0.00850261, 0.00855353, 0.00860428, 0.00865487, 0.0087053,
0.00875555, 0.00880565, 0.00885558, 0.00890534, 0.00895495, 0.0090044, 0.00905368,
0.0091028, 0.00915177, 0.00920058, 0.00924922, 0.00929771, 0.00934604, 0.00939421,
0.00944222, 0.00949008, 0.00953778, 0.00958533, 0.00963273, 0.00967998, 0.00972708,
0.00977403, 0.00982083, 0.00986748, 0.00991399, 0.00996035, 0.0100066, 0.0100526,
0.0100986, 0.0101443, 0.01019, 0.0102355, 0.0102808, 0.010326, 0.0103711, 0.010416,
0.0104608, 0.0105055, 0.01055, 0.0105944, 0.0106386, 0.0106827, 0.0107267, 0.0107706,
0.0108143, 0.0108579, 0.0109014, 0.0109447, 0.0109879, 0.011031, 0.0110739,
0.0111168, 0.0111595, 0.011202, 0.0112445, 0.0112868, 0.011329, 0.0113711, 0.0114131,
0.0114549, 0.0114967, 0.0115383, 0.0115798, 0.0116212, 0.0116624, 0.0117036,
0.0117446, 0.0117855, 0.0118263, 0.011867, 0.0119076, 0.0119481, 0.0119884,
0.0120287, 0.0120688, 0.0121088, 0.0121488, 0.0121886, 0.0122283, 0.0122679,
0.0123074, 0.0123468, 0.012386, 0.0124252, 0.0124643, 0.0125033, 0.0125421,
0.0125809, 0.0126195, 0.0126581, 0.0126966, 0.0127349, 0.0127732, 0.0128114,
0.0128494, 0.0128874, 0.0129253, 0.012963, 0.0130007, 0.0130383, 0.0130758,
0.0131131, 0.0131504, 0.0131876, 0.0132247, 0.0132617, 0.0132987, 0.0133355,
0.0133722, 0.0134088, 0.0134454, 0.0134818, 0.0135182, 0.0135545, 0.0135906,
0.0136267, 0.0136627, 0.0136986, 0.0137345, 0.0137702, 0.0138059, 0.0138414,
0.0138769, 0.0139123, 0.0139476, 0.0139828, 0.0140179, 0.014053, 0.0140879,
0.0141228, 0.0141576, 0.0141922, 0.0142269, 0.0142614, 0.0142958, 0.0143302,
0.0143644, 0.0143986, 0.0144327, 0.0144667, 0.0145007, 0.0145345, 0.0145683,
0.014602, 0.0146356, 0.0146692, 0.0147026, 0.014736, 0.0147693, 0.0148025, 0.0148356,
0.0148686, 0.0149016, 0.0149345, 0.0149672, 0.015, 0.0150326, 0.0150651, 0.0150976,
0.01513, 0.0151623, 0.0151946, 0.0152267, 0.0152588, 0.0152908, 0.0153227, 0.0153546,
0.0153863, 0.015418, 0.0154496, 0.0154812, 0.0155126, 0.015544, 0.0155753, 0.0156065,
0.0156377, 0.0156687, 0.0156997, 0.0157306, 0.0157615, 0.0157922, 0.0158229,
0.0158535, 0.015884, 0.0159145, 0.0159449, 0.0159751, 0.0160054, 0.0160355,
0.0160656, 0.0160956, 0.0161255, 0.0161553, 0.0161851, 0.0162148, 0.0162444,
0.016274, 0.0163034, 0.0163328, 0.0163621, 0.0163914, 0.0164205, 0.0164496,
0.0164786, 0.0165076, 0.0165364, 0.0165652, 0.016594, 0.0166226, 0.0166512,
0.0166797, 0.0167081, 0.0167364, 0.0167647, 0.0167929, 0.016821, 0.0168491,
0.0168771, 0.016905, 0.0169328, 0.0169606, 0.0169883, 0.0170159, 0.0170434,
0.0170709, 0.0170983, 0.0171256, 0.0171529, 0.01718, 0.0172071, 0.0172342,
0.0172611, 0.017288, 0.0173148, 0.0173416, 0.0173682, 0.0173948, 0.0174214,
0.0174478, 0.0174742, 0.0175005, 0.0175267, 0.0175529, 0.017579, 0.017605, 0.017631,
0.0176569, 0.0176827, 0.0177084, 0.0177341, 0.0177597, 0.0177852, 0.0178107,

0.0178361, 0.0178614, 0.0178866, 0.0179118, 0.0179369, 0.0179619, 0.0179869,
0.0180118, 0.0180366, 0.0180614, 0.018086, 0.0181106, 0.0181352, 0.0181597,
0.0181841, 0.0182084, 0.0182327, 0.0182568, 0.018281, 0.018305, 0.018329, 0.0183529,
0.0183768, 0.0184006, 0.0184243, 0.0184479, 0.0184715, 0.018495, 0.0185185,
0.0185418, 0.0185652, 0.0185884, 0.0186116, 0.0186347, 0.0186577, 0.0186807,
0.0187036, 0.0187264, 0.0187491, 0.0187718, 0.0187945, 0.018817, 0.0188395,
0.0188619, 0.0188843, 0.0189066, 0.0189288, 0.018951, 0.0189731, 0.0189951,
0.019017, 0.0190389, 0.0190608, 0.0190825, 0.0191042, 0.0191259, 0.0191474,
0.0191689, 0.0191904, 0.0192117, 0.019233, 0.0192543, 0.0192755, 0.0192966,
0.0193176, 0.0193386, 0.0193595, 0.0193804, 0.0194012, 0.0194219, 0.0194426,
0.0194632, 0.0194837, 0.0195042, 0.0195246, 0.0195449, 0.0195652, 0.0195854,
0.0196055, 0.0196256, 0.0196456, 0.0196656, 0.0196855, 0.0197053, 0.0197251,
0.0197448, 0.0197645, 0.0197841, 0.0198036, 0.019823, 0.0198425, 0.0198618,
0.0198811, 0.0199003, 0.0199194, 0.0199385, 0.0199576, 0.0199766, 0.0199955,
0.0200143, 0.0200331, 0.0200518, 0.0200705, 0.0200891, 0.0201077, 0.0201262,
0.0201446, 0.020163, 0.0201813, 0.0201995, 0.0202177, 0.0202359, 0.0202539,
0.020272, 0.0202899, 0.0203078, 0.0203257, 0.0203434, 0.0203612, 0.0203788,
0.0203964, 0.020414, 0.0204315, 0.0204489, 0.0204663, 0.0204836, 0.0205009,
0.0205181, 0.0205352, 0.0205523, 0.0205693, 0.0205863, 0.0206032, 0.0206201,
0.0206369, 0.0206536, 0.0206703, 0.020687, 0.0207035, 0.0207201, 0.0207365,
0.020753, 0.0207693, 0.0207856, 0.0208019, 0.0208181, 0.0208342, 0.0208503,
0.0208663, 0.0208823, 0.0208982, 0.0209141, 0.0209299, 0.0209457, 0.0209614,
0.020977, 0.0209926, 0.0210082, 0.0210237, 0.0210391, 0.0210545, 0.0210698,
0.0210851, 0.0211004, 0.0211155, 0.0211307, 0.0211457, 0.0211607, 0.0211757,
0.0211906, 0.0212055, 0.0212203, 0.0212351, 0.0212498, 0.0212644, 0.021279,
0.0212936, 0.0213081, 0.0213226, 0.021337, 0.0213513, 0.0213657, 0.0213799,
0.0213941, 0.0214083, 0.0214224, 0.0214365, 0.0214505, 0.0214644, 0.0214784,
0.0214922, 0.021506, 0.0215198, 0.0215335, 0.0215472, 0.0215608, 0.0215744,
0.0215879, 0.0216014, 0.0216148, 0.0216282, 0.0216415, 0.0216548, 0.0216681,
0.0216813, 0.0216944, 0.0217075, 0.0217206, 0.0217336, 0.0217465, 0.0217594,
0.0217723, 0.0217851, 0.0217979, 0.0218106, 0.0218233, 0.021836, 0.0218486,
0.0218611, 0.0218736, 0.0218861, 0.0218985, 0.0219109, 0.0219232, 0.0219355,
0.0219477, 0.0219599, 0.0219721, 0.0219842, 0.0219963, 0.0220083, 0.0220203,
0.0220322, 0.0220441, 0.0220559, 0.0220678, 0.0220795, 0.0220912, 0.0221029,
0.0221146, 0.0221261, 0.0221377, 0.0221492, 0.0221607, 0.0221721, 0.0221835,
0.0221948, 0.0222061, 0.0222174, 0.0222286, 0.0222398, 0.0222509, 0.022262,
0.0222731, 0.0222841, 0.0222951, 0.022306, 0.0223169, 0.0223278, 0.0223386,
0.0223494, 0.0223601, 0.0223708, 0.0223815, 0.0223921, 0.0224027, 0.0224132,
0.0224237, 0.0224342, 0.0224446, 0.022455, 0.0224654, 0.0224757, 0.0224859,
0.0224962, 0.0225064, 0.0225165, 0.0225267, 0.0225367, 0.0225468, 0.0225568,
0.0225668, 0.0225767, 0.0225866, 0.0225965, 0.0226063, 0.0226161, 0.0226258,
0.0226356, 0.0226452, 0.0226549, 0.0226645, 0.0226741, 0.0226836, 0.0226931,
0.0227026, 0.022712, 0.0227214, 0.0227308, 0.0227401, 0.0227494, 0.0227587,
0.0227679, 0.0227771, 0.0227863, 0.0227954, 0.0228045, 0.0228135, 0.0228226,
0.0228315, 0.0228405, 0.0228494, 0.0228583, 0.0228672, 0.022876, 0.0228848,
0.0228935, 0.0229022, 0.0229109, 0.0229196, 0.0229282, 0.0229368, 0.0229454,
0.0229539, 0.0229624, 0.0229708, 0.0229793, 0.0229877, 0.022996, 0.0230044,
0.0230127, 0.0230209, 0.0230292, 0.0230374, 0.0230456, 0.0230537, 0.0230619,
0.0230699, 0.023078, 0.023086, 0.023094, 0.023102, 0.0231099, 0.0231178, 0.0231257,
0.0231336, 0.0231414, 0.0231492, 0.0231569, 0.0231647, 0.0231724, 0.02318,
0.0231877, 0.0231953, 0.0232029, 0.0232104, 0.023218, 0.0232255, 0.0232329,
0.0232404, 0.0232478, 0.0232552, 0.0232625, 0.0232699, 0.0232772, 0.0232844,
0.0232917, 0.0232989, 0.0233061, 0.0233133, 0.0233204, 0.0233275, 0.0233346,
0.0233416, 0.0233487, 0.0233557, 0.0233626, 0.0233696, 0.0233765, 0.0233834,

```
0.0233903, 0.0233971, 0.0234039, 0.0234107, 0.0234175, 0.0234242, 0.0234309,
0.0234376, 0.0234443, 0.0234509, 0.0234575, 0.0234641, 0.0234707, 0.0234772,
0.0234837, 0.0234902, 0.0234967, 0.0235031, 0.0235095, 0.0235159, 0.0235223,
0.0235286, 0.023535, 0.0235412, 0.0235475, 0.0235538, 0.02356, 0.0235662,
0.0235723, 0.0235785, 0.0235846, 0.0235907, 0.0235968, 0.0236028, 0.0236089,
0.0236149, 0.0236209, 0.0236268, 0.0236328, 0.0236387, 0.0236446, 0.0236504,
0.0236563, 0.0236621, 0.0236679, 0.0236737, 0.0236795, 0.0236852, 0.0236909,
0.0236966, 0.0237023, 0.0237079, 0.0237136, 0.0237192, 0.0237248, 0.0237303,
0.0237359, 0.0237414, 0.0237469, 0.0237524, 0.0237578, 0.0237633, 0.0237687,
0.0237741, 0.0237794, 0.0237848, 0.0237901, 0.0237954, 0.0238007, 0.023806,
0.0238113, 0.0238165, 0.0238217, 0.0238269, 0.0238321, 0.0238372, 0.0238423,
0.0238475, 0.0238526, 0.0238576, 0.0238627, 0.0238677, 0.0238727, 0.0238777,
0.0238827, 0.0238876, 0.0238926, 0.0238975, 0.0239024, 0.0239073, 0.0239121,
0.023917, 0.0239218, 0.0239266, 0.0239314, 0.0239362, 0.0239409, 0.0239457,
0.0239504, 0.0239551, 0.0239598, 0.0239644, 0.0239691, 0.0239737, 0.0239783,
0.0239829, 0.0239875, 0.023992, 0.0239965, 0.0240011, 0.0240056, 0.02401,
0.0240145, 0.024019, 0.0240234, 0.0240278, 0.0240322, 0.0240366, 0.024041,
0.0240453, 0.0240496, 0.024054, 0.0240582, 0.0240625, 0.0240668, 0.024071,
0.0240753, 0.0240795, 0.0240837, 0.0240879, 0.024092, 0.0240962, 0.0241003,
0.0241044, 0.0241085, 0.0241126, 0.0241167, 0.0241208, 0.0241248, 0.0241288}
```

```
ListPlot[ Transpose[{time, meancurve00}], PlotJoined → True,
 PlotStyle → AbsoluteThickness[2], Frame → True, Axes → False ]
```

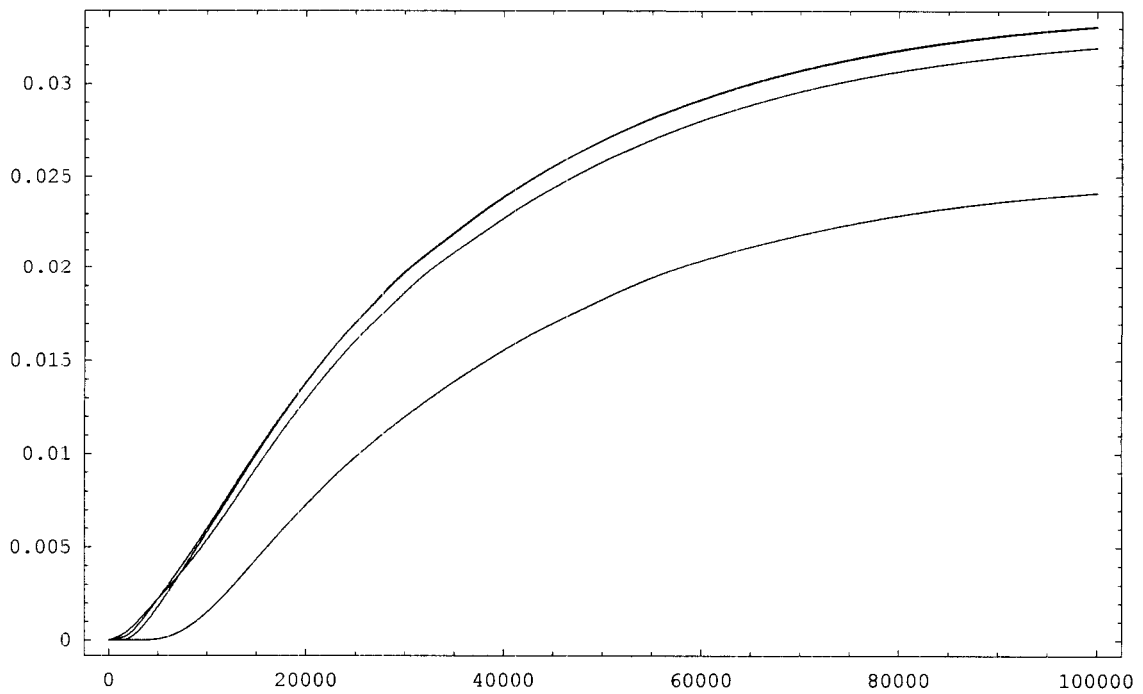


- Graphics -

DCP 1/18/2010

DisplayTogether

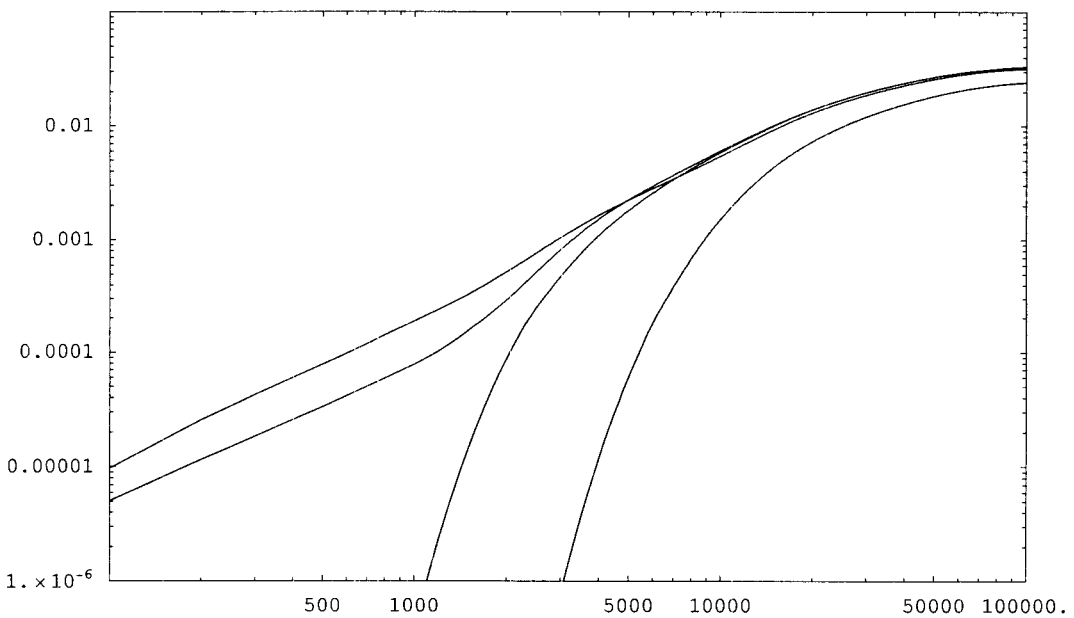
```
DisplayTogether[
 ListPlot[ Transpose[{time, meancurve00}], PlotJoined → True],
 ListPlot[ Transpose[{time, meancurve10}], PlotJoined → True],
 ListPlot[ Transpose[{time, meancurve100}], PlotJoined → True],
 ListPlot[Transpose[{time, meancurve1000}], PlotJoined → True] ,
  Frame → True
]
```



- Graphics -

ED 1/18/2010

```
DisplayTogether[
 LogLogListPlot[ Transpose[{time, meancurve00}], PlotJoined → True],
 LogLogListPlot[ Transpose[{time, meancurve10}],
  PlotJoined → True, PlotPoints → 500],
 LogLogListPlot[ Transpose[{time, meancurve100}],
  PlotJoined → True, PlotPoints → 300],
 LogLogListPlot[ Transpose[{time, meancurve1000}],
  PlotJoined → True, PlotPoints → 500],
 Frame → True, PlotRange → {{2, 5}, {-6, -1}}, Axes → False]
```



- Graphics -

SB 1/18/2018

Colloid Sorption Capacity in Irreversible Model

```
<< Graphics`Graphics`
```

This notebook is used to develop colloid sorption capacity in the abstraction for release due to irreversible sorption to colloids. Consider two cases: one with full range of specific surface areas as reported by DOE, and one with truncated range.
Scott Painter
4/25/06

```
$TextStyle = {FontFamily → "Times", FontSize → 12}
```

```
{FontFamily → Times, FontSize → 12}
```

Sorption Capacity: Full range for surface area

Note: result is calculated by Monte Carlo and will be slightly different each time this is executed.Thus, these results may vary slightly from those in TPA input file.

■ Parameters affecting maximum sorption capacity

Colloid Concentration g/l

```
Cdist = Interpolation[ {{0.0, 0.0005}, {1, 0.05}}, InterpolationOrder → 1]
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

Area m2/g: Log Uniform

```
Adist = Interpolation[ {{0.0, Log[1.8]}, {1, Log[720.]}}, InterpolationOrder → 1]
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

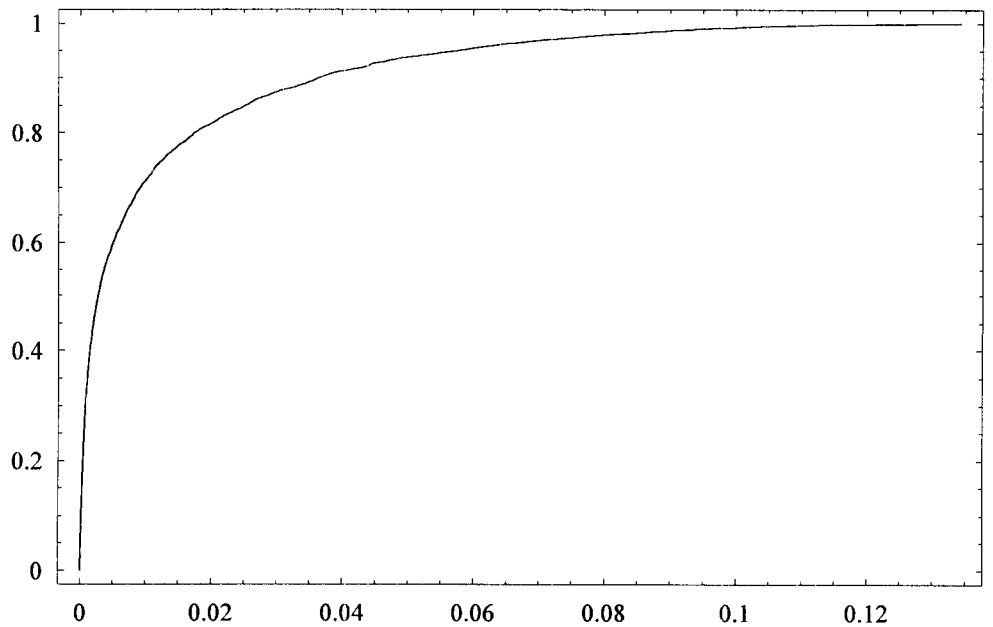
■ Calculate

capacity is SitesPerArea [sites/m2] * SpecificSurfaceArea [m2/g] *
ColloidConcentration [g/liter] * 1000 [liter/m3] /AvogadrosNumber [sites/mole]
result is in moles/m3

```
result = Table[
   Avogadro = 6.022 10^23;
SitesPerArea = 2.3 10^18;
Cc = Cdist[Random[ ] ] ;
A = Exp[ Adist[ Random[ ] ] ];
    SitesPerArea / Avogadro * A *Cc 1000., {5000}];


cdist = Transpose[{Sort[result], (Range[Length[result]] - 0.5) / Length[result]}];
```

Plot is CDF for sorption capacity (moles/m3)

```
ListPlot[cdist, PlotRange → All, PlotJoined → True, Frame → True]
```




```
- Graphics -


foo = Transpose[cdist];
foo = Transpose[{foo[[2]], foo[[1]]}];
f = Interpolation[foo]

InterpolatingFunction[{{0.0001, 0.9999}}, <>]
```

```
Table[ {f[x], x}, {x, 0, 1.0, 0.05}]
```

```
InterpolatingFunction::dmval : Input value {0} lies outside the
   range of data in the interpolating function. Extrapolation will be used. More...

InterpolatingFunction::dmval : Input value {1.} lies outside the
   range of data in the interpolating function. Extrapolation will be used. More...
```

$\{\{5.06387\times10^{-6}, 0\}, \{0.000117724, 0.05\}, \{0.000218907, 0.1\}, \{0.000329752, 0.15\},$
$\{0.000447098, 0.2\}, \{0.000598368, 0.25\}, \{0.000809598, 0.3\}, \{0.00110485, 0.35\},$
$\{0.00147985, 0.4\}, \{0.00200046, 0.45\}, \{0.0027358, 0.5\}, \{0.00367531, 0.55\},$
$\{0.00505319, 0.6\}, \{0.00691716, 0.65\}, \{0.00923996, 0.7\}, \{0.0125958, 0.75\},$
$\{0.0175507, 0.8\}, \{0.0253004, 0.85\}, \{0.0364989, 0.9\}, \{0.0578212, 0.95\}, \{0.135879, 1.\}\}$

```
ColumnForm[%]
```

$\{5.06387\times10^{-6}, 0\}$
{0.000117724, 0.05}
{0.000218907, 0.1}
{0.000329752, 0.15}
{0.000447098, 0.2}
{0.000598368, 0.25}
{0.000809598, 0.3}
{0.00110485, 0.35}
{0.00147985, 0.4}
{0.00200046, 0.45}
{0.0027358, 0.5}
{0.00367531, 0.55}
{0.00505319, 0.6}
{0.00691716, 0.65}
{0.00923996, 0.7}
{0.0125958, 0.75}
{0.0175507, 0.8}
{0.0253004, 0.85}
{0.0364989, 0.9}
{0.0578212, 0.95}
{0.135879, 1.}

Sorption Capacity: Truncated range for specific surface area

Truncate surface area distribution at maximum observed value for goethite.

■ Parameters affecting maximum sorption capacity

Colloid Concentration g/l

```
Cdist = Interpolation[ {{0.0, 0.0005}, {1, 0.05}}, InterpolationOrder → 1]
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

Area m2/g: Log Uniform

```
Adist = Interpolation[ {{0.0, Log[1.8]}, {1, Log[170.]}}, InterpolationOrder → 1]
```

```
General::spell1 :
 Possible spelling error: new symbol name "Adist" is similar to existing symbol "Cdist". More...
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

■ Calculate

capacity is SitesPerArea [sites/m2] * SpecificSurfaceArea [m2/g] *
ColloidConcentration [g/liter] * 1000 [liter/m3] /AvogadrosNumber [sites/mole]
result is in moles/m3

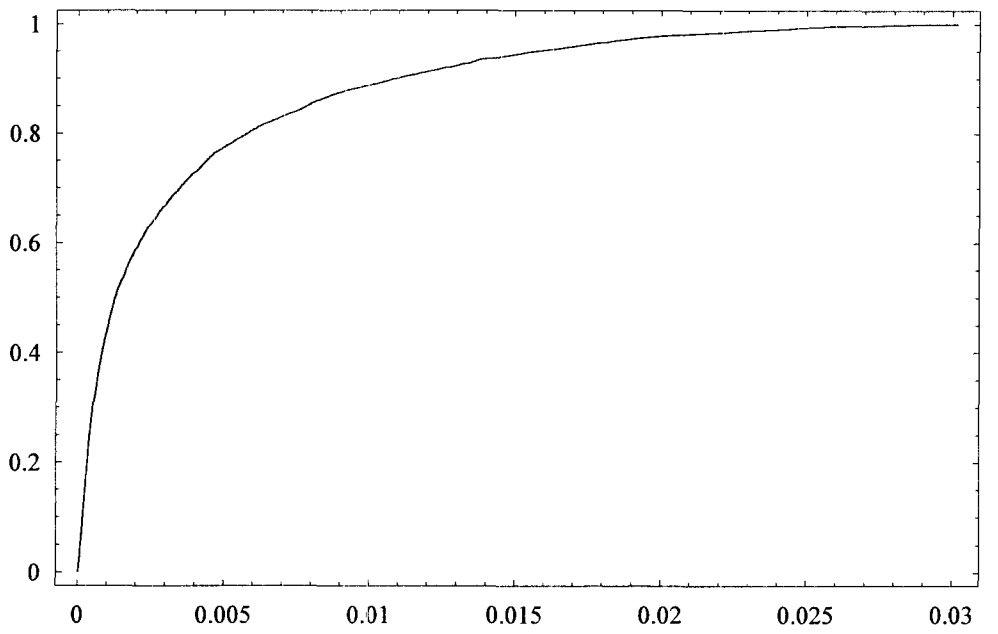
```
result = Table[
   Avogadro = 6.022 10^23;
SitesPerArea = 2.3 10^18;
Cc = Cdist[Random[ ] ] ;
A = Exp[ Adist[ Random[ ] ] ];
    SitesPerArea / Avogadro * A *Cc 1000., {5000}];
```

```
cdist = Transpose[{Sort[result], (Range[Length[result]] - 0.5) / Length[result]}];
```

```
General::spell1 :
 Possible spelling error: new symbol name "cdist" is similar to existing symbol "Cdist". More...
```

Plot is CDF for sorption capacity (moles/m3)

```
ListPlot[cdist, PlotRange → All, PlotJoined → True, Frame → True]
```



- Graphics -

```
foo = Transpose[cdist];
foo = Transpose[{foo[[2]], foo[[1]]}];
f = Interpolation[foo]
```

```
InterpolatingFunction[{{0.0001, 0.9999}}, <>]
```

```
Table[ {f[x], x}, {x, 0, 1.0, 0.05}]
```

```
InterpolatingFunction::dmval : Input value {0} lies outside the
   range of data in the interpolating function. Extrapolation will be used. More...

InterpolatingFunction::dmval : Input value {1.} lies outside the
   range of data in the interpolating function. Extrapolation will be used. More...
```

$\{\{4.4745\times10^{-6}, 0\}, \{0.00010041, 0.05\}, \{0.000174726, 0.1\},$
$\{0.000254874, 0.15\}, \{0.000330204, 0.2\}, \{0.000410312, 0.25\},$
$\{0.000520734, 0.3\}, \{0.000664109, 0.35\}, \{0.000826388, 0.4\},$
$\{0.00103822, 0.45\}, \{0.0012793, 0.5\}, \{0.00165092, 0.55\}, \{0.00210943, 0.6\},$
$\{0.00271641, 0.65\}, \{0.00349138, 0.7\}, \{0.00441781, 0.75\}, \{0.00584205, 0.8\},$
$\{0.00788329, 0.85\}, \{0.0109236, 0.9\}, \{0.0157206, 0.95\}, \{0.0315164, 1.\}\}$

```
ColumnForm[%]
```

$\{4.4745\times10^{-6}, 0\}$
{0.00010041, 0.05}
{0.000174726, 0.1}
{0.000254874, 0.15}
{0.000330204, 0.2}
{0.000410312, 0.25}
{0.000520734, 0.3}
{0.000664109, 0.35}
{0.000826388, 0.4}
{0.00103822, 0.45}
{0.0012793, 0.5}
{0.00165092, 0.55}
{0.00210943, 0.6}
{0.00271641, 0.65}
{0.00349138, 0.7}
{0.00441781, 0.75}
{0.00584205, 0.8}
{0.00788329, 0.85}
{0.0109236, 0.9}
{0.0157206, 0.95}
{0.0315164, 1.}

Scoping Calculations for Radionuclide Release due to Irreversible Sorption to Colloids

```
<< Graphics`Graphics`
```

This notebook has scoping calculations used in developing an abstraction for radionuclide release due to irrevesible sorption to colloids.
Scott Painter
original 8.21.05
finalized with comments 4.21.06

```
$TextStyle = {FontFamily → "Times", FontSize → 12}
```

```
{FontFamily → Times, FontSize → 12}
```

Without Uranium

Parameters affecting relative Affinity for Plutonium and Americium

Plutonium Kd in l/g = m3/kg

```
kPdist = Interpolation[
  {{0, 10^1}, {0.15, 50}, {0.35, 100}, {0.85, 500}, {1, 1000}} , InterpolationOrder → 1]
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

Americium Kd in l/g = m3/kg

```
kAdist = Interpolation[ {{0, 100}, {0.15, 500},
   {0.35, 1000}, {0.90, 5000}, {1, 10000}} , InterpolationOrder → 1]
```

```
General::spell1 :
 Possible spelling error: new symbol name "kAdist" is similar to existing symbol "kPdist". More...
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

Solubility Pu kg/m3

```
SPdist = Interpolation[ {{0.0, 2.4 10^-6}, {1, 2.4 10^-4}}, InterpolationOrder → 1]
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

Solubility Am kg/m3

```
SAdist = Interpolation[ {{0.0, 2.4 10^-8}, {1, 2.4 10^-4}}, InterpolationOrder → 1]
```

```
General::spell1 :
 Possible spelling error: new symbol name "SAdist" is similar to existing symbol "SPdist". More…
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

■ Parameters affecting maximum sorption capacity

Colloid Concentration g/l

```
Cdist = Interpolation[ {{0.0, 0.0005}, {1, 0.05}}, InterpolationOrder → 1]
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

Area m2/g: Log Uniform

```
Adist = Interpolation[ {{0.0, Log[1.8]}, {1, Log[720.]}}, InterpolationOrder → 1]
```

```
General::spell : Possible spelling error: new symbol
   name "Adist" is similar to existing symbols {Cdist, kAdist, SAdist}. More…
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

1/18/2010

EP

Calculate

```
result = Table[
   Avogadro = 6.022 10^23;
SitesPerArea = 2.3 10^18;
Cc = Cdist[Random[ ] ] ;
A = Exp[ Adist[ Random[ ] ] ];
   capacity = SitesPerArea / Avogadro * A *Cc;
   solA = SAdist[Random[] ];
   solP = SPdist[Random[] ];

   kP = kPdist[ Random[] ];
   kA = kAdist[Random[] ];
   denom = solA * kA + solP * kP;
   capA = solA * kA / denom * capacity * 242;
   capP = solP * kP / denom * capacity * 240;
   {capA, capP}, {500}];
```

```
General::spell1 :
 Possible spelling error: new symbol name "solP" is similar to existing symbol "solA". More...

General::spell1 :
 Possible spelling error: new symbol name "capP" is similar to existing symbol "capA". More...
```

```
{resultA, resultP} = Transpose[result];
cdistA = Transpose[{Sort[resultA], (Range[Length[resultA]] - 0.5) / Length[resultA]}];
cdistP = Transpose[{Sort[resultP], (Range[Length[resultP]] - 0.5) / Length[resultP] }];
```

```
General::spell1 :
 Possible spelling error: new symbol name "resultA" is similar to existing symbol "result". More...

General::spell :
 Possible spelling error: new symbol name "resultP" is similar to existing symbols {result, resultA}. More...

General::spell1 :
 Possible spelling error: new symbol name "cdistP" is similar to existing symbol "cdistA". More...
```

With Uranium

repeat, but add uranium

Parameters affecting relative Affinity for Plutonium and Americium

Plutonium Kd in l/g = m3/kg

```
kPdist = Interpolation[
  {{0, 10^1}, {0.15, 50}, {0.35, 100}, {0.85, 500}, {1, 1000}} , InterpolationOrder → 1]
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

Americium Kd in l/g = m3/kg

```
kAdist = Interpolation[ {{0, 100}, {0.15, 500},
   {0.35, 1000}, {0.90, 5000}, {1, 10000}} , InterpolationOrder → 1]
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

Uranium Kd

```
kUdist = Interpolation[ {{0, Log[0.1]}, {1, Log[50.]} }, InterpolationOrder → 1]
```

```
General::spell :
 Possible spelling error: new symbol name "kUdist" is similar to existing symbols {kAdist, kPdist}. More...
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

Solubility Pu kg/m3

```
SPdist = Interpolation[ {{0.0, 2.4 10^-6}, {1, 2.4 10^-4}}, InterpolationOrder → 1]
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

Solubility Am kg/m3

```
SAdist = Interpolation[ {{0.0, 2.4 10^-8}, {1, 2.4 10^-4}}, InterpolationOrder → 1]
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

Solubility of Uranium (log triangular)

```
a = Log[2.4 10^-6];
b = Log[2.4];
c = Log[7.6 10^-3];
SUdist = Interpolation[ {{0, a}, {c, (c - a) / (b - a)}, {1, b}}, InterpolationOrder → 2];
```

```
General::spell :
 Possible spelling error: new symbol name "SUdist" is similar to existing symbols {SAdist, SPdist}. More...
```

■ Parameters affecting maximum sorption capacity

Colloid Concentration g/l

```
Cdist = Interpolation[ {{0.0, 0.0005}, {1, 0.05}}, InterpolationOrder → 1]
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

Area m2/g: Log Uniform

```
Adist = Interpolation[ {{0.0, Log[1.8]}, {1, Log[720.]}}, InterpolationOrder → 1]
```

```
InterpolatingFunction[{{0., 1.}}, <>]
```

■ Calculate

```
result = Table[
   Avogadro = 6.022 10^23;
SitesPerArea = 2.3 10^18;
Cc = Cdist[Random[ ] ] ;
A = Exp[ Adist[ Random[ ] ] ];
   capacity = SitesPerArea / Avogadro * A * Cc;
   solA = SAdist[Random[] ];
   solP = SPdist[Random[] ];
   solU = Exp[SAdist[Random[]] ];

   kP = kPdist[ Random[] ];
   kA = kAdist[Random[] ];
   kU = Exp[ kUdist[Random[] ] ];
   denom = solA * kA + solP * kP + solU * kU;
   capU = solU * kU / denom * capacity * 238;
   capA = solA * kA / denom * capacity * 242;
   capP = solP * kP / denom * capacity * 240;
   {capA, capP}, {500}];
```

```
General::spell :
 Possible spelling error: new symbol name "solU" is similar to existing symbols {solA, solP}. More...

General::spell :
 Possible spelling error: new symbol name "capU" is similar to existing symbols {capA, capP}. More...
```

```
{resultA, resultP} = Transpose[result];
cdistAwithU =
  Transpose[{Sort[resultA], (Range[Length[resultA]] - 0.5) / Length[resultA]}];
cdistPwithU = Transpose[{Sort[resultP],
    (Range[Length[resultP]] - 0.5) / Length[resultP] }];
```

```
General::spell1 :
 Possible spelling error: new symbol name "cdistPwithU" is similar to existing symbol "cdistAwithU". More...
```

Old distributon for comparison

old abstraction defined a constant release factor -- net effect is fixed multiplier on solubility

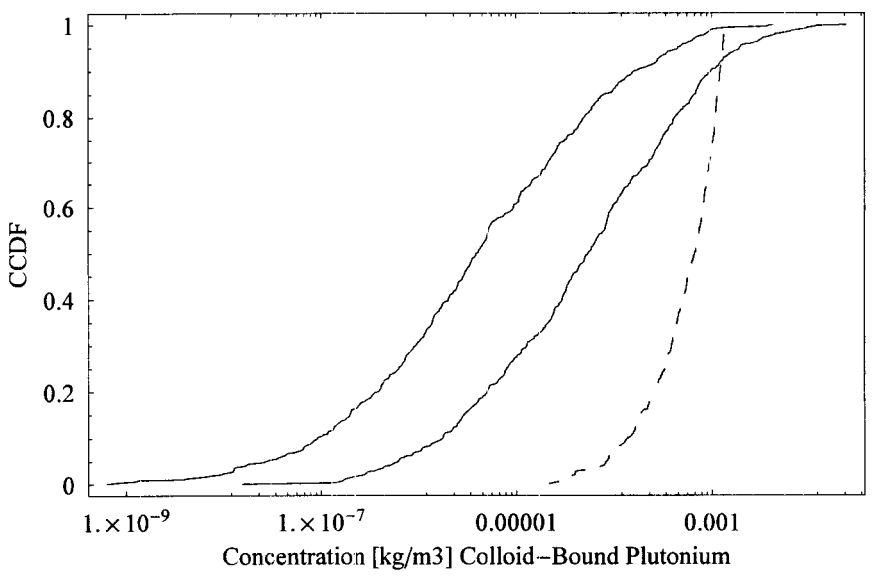
```
resultAold = Table[ solA = SAdist[Random[] ]; solA * 10., {500}];
resultPold = Table[solP = SPdist[Random[]]; solP * 0.85 / (1 - 0.85), {500}];
```

```
General::spell1 :
 Possible spelling error: new symbol name "resultPold" is similar to existing symbol "resultAold". More...
```

```
cdistAold =
  Transpose[{Sort[resultAold], (Range[Length[resultA]] - 0.5) / Length[resultA]}];
cdistPold = Transpose[{Sort[resultPold],
    (Range[Length[resultP]] - 0.5) / Length[resultP] }];
```

```
General::spell1 :
 Possible spelling error: new symbol name "cdistPold" is similar to existing symbol "cdistAold". More...
```

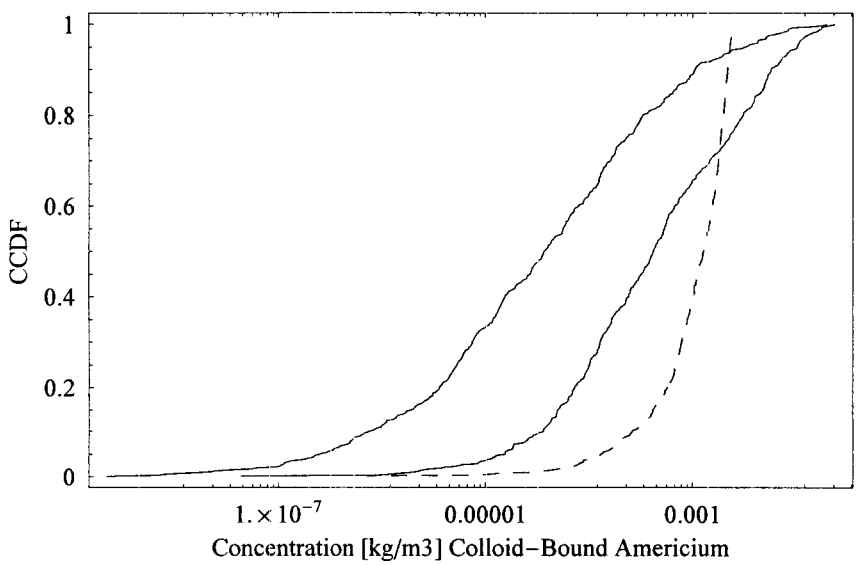
```
DisplayTogether[
 LogLinearListPlot[ cdistPold, PlotRange → All,
  Frame → True, PlotJoined → True, PlotStyle → Dashing[{0.02, 0.02}]],
 LogLinearListPlot[ cdistPwithU, PlotRange → All, Frame → True, PlotJoined → True],
 LogLinearListPlot[ cdistP, PlotRange → All, Frame → True, PlotJoined → True],
  PlotRange → All, Axes → False,
 FrameLabel → {"Concentration [kg/m3] Colloid-Bound Plutonium", "CCDF"} ]
```



- Graphics -

1/18/2010

```
DisplayTogether[
 LogLinearListPlot[ cdistAold, PlotRange → All,
  Frame → True, PlotJoined → True, PlotStyle → Dashing[{0.02, 0.02}]],
 LogLinearListPlot[ cdistAwithU, PlotRange → All, Frame → True, PlotJoined → True],
 LogLinearListPlot[ cdistA, PlotRange → All, Frame → True, PlotJoined → True ],
  PlotRange → All, Axes → False,
 FrameLabel → {"Concentration [kg/m3] Colloid-Bound Americium", "CCDF"} ]
```



- Graphics -

1/15/2010

Show Parameter Distributions

This notebook generates Figures 2-4 for the paper "Effect of kinetic limitations on colloid-facilitated radionuclide transport at the field scale" for the International High-level Radioactive Waste Management Conference. These are assumed parameter distributions for colloid retardation, radionuclide retardation, colloid concentration, and groundwater travel time.

Scott Painter
12-21-2005

```
<< Statistics`DescriptiveStatistics`
<< Graphics`Graphics`
<< Statistics`ContinuousDistributions`
```

```
$TextStyle = {FontFamily → "Times", FontSize → 14}
```

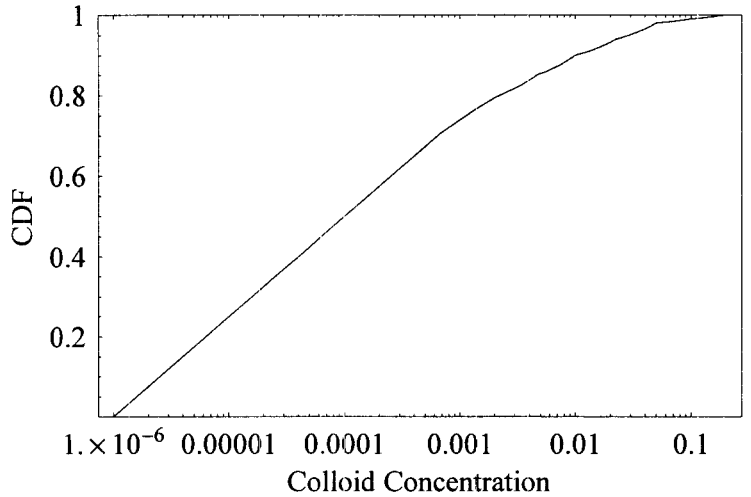
```
{FontFamily → Times, FontSize → 14}
```

Colloid Concentrations

```
ccfunc =
  Interpolation[ Transpose[ {{10^-6, 2.51 10^-6, 6.31 10^-6, 1.58 10^-5, 3.98 10^-5,
      1.0 10^-4, 2.51 10^-4, 6.31 10^-4, 1.0 10^-3, 2.15 10^-3, 4.64 10^-3,
      1.0 10^-2, 2.24 10^-2, 0.05, 0.1, 0.2}, {0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6,
      0.7, 0.75, 0.8, 0.85, 0.9, 0.94, 0.98, 0.99, 1.0}}], InterpolationOrder → 1];
```

1/18/2010

```
LogLinearPlot[ ccfunc[x], {x, 10^-6, 0.2},
 PlotPoints → 300, Frame → True, PlotRange → {All, {0, 1}},
 FrameLabel → {"Colloid Concentration", "CDF"}]
```



- Graphics -

Retardations

Colloids

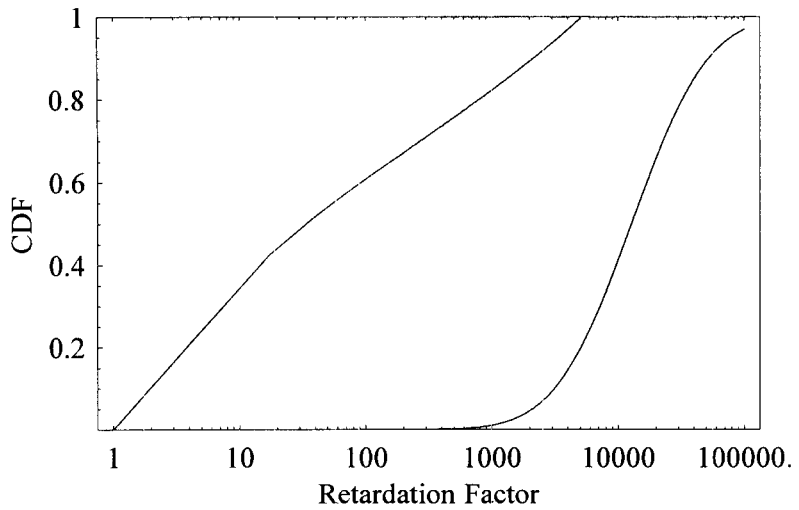
```
Rdist =
  Interpolation[ Transpose[ { Log[{1.0, 8., 33.96, 5188.}], {0, 0.331, 0.5, 1.0} } ]];
```

Plutonium

```
ldist = LogNormalDistribution[9.45708, 1.10];
```

1/18/2010

```
DisplayTogether[
 LogLinearPlot[ CDF[ldist, x], {x, 100, 100000 }, PlotPoints → 300],

 LogLinearPlot[ Rdist[Log[ x]], {x, 1, 5000 }, PlotPoints → 300], Frame → True,
 PlotRange → {All, {0, 1}}, Axes → False,
 FrameLabel → {"Retardation Factor", "CDF"}]
```



- Graphics -

Tau

These are based on TPA 5.0 input

```
nreal = 1000;
```

■ Tau on Central Streamtube

```
centralst = Partition[{0.0, 1500., 0.,
1.0, 1500., 1350.,
3.0, 450., 4350.,
9.5, 250., 10850.,
13.0, 400., 14350.,
13.5 , 375., 14850.,
15.0, 325., 16350.,
18.0 , 225., 19350.}, 3] ;
centralst = Transpose[centralst];
centralst1 = Transpose[{centralst[[1]], centralst[[3]]}]
centralst2 = Transpose[{centralst[[3]], centralst[[2]]}]
```

```
{{0., 1500.}, {1350., 1500.}, {4350., 450.}, {10850., 250.},
 {14350., 400.}, {14850., 375.}, {16350., 325.}, {19350., 225.}}
```

```
cstfunc1 = Interpolation[centralst1]
cstfunc2 = Interpolation[centralst2]
```

```
InterpolatingFunction[{{0., 18.}}, <>]
```

```
InterpolatingFunction[{{0., 19350.}}, <>]
```

```
Q = 252;
taudist = Table[
   sldist = Random[Real, {10, 18.}];
   tddist = cstfunc1[sldist];
avgwidth = NIntegrate[
      cstfunc2[td], {td, tddist, 18000},
      MinRecursion → 5, MaxRecursion → 12] / (18000 - tddist);
   darcyv = Q / avgwidth;
   porosity = Random[Real, {0.1, 0.15}];
   velocity = darcyv / porosity;
   tau = (18000 - tddist) / velocity;
   Max[tau, 10.], {nreal}];
```

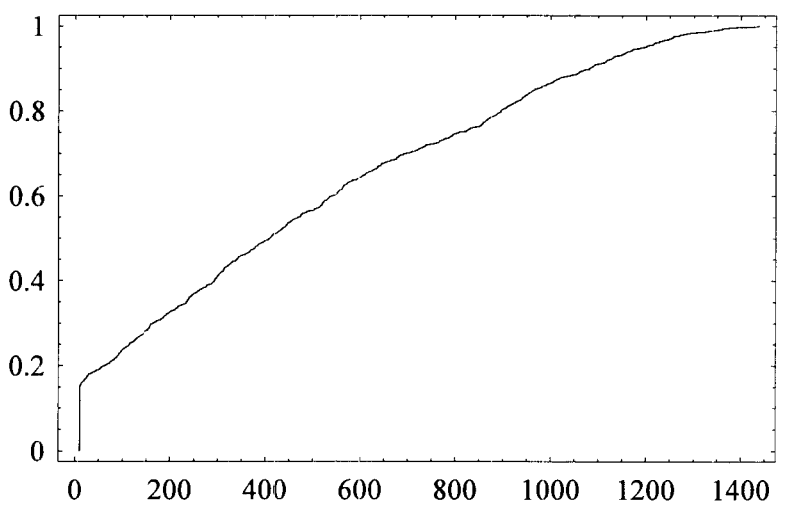
```
Length[taudist]
```

```
1000
```

```
cdf = Transpose[ {Sort[taudist], (Range[Length[taudist]] - 0.5) / Length[taudist]}];
```

```
General::spell1 : Possible spelling error: new symbol name "cdf" is similar to existing symbol "CDF". More…
```

```
ListPlot[ cdf, PlotJoined → True, Frame → True]
```




```
- Graphics -
```

SCIENTIFIC NOTEBOOK

by

Scott Painter

Southwest Research Institute
Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

May 8, 2006

Table of Contents

INITIAL ENTRIES....3
Radionuclide Transport Analysis for Yucca Mountain....4
Objective:....4
5.08.06 Alternative distribution for plutonium solubility limit....5
6.5.06 Repeat of previous entry....5
6.27.06 Relative affinity factors in colloid-facilitated release abstraction....5

INITIAL ENTRIES

Scientific notebook: #318E Vol. 13
Issued to: S. L. Painter
Original Issue Date: 22-March-1999

By agreement with CNWRA QA this notebook will be printed at approximate quarterly intervals. This computerized electronic notebook is intended to address the criteria of CNWRA QAP-001.

Radionuclide Transport Analysis for Yucca Mountain

Account number: 20-06002.01.241 and 242
Description: Radionuclide Transport Analysis.
Collaborators: Drs. D. Pickett, and V. Cvetkovic (consultant)

Objective:
Development and testing of transport analysis methodology for spatially variable systems including demonstrations and applications to the proposed repository at Yucca Mountain.

28 1/15/2010

SN 318E Vol. 13 Pg. 4, Scott Painter

5.08.06 Alternative distribution for plutonium solubility limit

An alternative calculation for the plutonium solubility limit distribution based on an approach of DOE is given in the attached Mathematica Notebook: *Pu Solubility Limit.nb*. This work is for the ENG4 ISI.

6.5.06 Repeat of previous entry

The calculation described in the entry dated 5.08.06 was repeated with a smaller range of uncertainty. See *Pu Solubility Limit-2.nb* for details. The resulting cumulative distribution function is shown in the following table. This first column is Pu solubility in kg/m3. The second is probability value.

7.112×10^{-7}	0
3.936×10^{-6}	0.05000
5.807×10^{-6}	0.1000
7.817×10^{-6}	0.1500
0.00001014	0.2000
0.00001249	0.2500
0.00001548	0.3000
0.00001817	0.3500
0.00002186	0.4000
0.00002638	0.4500
0.00003149	0.5000
0.00003751	0.5500
0.00004495	0.6000
0.00005351	0.6500
0.00006318	0.7000
0.00007805	0.7500
0.00009507	0.8000
0.0001204	0.8500
0.0001579	0.9000
0.0002457	0.9500
0.0003228	0.9700
0.0005152	0.9900
0.001324	1.000

6.27.06 Relative affinity factors in colloid-facilitated release abstraction

The abstraction for colloid-facilitated release includes relative affinity factor parameters (see 4.25.06 entry). These are defined relative to plutonium (i.e. value for plutonium is set to 1). This choice is arbitrary. The abstraction is mathematically equivalent if all affinity factors had been defined relative to another radionuclide.

The affinity factors can be shown to be ratios of forward rate constants. The forward rate constants do not appear explicitly in the abstraction because they are assumed to be fast relative to transport

processes. The ratios of the rates (i.e. the relative affinity factors) do affect the competition for sites between radionuclides.

Little information exits on the forward sorption rates. Review of data from the kinetic sorption experiments of Lu et al (1998, 2000) suggest that americium sorbs onto hematite colloids faster than plutonium and plutonium sorbs faster than uranium. After review of that data, a uniform distribution in the range 5 – 15 was chosen for americium relative affinity. Recall that this is a dimensionless number. A log-uniform distribution between 0.005 and 0.05 was chosen for uranium. Affinity factors for other radionuclides were assigned the same distribution as americium. Note that these distributions are based largely on expert judgment after reviewing the limited amount of information from the Lu et al experiments. The abstraction is much less sensitive to the choice of the affinity factors than to the total sorption capacity for colloids.

References:

Lu et al. 1998 Los Alamos National Laboratory Technical Report LA-UR-98-3057
Lu et al. 2000 Los Alamos National Laboratory Technical Report LA-UR-00-5121.

2.23.2007 Recalculation of colloid sorption capacity

Recalculated colloid sorption capacity for use in TPA abstraction of release of colloid-associated radionuclides. New calculation uses a smaller range of specific surface areas and a smaller range of colloid concentration. Rationale for the new values are given by D. Pickett in notebook 133. Calculation is otherwise identifical to previous calculation. Details are in the attached Mathematica notebook *Colloid Sorption Capacity revised.nb*.

Colloid concentration moles/m3	Cumulative probability
7.37E-06	0
9.45E-05	0.05
1.93E-04	0.1
2.84E-04	0.15
3.76E-04	0.2
4.63E-04	0.25
5.49E-04	0.3
6.39E-04	0.35
7.49E-04	0.4
8.69E-04	0.45
1.03E-03	0.5
1.23E-03	0.55
1.50E-03	0.6
1.76E-03	0.65
2.09E-03	0.7
2.49E-03	0.75
3.04E-03	0.8
3.73E-03	0.85

4.53E-03	0.9
5.97E-03	0.95
1.10E-02	1

10.1.2007 Multirate model applied to alluvial aquifer: initial entry

Received a draft manuscript from V. Cvetkovic (consultant) on application of the multirate model to transport in the alluvium near Yucca Mountain Nevada. The manuscript is at an early draft stage and will need extensive revision. It is attached as *multirateD1.pdf*. The mathematica notebooks containing the supporting calculations are attached as part of this notebook. They are titled *D1Figure2.nb* etc.

11.26.2007 Multirate manuscript

The following manuscript will be submitted to Water Resources Research.

Field-scale transport implications of sorption kinetics with multiple rates

V. Cvetkovic. Water Resources Engineering, Royal Institute of Technology (KTH), SE-10044 Stockholm, Sweden (vdc@kth.se).

S. Painter. CNWRA, Southwest Research Institute, San Antonio, Texas 78228.

Abstract. Sorption kinetics governed by multiple kinetic rates, a likely consequence of diffusion limitations induced by fine-scale heterogeneity, has been observed previously in field tests. Effects of advection, multirate exchange (diffusion–sorption) and spatial variability in the partitioning (distribution) coefficient, are combined using a simple probabilistic model for tracer residence time. Kinetic limitations are shown to be important when a Damkohler number defined as the product of the geometric mean sorption rate and the mean advective travel time is less than approximately 0.1. The result of kinetics in this situation is earlier arrival and higher expected

peak discharge. A combination of batch and batch-flush laboratory tests on disturbed and undisturbed samples using multiple tracers with contrasting sorption properties is suggested as a means for constraining the retention model and associated parameters. A naive interpretation of batch tests as equilibrium and subsequent predictive modeling assuming equilibrium may underestimate retention, depending on the groundwater travel time. Global estimates of multirate distribution parameters yield a geometric mean of <0.001 1/yr and log-variance around 2-3 for strongly sorbing tracers and a representative elementary volume (REV) scale > 1 m; groundwater travel times of a few hundred years are required in this situation to reach equilibrium. A set of type curves link mean groundwater residence time, sorption coefficient and REV scale for a rough estimate of the applicability of the equilibrium approximation for field-scale transport modeling.

1. Introduction

Important applications in subsurface hydrology, such as those related to geological disposal of nuclear waste, require predictions of transport on large scales and over long times. The purpose of such predictions is to assess the capacity of a geological medium to act as a barrier to contaminant migration.

The barrier function of geological media for any type of contaminant rests on a balance between three interacting, and to some extent, competing processes. First, if significant flow exists,

advection (and the resulting hydrodynamic dispersion) is the main mechanism to spread a dissolved tracer. As a rule, contaminants of interest (both inorganic and organic) interact by physical and chemical mechanisms with the solids and immobile fluids of an aquifer, resulting in tracer retention. With advection spreading contaminants, and retention slowing down the spreading, a key process from the safety perspective is decay, or degradation. Thus, the interaction between the three mechanisms of advection, retention, and decay will ultimately determine the level of risk in any given application.

Given that many contaminants of interest interact strongly with immobile minerals in the subsurface, retention processes are often key in long-term predictive modeling. When the retention processes are relatively rapid compared with the transport time scales of interest, the primary retention parameter of interest is the equilibrium distribution coefficient, applicable for a given chemical condition. Chemical sorption is the relevant retention mechanism in this case. In heterogeneous media, however, the scale of a representative elementary volume (REV) may be on the order of meters or larger, with an internal (microscale) heterogeneous structure that combines high- and low-permeability (porosity) regions. One possible effect of this internal (sub-REV) heterogeneous structure is to introduce potentially significant rate limitations because tracers must first diffuse from mobile zones into immobile zones before accessing sorption sites. The importance of these kinetic limitations for field-scale transport has long been recognized and single-rate, first-order kinetic models have traditionally been used to represent the effect (e.g., *Painter et al.* [2001]).

There is significant evidence that retention processes in geological media not only deviate from equilibrium, but also exhibit more complex kinetics than implied by the simplest, first-order model with a single rate. The existence of multiple rates may be caused by a complex internal structure of geological media on smaller scales, which may create immobile or nearly immobile zones with a range of sizes that inhibit access to sorption sites. Multirate kinetic models [*Haggerty and Gorelick*, 1995; *Haggerty et al.*, 2000,2001; *Cvetkovic and Haggerty*, 2002] that use a continuous spectrum of rates have been used to represent processes governed by multiple rates.

Although kinetic limitations, including multirate processes, are generally recognized as being important for transport over short distances, kinetic effects are typically neglected when considering transport on larger scales. Such an equilibrium approximation is clearly appropriate when the kinetic effects are governed by a single rate that is relatively rapid on the time scales of interest. However, the applicability of equilibrium models is less clear when the kinetic limitations are controlled by a broad spectrum of rate constants. In this paper, we present a simple modeling tool that couples the effects of microscale heterogeneity (rate-limited mass transfer) and field-scale heterogeneity (macrodispersion and field-scale variations in sorption coefficients). This model is used to explore the implications of multirate kinetic sorption effects on field scale transport modeling in generic and site-specific contexts. In addition, issues regarding design of laboratory experiments for estimating retention parameters are addressed.

2 Tracer transport with kinetic sorption governed by multiple rates

2.1 Problem description and assumptions

Three key processes are assumed to take place in an aquifer: advection (including macrodispersion), retention and decay. We consider these processes dominant for characterizing transport in a mean sense and on large scales. The transport domain is heterogeneous on a wide range of scales (from cm to km) with spatially varying physical and chemical properties. A configuration sketch of the transport problem is given in Figure 1.

The groundwater flow in the aquifer is controlled by a steady-state regional gradient and is spatially variable due to varying hydraulic properties. The longitudinal (transport) scale is denoted by L, whereas in the transverse direction the plume is visualized as meandering, with an average roughly determined by the source size assumed several orders of magnitude smaller than L. Based on available hydraulic information, spatially variable groundwater flow is resolved on the scale of a "representative elementary volume" (REV). We have $\ell/L \ll 1$ with macroscale (resolved) heterogeneity on the scale $> \ell$ and the (unresolved) microscale heterogeneity is on the sub-REV scale $< \ell$. The plume can be viewed as advected along "stream tubes" derived from a spatially variable velocity field with support on the scale ℓ with varying travel times that quantify macroscopic (field-scale) dispersion.

Retention takes place due to mass transfer processes within an REV. Typically, equilibrium is assumed and a sorption coefficient (K_d) is obtained from a batch test by saturating samples whose

internal structure has been disturbed (destroyed or homogenized), with chemical sorption being the dominant exchange mechanism. In large-scale problems however the scale of an REV may be on the order of meters or tens of meters, with an internal (microscale) heterogeneous structure that combines high and low permeability (porosity) regions. An undisturbed sample that preserves the internal (sub-REV, microscale) heterogeneous structure would therefore exhibit potentially significant rate-limitations due to combined effects of diffusion and sorption, but would ultimately yield the same K_d as batch tests on disturbed samples, given sufficiently long time.

For the present analysis the following is assumed.

• On the macroscale hydraulic and retention properties are heterogeneous; set in the context of the problem scale and the scale of hydraulic field measurements, the REV scale for the flow (on which groundwater velocity is resolved) is on the order of meters or larger.

• On the microscale (sub-REV), the medium is heterogeneous with potentially high contrasts in permeability and porosity.

• Due to the microscale heterogeneity, flow within an REV is conceptualized as a dual-porosity system, where a tracer is advected through the faster regions and slowly advected through and/or diffused into/through the low-permeability regions.

• Mass transfer takes place between the high-permeability ("mobile") and low-permeability ("immobile") zones within an REV; we capture this by assuming a multirate diffusion/sorption retention model with K_d that is variable on the macroscale.

2.2 Theory

The groundwater velocity is assumed to be steady in time, but spatially variable due to the spatial variability of the hydraulic conductivity $K(\mathbf{x})$. A dissolved tracer is injected at a given location $\mathbf{x}_0$, and is transported downstream toward a monitoring boundary. For simplicity, the mean groundwater flow direction is aligned with the x_1-axis of a three-dimensional coordinate system, $\mathbf{x}(x_1, x_2, x_3)$. The transport distance from the injection point to a monitoring location is $x_1 = L$, where the monitoring surface is orthogonal to the mean flow.

For predicting field-scale tracer discharge across the monitoring surface at L, an estimate of the groundwater residence time density from the source area to the monitoring location is required. If the transport problem is on a relatively large scale where tracer tests are not feasible, the distribution of groundwater residence time, τ, may be estimated based on the statistics of the hydraulic properties.

The injected tracer is advected by groundwater along stream tubes defined on the macroscale by $\mathbf{Z}(t)$ [*Dagan*, 1984]. The dimensionless partitioning (distribution) coefficient (once equilibrium is reached) is $K_d(\mathbf{x})$. To account for spatial variability in K_d, we require the statistics of a global

Lagrangian random variable B defined as [*Cvetkovic et al.*, 1998]

$$B \equiv \int_0^{\tau} K_d(\tau')d\tau' \quad (1)$$

where $K_d(\tau)$ follows $\mathbf{Z}(t)$ as $K_d(\tau) = K_d[\mathbf{Z}(\tau)]$ where $\mathbf{x}_0 = \mathbf{Z}(0)$. Random variables τ and B will be correlated to some extent; hence in the general case we require a joint density for τ and B. If K and K_d are uncorrelated or weakly correlated, it is reasonable to assume τ and B as uncorrelated [*Painter et al.,* 2001].

Unconditional residence time density of a tracer at a monitoring boundary at L is defined by [*Cvetkovic et al.*, 1998]

$$h(t;L) = \iint \gamma(t;\tau,B) f(\tau) f(B) d\tau dB \quad (2)$$

where $f(\tau)$ and $f(B)$ denote densities for τ and B at L, respectively, and γ quantifies the conditional residence time density, which can be derived from mass balance equations (Appendix A). Expected tracer discharge across the monitoring location at $x_1 = L$ is proportional to h; if continuous tracer injection is applied, expected tracer discharge is obtained from equation (2) by convolution with the specified injection function.

Previous work has shown that for generalized linear retention processes (Appendix A)

$$\hat{\gamma} = \exp[-s(\tau + B\hat{g})] \quad (3)$$

where $\hat{g}$ is the Laplace transform of the so-called "memory function" that characterizes diffusion-sorption kinetics. A convenient form of g is obtained using a distribution of kinetic rates as $\hat{g} = \int k\varphi(k)/(s+k)dk$ [*Haggerty and Gorelick*, 1995; *Cvetkovic and Haggerty*, 2002]. In

this study, we use a discrete version defined by

$$\hat{g} = \sum_{i=1}^{N} \frac{\varphi_i k_i}{s + k_i} \tag{4}$$

where k_i[1/T] is a mass transfer rate and φ_i [-] is a weighting function for sorption sites associated with the mass transfer rate k_i. In other words, we view diffusion–sorption (mass transfer) as taking place between mobile and immobile zones within an REV through a finite number N of sites types, each type with its own rate k_i, weighted by a specified (discrete) function φ_i which reflects the internal (microscale) heterogeneity; we require $\sum_{i=1}^{N} \varphi_i = 1$. The classical single-site (first-order linear) sorption model considered, for example, by *Painter et al.* [2001] is recovered with N = 1, where $k_1 K_d$ is the forward rate and k_1 is the reverse rate.

Combining equations (2–3) and (4), the unconditional residence time density h can be written in the Laplace domain as

$$h(t;L) = \mathcal{L}^{-1}\left\{ \hat{f}_\tau(s) \hat{f}_B\left(s \sum_{i=1}^{n} \frac{\varphi_i k_i}{s + k_i} \right) \right\} \tag{5}$$

where $\mathcal{L}^{-1}$ denotes inverse Laplace transform. Equation (5) is to be used for predicting expected tracer discharge across a monitoring boundary at $x_1 = L$, for given forms of $f(\tau)$, $f(B)$ and φ. For a decaying tracer, s should be replaced in equation (5) by $s + \lambda$ where λ is the decay rate.

Different models for φ have been suggested in the literature, such as log-normal, gamma and

power-law. In the following analysis, a log-normal distribution with a geometric mean k_G and log-variance $\sigma^2_{\ln k}$ is assumed as the multirate model.

The advective travel time density to be used in this study is the one derived from the advection-dispersion equation for a tracer pulse in a semi-infinite domain

$$\hat{f}_\tau(s) = \exp\left[c_\tau \alpha_\tau^{1/2} - c_\tau(\alpha_\tau + s)^{1/2}\right] \quad (6)$$

where a_τ and c_τ are parameters related to the mean groundwater velocity U and the longitudinal macrodispersion coefficient D_x as $a_\tau = U^2/4D_x$ and $c_\tau = L/\sqrt{D_x}$ [*Kreft and Zuber,* 1978]. Note that this distribution can be generalized as the "truncated one-sided stable" distribution where 1/2 in equation (6) is replaced by an exponent between 0 and 1 (*Cvetkovic and Haggerty* [2002]), suitable for studying non-Fickian transport (with $a_\tau \neq 0$) or anomalous transport (with $a_\tau = 0$, e.g., *Scher et al.* [2002]).

Given the definition (1), we shall assume that B has a distribution analogous to τ (6), i.e.,

$$\hat{f}_B(s) = \exp\left[c_B \alpha_B^{1/2} - c_B(\alpha_B + s)^{1/2}\right] \quad (7)$$

2.3 Generic sensitivities

Transport results are illustrated in this section using dimensionless quantities. The mean groundwater residence time from the source to the monitoring boundary, denoted by τ_m, is used as the normalization time for transport. Since τ and B have both dimensions time, c is normalized

in (6) and (7) as $c/\tau_m^{1/2}$ and a as $a\tau_m$. The dimensionless parameters a and c in (6)-(7) are defined in terms of the coefficient of variation as

$$c = \frac{\sqrt{2}}{\mathrm{CV}} \qquad a = \frac{1}{2\,\mathrm{CV}^2} \tag{8}$$

where CV is either $\mathrm{CV}_\tau \equiv \sigma_\tau/\tau_m$ or $\mathrm{CV}_\mathrm{B} \equiv \sigma_B/B_m$; first- order expressions for B_m, σ_B^2 are given in *Cvetkovic et al.* [1998].

The Laplace transform variable is also normalized with τ_m, implying that h is dimensionless. Finally, the normalization of retention rates k_i with τ_m implies that for a log-normal φ_i, dimensionless k_G sets kinetics relative to advection and corresponds to the Damkohler number. Note that the normalized B_m is the arithmetic mean sorption coefficient $K_d^A = B_m/\tau_m$ (Appendix B).

For illustrating potential effects of field-scale advection and multirate sorption, under conditions of spatial variability in the sorption properties, we shall use (6) and (7) for τ and B, respectively. For pulse injection of a nondecaying tracer, the dimensionless tracer discharge is equivalent to h (2). In computing h from (2), Laplace inversion is carried out numerically. In these computations, we assumed a generic value λ =0.01. As already noted, all time-related quantities in this section are normalized with τ_m.

The impact of the geometric mean rate (Damkohler number) k_G is illustrated in Figure 2 where

we set $\mathrm{CV}(\tau) \equiv \mathrm{CV}_\tau = 0.5$ and $\mathrm{CV}(B) \equiv \mathrm{CV}_B = 0.9$, and assume $\sigma^2_{\ln k} = 2$, with $K_d^A = 1000$. The geometric mean rate is considered in the four-order of magnitude range $k_G = 0.0001-1$. The kinetic effects are apparent for $k_G = 0.0001-0.01$, whereas for $k_G = 1$ the BTC can be well approximated by an equilibrium model; for $k_G = 0.1$, the BTC deviates only slightly from the equilibrium model. The BTC is quite sensitive to k_G in the range $k_G = 0.001-0.1$, where the shape of the BTC can change significantly.

We also tested the sensitivity of the BTCs to the microscale heterogeneity as quantified by the multirate variance $\sigma^2_{\ln k}$, and found that for $\sigma^2_{\ln k}$ between 1.5-3, the BTCs of Figure 2 where relatively little affected; a slight impact could be seen in the tail parts of the BTCs for k_G in the kinetic rage 0.0001-0.01.

3. Physical model for the kinetic rate distribution

We wish to estimate the multirate distribution parameters for a given aquifer. Significant for our choice of a log-normal multirate distribution $\varphi(k)$ is the fact that it can be parameterized using a spherical model for the low-permeability (immobile) zones. To this end, the REV of a real medium is replaced by an idealized equivalent medium, with the same REV velocity but spherical immobile zones (Figure 3). We emphasize that by introducing spheres we are not intending to capture the actual shape, but rather the scale of the low-permeability zones in which a potentially substantial part of the sorption sites is present. The spheres (or "sites") within an REV are of varying diameters, with varying physical-chemical properties.

Comparing spherical and multirate models, it can be shown that $k_i \approx 3D_i/R_i A_i^2$ (Appendix B, *Haggerty et al.* [2000]). In other words, D_i, R_i and A_i may be used to parameterize the log-normal multirate distribution, where A_i [L] is the diameter, D_i [L^2 /T] is the diffusion coefficient into a "sphere" (or site type i), and $R_i = 1 + K_{di}$ with K_{di} being the sorption coefficient once equilibrium is reached for site i.

To make useful estimates, it is first recognized that the diffusion coefficient can be expressed as

$$D_i = \frac{F_i D_w}{\phi_i} \qquad F_i = \phi_i^{m_i} \tag{9}$$

where F_i is the formation factor, D_w is the diffusion coefficient in water, ϕ_i is the porosity, for site i and we have used Archie's law [*Clennell*, 1997], with m_i being the cementation exponent, typically in the range 1.3–2. Thus $D_i = \phi_i^{m_i - 1} D_w$ and the diffusion-sorption rate becomes

$$k_i \approx \frac{3\phi_i^{m_i - 1} D_w}{R_i A_i^2} \tag{10}$$

where $R_i = 1 + (1-\phi_i)\rho K'_{di}/\phi_i$ with K'_{di} [L^3/M] being the partitioning coefficient at site type i; ρ is the solid material density assumed approximately constant. Global estimates of k_i statistical parameters are made using equation (10) in the following.

In order to provide global estimates, we recognize the following.

• Data from boreholes taken for instance from alluvial aquifers (e.g., http://www.nyecounty.com/ewdpmain.htm), typically indicates zones of diverse sedimentary material, from sand and gravel to clay, with varying electrical conductivity (porosity) on a relatively small scale; such heterogeneity could imply a relatively wide sub-REV porosity range.

• Formation factor has been studied in detail for glacial sediments [Salem, 2001]; it has been found that the cementation exponent m varies in the range 1.5-2. For our illustration purpose we assume $m = 1.7$.

• The spherical diameter A, central to the idealized concept of the microscale heterogeneity (Figure 3), should be understood as a characteristic scale of sub-REV heterogeneity and it depends on the scale of the REV. We assume a uniform distribution in the range $A_{\min} < A < A_{\max}$ with $A_{\min} = 0.03$ m and $A_{\max} \approx \ell/3$, where ℓ is the REV scale, and treat ℓ as a sensitivity parameter. In applications, detailed analysis of well logs or outcrops might provide some constraints on realistic range A_{min} or A_{max}.

With ranges of A and ϕ specified (assumed), k can be computed/sampled using equation (9) and a global estimate of the multirate distribution of k can be obtained. A straightforward Monte Carlo methodology is implemented where A is sampled uniformly in the range 0.03- $\ell/3$ m with $D_w = 0.03$ m^2/yr, $m = 1.7$ and solid phase density of 1800 kg/m^3 ; for these estimates, K_d is to be assumed constant. The geometric mean k_G and log-variance $\sigma^2_{\ln k}$ are shown as a function of the

REV scale ℓ in the range $\ell = 1-10$ m in Figure 4. It can be seen that with increasing ℓ, k_G decreases and $\sigma^2_{\ln k}$ increases monotonically. Clearly these ranges could be further explored by considering the porosity and the cementation exponent as random (possibly correlated) variables; the values in Figure 4 provide a rough estimate of the multirate parameters as a function of the REV scale.

5. Possible strategies for *ex-situ* experiments

In-situ tracer tests using either weakly sorbing tracers [e.g. *Meigs and Beauheim*, 2001; *Haggerty et al.*, 2001; *McKenna et al.*, 2001; *Reimus et al.,* 2003, 2007] or multiple tracers with strongly contrasting sorption properties [*Widestrand et al.,* 2007; *Cvetkovic et al.*, 2007] are indispensable for detecting and understanding diffusion-controlled retention processes. However, such tests are resource-demanding, difficult to interpret, and generally limited by practical considerations to relatively small subsurface volumes. For large subsurface volumes exhibiting significant spatial variability in retention properties, such as would be typical for a high-level radioactive waste repository, *in-situ* tracer tests alone are unlikely to be sufficient for inferring model parameters.

One practical approach for characterizing retention properties of large subsurface volumes in fractured or granular media is to extract multiple samples from different locations and perform *ex-situ* tests. Typically, tracers in solution are added to repacked samples and allowed to partition over a given time; the sorption coefficient is then inferred assuming equilibrium. The relative simplicity of this approach makes it feasible to sample multiple locations, thus generating

information about large-scale variability in the region of interest. However, batch sorption experiments on samples that have been repacked yield no information on diffusion-limited rates because the internal microstructure that is responsible for the rate limitations is destroyed in the repacking. An alternative approach [*Turner et al.,* 2002] that does not require sampling aquifer material is to extract groundwater and determine the sorption coefficient based on the water chemistry and a geochemical model, again implicitly assuming equilibrium.

A more ambitious laboratory testing campaign might be based on (relatively) undisturbed samples that arguably provide a more realistic picture of the retention processes since the structure of aquifer material is preserved. Even more advanced *ex-situ* methods could involve combinations of batch and flushing tracer tests, and ultimately column tracer experiments. How parameters in a field-scale multirate model might be inferred from such small-scale, laboratory experiments or samples is not a straightforward question. Practical limitations on the experiment duration and sample size may prevent the full spectrum of relevant rates from being observed. Clearly, the existence of multiple kinetic rates has the potential to introduce biases in parameters inferred from laboratory experiments, with as yet unexplored consequences for field-scale transport modeling.

Here we consider a series of hypothetical laboratory experiments using undisturbed, and compare potential outcomes to classical disturbed sample experiments. With undisturbed samples, one can run various types of tests: batch, batch followed by flushing of tracers, or column experiments. Single or multiple tracers can be used, where a strong contrast in sorption properties

of the tracer can be critical for interpreting the outcome [*Cvetkovic et al.,* 2007]. Once tests are carried out on undisturbed samples, the same samples can be used by breaking the internal (microscale) structure and running them in a classical (disturbed) batch mode. A combination of these tests can provide information for characterization of multirate sorption. In the following, we limit the discussion to the batch and batch-flush tests with multiple tracers of strongly varying sorption properties, illustrating a few interesting possible outcomes.

5.1 Batch test

Classical batch tests are carried out by saturating a sample with a tracer solution and leaving it for some time (denoted by T_0 [T]) to equilibrate. The concentration in the aqueous phase can be monitored over time $t < T_0$; from the concentration versus time curve, sorption parameters (usually assuming a first-order model) can be inferred. Alternatively, one measures concentration only at $t = T_0$ when the test is terminated; from the partitioning of the tracer between the mobile and immobile phases, an equilibrium sorption coefficient can be inferred.

If the exchange is controlled by the diffusion-sorption that is described with multiple rates, then the actual outcome of a batch test can be quantified from the mass balance equations. Specifically, we arrive at the following system of equations in the Laplace domain:

$$s\hat{X} - 1 = -\sum_{i=1}^{N} k_i\left(\varphi_i K_d \hat{X} - \hat{X}_i^*\right)$$
$$s\hat{X}_i^* = k_i\left(\varphi_i K_d \hat{X} - \hat{X}_i^*\right) \qquad (11)$$

where X [-] is the normalized tracer mass in the mobile phase, X_i^* [-] is the normalized tracer mass on the immobile phase on site type i, k_i is the backward rate for site type i, φ_i is as before a weighting function (probability) for site type i, and K_d is the dimensionless partitioning coefficient once equilibrium is reached.

The solution of (11) is

$$\hat{X}_i^* = \frac{K_d k_i \varphi_i}{s + k_i} \hat{X}$$

$$\hat{X} = \left[s\left(1 + \sum \frac{k_i \varphi_i K_d}{s + k_i} \right) \right]^{-1} \qquad (12)$$

Inverting equation (12) yields the partitioning $\sum_i X_i^* / X$ that would actually take place in a batch test on an undisturbed sample, assuming that exchange is controlled by a multirate diffusion–sorption process.

The ratio $\kappa \equiv \frac{K_d^*}{K_d}$ is shown in Figure 5 as a function of the dimensionless group $k_G T_0$ with $\sigma^2_{\ln k} = 2$. Here $K_d^* \equiv \sum_i X_i^* / X$ is the estimate that would have be obtained from undisturbed samples by measuring the partitioning between mobile and immobile phases at time T_0 (for a given k_G); K_d is the (actual) sorption coefficient as inferred from a disturbed batch sample. For small $k_G T_0$, κ is small, thus demonstrating that if an equilibrium assumption is used to interpret

results of an undisturbed sample with microscale structure and multirate sorption, K_d will be underestimated considerably. Only for sufficiently large $k_G T_0$, is κ close to 1. A slight dependence of κ versus $k_G T_0$ on K_d is observed, here computed for $K_d = 1$ and $K_d = 100$ (Figure 5). Figure 5 shows that performing a batch test on undisturbed samples and with different durations, combined with subsequent batch tests on the same disturbed samples, should provide a clear indication of multirate sorption, as well as a first estimate of k_G.

5.2 Batch-flush test

Next, we consider a combination of a batch test with flushing (or "batch-flush" test for short). After letting the batch test run for a period T_0, the tracer is flushed out by applying a flow Q_0 [L^3/T] with (zero tracer concentration) discharge; tracer concentration is then monitored in the effluent. Assuming a given kinetic model, exchange rate can be inferred based on the breakthrough of the tracer from the sample.

If exchange is governed by a multirate process, then mass balance equations yield the following system in the Laplace domain

$$s\hat{X} + \omega\hat{X} - X_0 = -\sum_{i=1}^{N} k_i\left(\varphi_i K_d \hat{X} - \hat{X}_i^*\right)$$
$$s\hat{X}_i^* - X_{0i}^* = k_i\left(\varphi_i K_d \hat{X} - \hat{X}_i^*\right) \qquad (13)$$

where X_0 and X_{0i} is the normalized mass at time T_0 as obtained from a batch test in the mobile

phase and on site type i of immobile phase, respectively, and $\omega \equiv 1/\Delta\tau$ where $\Delta\tau$ is the turnover time in the sample for the flushing. In other words, after running a batch test for $t = T_0$ as described by equation (12), the partitioning becomes an initial condition for the flushing which turns the batch into a flow reactor. Note that we have $\sum_{i=1}^{N} X_{0i} + X_0 \equiv 1$.

The solution for the normalized tracer mass in the mobile phase for a batch-flush test is obtained from equation (13) as

$$\hat{X} = \left(X_0 + \sum \frac{k_i X_{0i}^*}{s + k_i} \right) \left(s + \omega + sK_d \frac{\sum \varphi_i k_i}{s + k_i} \right)^{-1} \tag{14}$$

With ω specified, and φ_i assumed as log-normal, numerical inversion of equation (14) yields the result of the batch-flush test when exchange is controlled by a multirate model.

An example of the normalized mass recovery in the flushing phase is shown in Figure 6. The cumulative breakthrough (flushing) curves start from values of X that reflects the partitioning at time T_0 and increase monotonically to 1; for the batch phase had a duration of T_0 =1000 h in this example. Four tracers were considered with a wide ranging K_d from 0 to 1000 and two values of $\sigma^2_{\ln k}$ were considered: 0.5 (as would be computed under identical conditions as those in Figure 3) and a higher value 3, which would be obtained using the model equation (9) but assuming that porosity and the cementation exponent vary independently. The sample size for this illustration is $\ell = 30$ cm whereby $A_{\max} \approx 10$ cm. The turnover time is assumed 10 h ($\omega = 0.1$ 1/h). Thus, if we

imagine a sample of 30 cm diameter and 50 cm length, $\omega = 0.1$ 1/h would imply a volumetric flushing flow rate of approximately 0.034 m^3/day.

Even for K_d =0 it can be seen that after T_0 =1000 h, pure diffusion leaves a clear trace in the flushing, taking more than 20 h for the tracer to be completely flushed from the sample (Figure 6). For the lower log-variance (solid curves in Figure 6), the three sorbing tracers leave a weak signal up to about 500 h of flushing, after which the signal becomes more distinct as the cumulative curves start diverging (Figure 6). Clearly, the most strongly sorbing tracers would take a very long time to be flushed out of the sample; for K_d =1000 and 10,000 h of flushing only about 25% of the tracer mass has been flushed out of the sample. If the multirate variance is larger (dashed curves in Figure 6) then the different sorbing tracers leave a much more distinct signal from the onset of the flushing. Curves in Figure 6 indicate that batch-flush tests would be a useful complement to the batch tests for estimating k_G, and possibly $\sigma^2_{\ln k}$, by varying sample size, batch test duration and flushing rate. Use of multiple tracers with sufficiently large difference in sorption properties is also advantageous.

Finally, it is noted that the batch-flush test is an *ex-situ* laboratory analog for the *in-situ* single-hole injection-withdrawal (SWIW) tests [*Tsang*, 1995; *Haggerty et al.*, 2001]; the main idea with SWIW tests is to create a (spatially distributed) tracer source in the medium through injection (and possible delay until withdrawal), and then use that source for tracer transport during withdrawal. Since the SWIW tests are by definition insensitive to macroscale dispersion, the main focus is on the kinetic features of retention; the signals to be found in SWIW breakthrough

curves could therefore resemble the ones seen in Figure 6.

6. Field-scale example and potential biases

Potential biases in transport predictions caused by ignoring multirate sorption in batch experiments are explored in this section. To explore the potential biases, a numerical experiment was performed. This experiment involved a hypothetical aquifer in which sorption is presumed to be governed by the multirate model with given parameters. Fifty undisturbed samples were "collected" from the aquifer by sampling from the K_d probability distribution that quantifies large-scale variability. A numerical batch experiment was then performed for each sample and a new K_d inferred by assuming equilibrium partitioning (naive interpretation). As discussed in subsection 5.1, this estimated K_d was biased because of the neglect of kinetic effects. The K_d sample moments from the tests were then used in field-scale transport predictions, again assuming equilibrium partitioning. The result of this biased prediction was then compared with the results of the multirate (unbiased) model.

The generic statistical parameters for K_d were $K_d^G = 664$ and $\sigma^2_{\ln K_d} = 0.82$, corresponding to an arithmetic mean of 1000. The integral scales of permeability and log K_d were both assumed to be 700 m and the transport distance was set at $L = 6000$ m. This parameter set was considered "synthetic truth" for the purposes of numerical experiments and was selected to be consistent with values used in previous analyses [*Painter et al.*, 2001] of transport in the alluvial aquifer near Yucca Mountain, Nevada. The value $K_d^G = 664$ is intermediate between estimated values for

neptunium and plutonium, whereas the log-variance of 0.82 has been established for neptunium based on geochemical data [*Turner et al.*, 2002]. Two values of the mean groundwater travel times were considered: 100 yrs and 500 yrs. These two values are near the 25th and 60th percentiles of the groundwater travel time distribution estimated previously [*Painter et al.*, 2006]. For illustration, we considered a generic tracer with a decay rate of $\lambda = 0.0001$ 1/yr.

To create the biased estimates of K_d, a "synthetic truth" but hidden value of K_d was sampled from a log-normal distribution with geometric mean 664 and log-variance 0.82. Sorption was assumed to be governed by the multirate model with rate constants given by equation (10) with A in the range between $A_{max} = 0.01$ m and $A_{min} = 0.1$ m, the latter estimated as approximately 1/3 of the hypothesized sample size of 30 cm. Three numerical batch experiments with durations T_0 =1/12 yr, T_0 =1/2 yr and T_0 = 1 yr were then performed for each sample. The biased K_d was then computed as X^*/X for each batch test; the shorter the test, the smaller the inferred K_d as already indicated in Figure 5. Geometric mean and log-variance of the biased K_d are summarized in Table 1 for the three selected T_0 values.

Using the biased estimates of K_d geometric mean and variance as given in Table 1, the moments of B were computed for a given mean groundwater velocity, U. First-order expressions of *Cvetkovic et al.* [1998] were used to obtain the B_m and CV_B reported in Table 1. Using these B moments with $\tau_m = 100$ yr ($U = 60$ m/yr) or $\tau_m = 500$ yr ($U = 12$ m/yr), and CV_τ =0.567 (obtained from first order expressions [*Shapiro and Cvetkovic*, 1988]), the expected tracer discharge at a monitoring boundary at L=6000 m was computed from equation (5) with $\hat{g} = \sum_{i=1}^{n} \frac{\varphi_i k_i}{s + k_i} \rightarrow 1$. This

presumes equilibrium sorption with spatial variability in K_d and variable travel time (macrodispersion). Results of this calculation are summarized in Figure 7a.

The curves in Figure 7a (shown for T_0=1 month and T_0=1 year) indicate that batch tests on undisturbed samples of varying duration may create a bias by underestimating retention if the results are interpreted as equilibrium. Plotted in Figure 7b are the expected BTCs with multirate sorption accounted for and under the same transport conditions as in Figure 7a and using the "hidden" moments of B from Table 1. Three BTCs are shown for three values of the REV scale (1m, 4m and 10m) assuming either τ_m = 100 yr or τ_m = 500 yr.

For each REV scale in Figure 7b, the parameters of the multirate distribution were estimated using equation (9) and are summarized in Table 2. Note that the relatively low estimated values of k_G are primarily due to the high sorption coefficient; as implied by equation (9), large K_d yields long times for the saturation of the immobile zones by diffusion. With a shorter mean groundwater travel time τ_m =100 yr, Case 3 (low k_G) is of greatest concern for risk assessment because of the high peak value (Figure 7b); coincidently, this peak would be well predicted with a short batch test on undisturbed samples (T_0=1 month) for which the retention is significantly underestimated. The kinetic effects are considerably less important for Case 3 if τ_m =500 yr, with a diminished peak in spite of relatively early tracer arrival (thick dashed curve in Figure 7b). The corresponding biased estimate with τ_m =500 yr and a short batch test duration overestimates the peak discharge by more than two orders of magnitude (thick dashed curve in Figure 7a), a fact that would clearly have consequences on the estimated risk. Kinetics of multirate sorption does

not seem to affect transport for Case 2 and τ_m =500 yr (medium dashed curve in Figure 7b). Thus we may set a limit for the Damkohler number (i.e., in our case $k_G\tau_m$) of around 0.1; above 0.1, the equilibrium assumption seems applicable. However, if the K_d is inferred from batch tests of relatively short duration, then the equilibrium prediction could significantly underestimate retention.

The curves in Figure 8 may be used as a quick reference for assessing the potential impact of kinetics as governed by multirate sorption for a wide range of the key parameters τ_m, K_d and the REV scale. In Figure 8, the Damkohler number $k_G\tau_m$ was set at 0.1 as the threshold value for significant kinetic effects; τ_m was then computed from equation (9) as a function of K_d for different REV scales with only A as random. Kinetic effects depend not only on groundwater transport time, REV scale and retention material properties (diffusivity and porosity), but also on the distribution coefficient of a given tracer as clearly seen in Figure 8. For instance, neptunium (with K_d roughly 100) and plutonium (with K_d roughly 1000), would be subject to retention kinetics quite differently. With an REV scale of 10 m, neptunium may require 100 yr to reach equilibrium; for plutonium close to a 1000 yr would be required.

7. Summary and conclusions

We implemented a framework for coupling the field-scale transport effects of microscale (sub-REV) and macroscale heterogeneity. The microscale heterogeneity is conceptualized as a mixture of mobile and immobile zones; transport effects are represented by a multirate kinetic model that

accounts for the exchange (diffusion–sorption) between these zones. Macroscale heterogeneity leads to field-scale dispersion and is represented by the distributions of two random variables resolved on the REV scale: the groundwater travel time, τ, and B, which integrates the spatially variable sorption coefficient K_d along stream tubes from the injection point to a monitoring boundary. The coupling between advective transport and retention is quantified by equation (2) [*Cvetkovic and Dagan*, 1994; *Cvetkovic et al.*, 1998] using the distribution of rate parameters and the τ and B distributions.

Clearly, kinetic effects will become unimportant and transport will approach equilibrium if the transport time is sufficiently long. The analysis presented here explored the approach to equilibrium quantitatively. The relative importance of kinetic effects is determined by a Damkohler number, which is defined as the product of the mean travel time and geometric-mean rate constant. When the Damkohler number is greater than about 0.1, equilibrium sorption is an adequate approximation.

The conventional view of sorption kinetics is that the kinetics may be important for laboratory experiments or short-duration field tests, but not for field-scale transport (because of the long times involved). Our model for the multirate distribution suggests that the geometric mean rate may be rather small for strongly sorbing species. This small rate implies that the time required to reach equilibrium may be longer than generally recognized if the species of interest is strongly sorbing. It can be seen from the curves in Figure 8, for example, that kinetic effects may be important for travel times as long as 60 years when the dimensionless distribution coefficient K_d

is 1000 and the REV scale is 1 meter. Note, however, that the weakly to moderately sorbing species that are generally of greater risk significance in nuclear waste repository studies approximate equilibrium after much shorter travel times.

Laboratory experiments using samples with internal structures left intact (i.e. "undisturbed") and multiple tracers may be used to establish kinetic effects due to diffusion and sorption. In particular, a batch-flush test with varying sample size and flushing rates on undisturbed samples may help assess possible kinetic effects. Such tests are *ex-situ* analogs of the *in-situ* single-well injection-withdrawal tests. However, extrapolation of both short-duration *in-situ* tests and their *ex-situ* analogs should both done with care. Both types of tests may significantly underestimate K_d if an equilibrium interpretation is used. Augmenting these tests with conventional batch tests on repacked samples to obtain K_d would avoid that bias. In addition, *ex-situ* tests suffer from potential biases associated with the fact that the limited sample size may truncate the distribution of rates. Moreover, that the approach to equilibrium is strongly dependent on K_d should be considered in efforts to extrapolate short-duration tests using weakly sorbing tracers to the field scale.

The goal of this work was to explore the interplay between microscale and macroscale heterogeneity in the context of field-scale transport. To that end, several plausible but generic approximations were adopted. The computed values of CV_τ and CV_B were based on first-order approximations of the flow equation [*Cvetkovic et al.*, 1998], which may underestimate the variability. In a site-specific application, numerical simulations conditioned on all available

hydrogeological information would be useful for further constraining CV_τ and CV_B. The method for inferring field scale values of K_d based on geochemical data [*Turner et al.*, 2002] is particularly useful for obtaining a broader geostatistical database for K_d. Analysis of microscale structures from outcrop or borehole data may be useful to constrain the sub-REV internal structure (porosity, size variations, diffusion) and thus the diffusion controlled retention models.

Numerical simulations using different heterogeneous structures can play a significant role in improving understanding of flow and advective transport in highly heterogeneous formations on a wide range of scales. In particular, the work of *Fiori et al.* [2006, 2007] could be extended to address the interplay between advection and diffusion into low permeability zones. An important and yet unresolved issue is the consequence of multirate sorption for higher-order moments of the tracer discharge, in particular the variance. Extending the results of *Fiori et al.* [2002] to include multirate sorption is one possible avenue for exploring that issue.

SP 1/18/2010

Appendix A: Mass balance formulation

The mass balance equations for a non-decaying tracer are

$$\frac{\partial C}{\partial t} = -\nabla \cdot \mathbf{J} - \frac{\partial}{\partial t}\sum_{i=1}^{N} C_i^* \qquad (15)$$

$$\frac{\partial C_i^*}{\partial t} = k_i\left[\Phi_i(x)C - C_i^*\right]$$

where C [M/L^3] is the tracer concentration in the mobile phase averaged over the REV, C^* [M/L^3] is total tracer concentration in the REV immobilized on the porous matrix and the immobile water, and $\mathbf{J}$ [M/TL^2] is a tracer mass flux vector. The quantities C_i^* [M/L^3], k_i [1/T] and Φ_i are immobilized tracer concentration, associated mass transfer rates and relative weighting, respectively, for site type i. We have

$$\sum_{i=1}^{N}\Phi_i \equiv K_d(\mathbf{x}); \quad \Phi_i \equiv K_d(\mathbf{x})\varphi_i \qquad (16)$$

where K_d [-] is the spatially variable dimensionless sorption coefficient for the REV. Focusing on advection only, we have $\mathbf{J} = \mathbf{V}C$ where $\mathbf{V}$ [L/T] is the spatially variable fulid velocity vector at $\mathbf{x}$. A corresponding Lagrangian mass balance system along a stream tube is [*Cvetkovic and Dagan*, 1994]

$$\frac{\partial X}{\partial t} = -\frac{\partial X}{\partial \tau} - \sum_{i=1}^{N} k_i\left[\varphi_i K_d(\tau)X - X_i^*\right] \qquad (17)$$

$$\frac{\partial X_i^*}{\partial t} = k_i\left[\varphi_i K_d(\tau)X - X_i^*\right]$$

where X is the normalized tracer discharge [1/T] and $K_d(\tau) \equiv K_d\left[\mathbf{Z}(\tau)\right]$. Alternatively, we can write [*Carrera et al.*, 1998]

$$\frac{\partial X}{\partial t} + \frac{\partial X}{\partial \tau} = -K_d(\tau)\left[g(t) * \frac{\partial X}{\partial t}\right] \tag{18}$$

where $\hat{g}$ is defined in equation (4). The solution in the Laplace domain is

$$\hat{X} = \exp\left[-s\left(\tau + B\sum_{i=1}^{N}\frac{\varphi_i k_i}{s+k_i}\right)\right] \equiv \hat{\gamma} \tag{19}$$

where B is defined in equation (1) and γ in equation (3). Thus, γ and X are equivalent where g was introduced in Section 3.

Appendix B: Multirate and spherical models

Consider a spherical diffusion-sorption model, and let K_G' and $\sigma_{\ln k}^2$ denote the geometric mean and log-variance of D/RA^2, respectively, where D is the diffusion coefficient, R is the

retardation coefficient in the spheres and A is the sphere diameter. Then it can be shown that the equivalent multirate mode is defined by the following distribution of rates (*Haggerty et al.*, 2000)

$$\varphi(k) = \sum_{j=1}^{\infty} \frac{8}{\sqrt{2\pi^5}(2j-1)^2 \sigma_{\ln k}^2 k} \exp\left\{ -\frac{1}{2}\sigma_{\ln k}^{-2}\left[\ln \frac{4k}{\pi^2 (2j-1)^2 k_G}\right]^2 \right\} \tag{20}$$

RB 1/18/2010

References

Carrera, J., X. Sanchez-Vila, I. Benet, A. Medina, G. Galarza, and J. Guimera (1998). On matrix diffusion: formulations, solutions and qualitative effects. Hydrogeology Journal, 6,178190.

Clennell, M. (1997). Tortuosity: A guide through the maze. Geological Society Special Publication 122, 299-344.

Cvetkovic, V., and G. Dagan (1994), Transport of kinetically sorbing solute by steady random velocity in heterogeneous porous formations, J. Fluid Mech., 265 , 189–215.

Cvetkovic, V., Dagan, G., and Cheng, H. (1998). Contaminant transport in aquifers with spatially variable hydraulic and sorption properties. Proc. Roy. Soc. Lond. A, 454, 2173–2207, 1998.

Cvetkovic, V. and Haggerty, R. (2002). Transport with exchange in disordered media. Phys. Rev. E, 65:051308–1–9.

Cvetkovic, V., S. Painter, D. Pickett, D. Turner and P. Bertetti 2004 Parameter and model sensitivities for colloid facilitated radionuclide transport on the field scale. Water Resour. Res., 40, W06504, doi:10.1029/ 2004WR00WR003048.

Cvetkovic, V., H. Cheng, H. Widestrand, J. Byegård, A. Winberg, P. Andersson (2007), Sorbing

tracer experiments in a crystalline rock fracture at Äspö (Sweden): 2. Transport model and effective parameter estimation, Water Resour. Res., 43, W10413, doi:10.1029/2006WR005278.

Dagan, G. (1984). Solute transport in heterogeneous porous formations. J. Fluid Mech., 145, 151–177.

Fiori, A., S. Berglund, V. Cvetkovic, and G. Dagan (2002), A first-order analysis of solute flux statistics in aquifers: The combined effect of pore-scale dispersion, sampling and linear sorption kinetics. Water Resour. Res., 38 , 1137, 10.1029/2001WR000678.

Fiori, A., I. Jankovic, and G. Dagan (2006), Modeling flow and transport in highly heterogeneous three-dimensional aquifers: Ergodicity, Gaussianity, and anomalous behavior –2: Approximate semianalytical solution, Water Resour. Res., 42, W06D13, doi:10.1029/2005WR004752.

Fiori, A., I. Jankovic, G. Dagan, and V. Cvetkovic (2007), Ergodic transport through aquifers of non-Gaussian log conductivity distribution and occurrence of anomalous behavior, Water Resour. Res., in press.

Haggerty, R., and S. M. Gorelick (1995), Multiple-rate mass transfer for modeling diffusion and surface reactions in media with pore-scale heterogeneity, Water Resour. Res., 31(10), 2383-2400.

Haggerty, R., S. A. McKenna, and L. C. Meigs (2000), On the late-time behavior of tracer test

breakthrough curves, Water Resour. Res., 36(12), 3467-3479.

Haggerty, R., S. W. Fleming, L. C. Meigs, S. A. McKenna (2001), Tracer tests in a fractured dolomite: 2. Analysis of mass transfer in single-well injection-withdrawal tests, Water Resour. Res., 37(5), 1129-1142, 10.1029/2000WR900334.

Kreft, A. and Zuber, A. (1978). On the physical meaning of the dispersion equation and its solutions for different initial and boundary conditions. Chem. Eng. Sci., 33, 1471–1480.

McKenna, S. A., L. C. Meigs, and R. Haggerty (2001), Tracer tests in a fractured dolomite, 3., Double porosity, multiple-rate mass transfer processes in two-well convergent-flow tests. Water Resources Research, 37(5), 1143-1154.

Meigs, L.M, R. L. Beauheim (2001), Tracer tests in a fractured dolomite 1. Experimental design and observed tracer recoveries, Water Resour. Res., 37(5), doi:10.1029/2000WR900335.

Mohanty, S., and T. McCartin (2001), NRC sensitivity and uncertainty analyses for a proposed HLW repository at Yucca Mountain, Nevada, using TPA 3.1: Conceptual models and data, Tech. Rep. NUREG-1668, Nuclear Regulatory Commission.

Neretnieks, I. (1980), Diffusion in the rock matrix: An important factor in radionuclide retardation, J. Geophys. Res., 85, 4379 – 4397.

Painter, S., D. Turner, and V. Cvetkovic (2001), Radionuclide retardation in the alluvial aquifer near Yucca Mountain, Nevada, Ground Water, 37, 326–338.

Reimus P., G. Pohll, T. Mihevc, J. Chapman, M. Haga, B. Lyles, S. Kosinski, R. Niswonger, P. Sanders (2003), Testing and parameterizing a conceptual model for solute transport in a fractured granite using multiple tracers in a forced‐gradient test, Water Resour. Res., 39 (12), 1356, doi:10.1029/2002WR001597.

Reimus P. W., T. J. Callahan (2007), Matrix diffusion rates in fractured volcanic rocks at the Nevada Test Site: Evidence for a dominant influence of effective fracture apertures, Water Resour. Res., 43, W07421, doi:10.1029/2006WR005746.

Salem, H.S. (2001), Determination of porosity, formation resistivity factor, Archie cementation factor, and pore geometry factor for a glacial aquifer, Energy Sources, 23, 589-596.

Scher, H., G. Margolin and B. Berkowitz (2002). Towards a unified framework for anomalous transport in heterogeneous media, Chemical Physics, 284, 349-359.

Shapiro, A., and V. Cvetkovic (1988), Stochastic analysis of solute arrival time in heterogeneous porous media, Water Resour. Res., 24, 1711–1718.

Tsang, Y. (1995), Study of alternative tracer tests in characterizing transport in fractured rock, Gephys. Rev. Lett., 22, 1421-1424.

Turner, D., F. Bertetti, and R. Pabalan (2002). The role of radionuclide sorption in high-level waste performance assessment: Approaches for the abstraction of detailed models, in Soil Science Society of America Proceedings Volume: Soil Geochemical Processes of Radionuclides , pp. 211–252.

Widestrand, H., J. Byegård, V. Cvetkovic, E.-L. Tullborg, A. Winberg, P. Andersson, and M. Siitari-Kauppi (2007), Sorbing tracer experiments in a crystalline rock fracture at Äspö (Sweden): 1. Experimental setup and microscale characterization of retention properties, Water Resour. Res., 43, W10413, doi:10.1029/2006WR005277.

SCP 1/18/2010

Table 1: Biased estimates of K_d and B moments using batch tests of varying duration T_0. "Actual" denotes assumed true values for the considered aquifer. Note that the log-variance of K_d decreases with increasing T_0 because the immobilized mass as a function of T_0 exhibits a steep gradient around T_0=1 month and a comparatively mild gradient for T_0=1/2 yr and T_0=1 yr. First-order expressions for τ and B moments are given e.g., in Cvetkovic et al. [1998].

	Batch duration time T_0	K_d^G	$\sigma^2_{\ln K_d}$	B_m (τ_m=100 yr)	B_m (τ_m =500 yr)	CV_B
Biased estimates	1 month	10.9	3.13	5 213	26 066	2.1
	½ year	197	0.8	29 383	146 945	0.84
	1 year	238	0.51	30 713	153 565	0.74
Actual values	-	664	0.82	100 000	500 000	0.85

Table 2: The multirate parameters obtained by sampling (9) that are used in Figure 7b, given for different REV scales.

Case	REV scale [m]	k_G [1/yr]	σ^2_{lnk}
1	1	0.00162	1.47
2	4	0.00014	2.67
3	10	0.000024	3.17

SP 1/18/2010

Figure captions

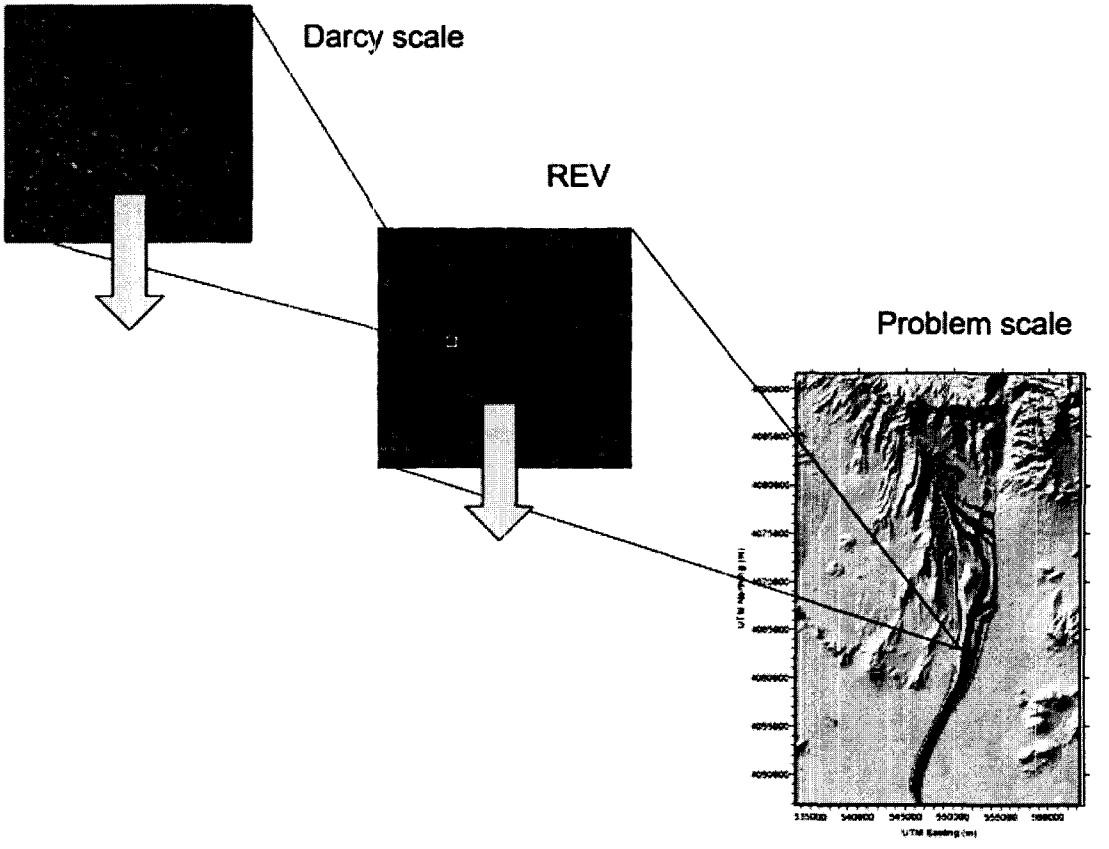


Figure 1: Configuration sketch of the problem scale, the underlying REV and the Darcy-scale. The flow is resolved on the REV scale, which can be on the order of meters or larger. The REV is internally heterogeneous with contrasts of high and low permeability–porosity zones constituting mobile and immobile regions. The problem scale illustration of advection flow paths is for the alluvial aquifer at Yucca Mountain obtained from numerical simulations.

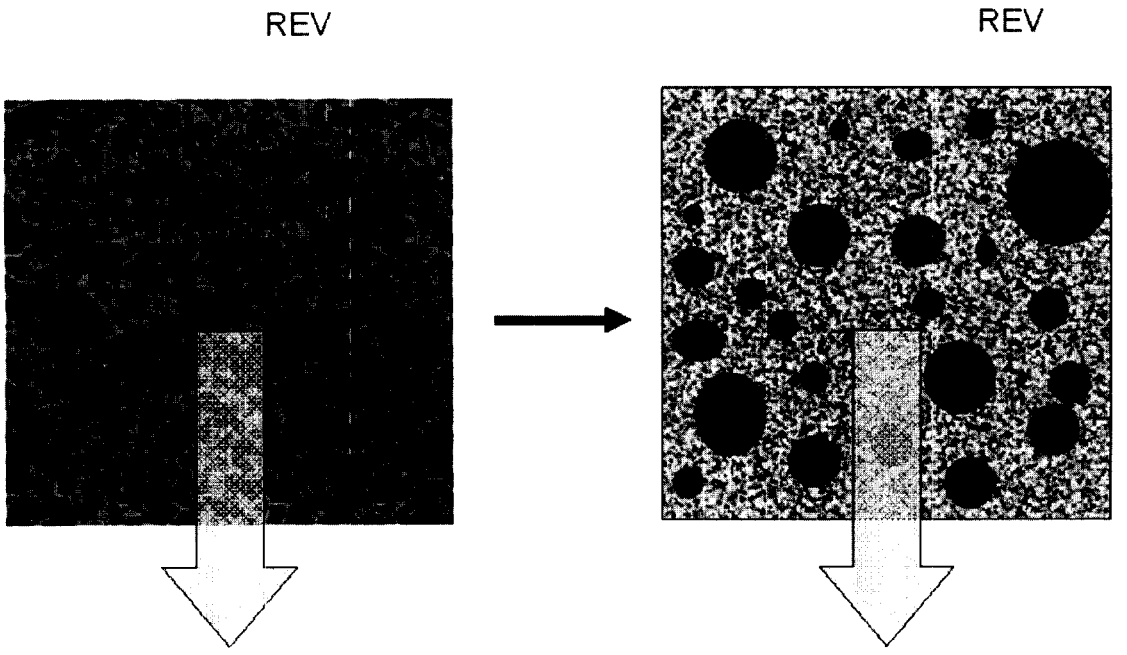


Figure 3: Conceptual model of the mobile-immobile zones within an REV; the internal heterogeneous structure is simplified as spheres of varying diameter.

SB 1/18/2010

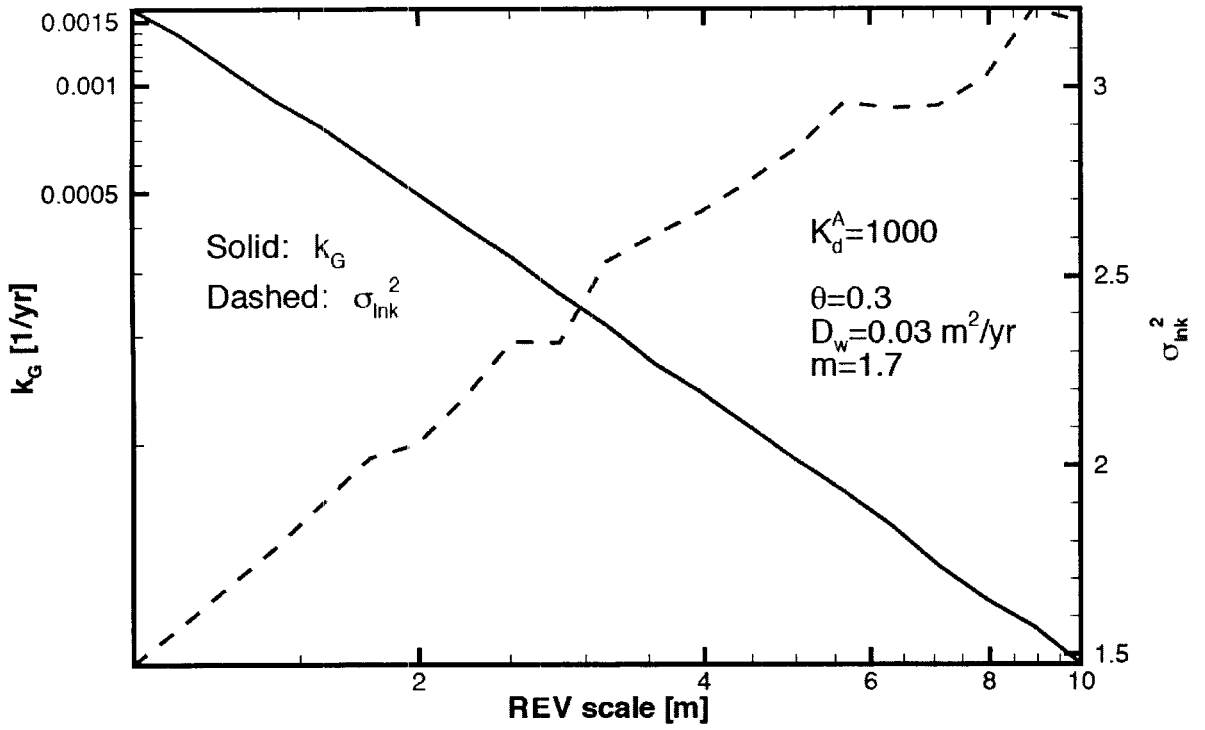


Figure 4: Global estimates of geometric mean and log-variance of kinetic rates as functions of the REV scale for a strongly sorbing tracer (the arithmetic mean of K_d is 1000).

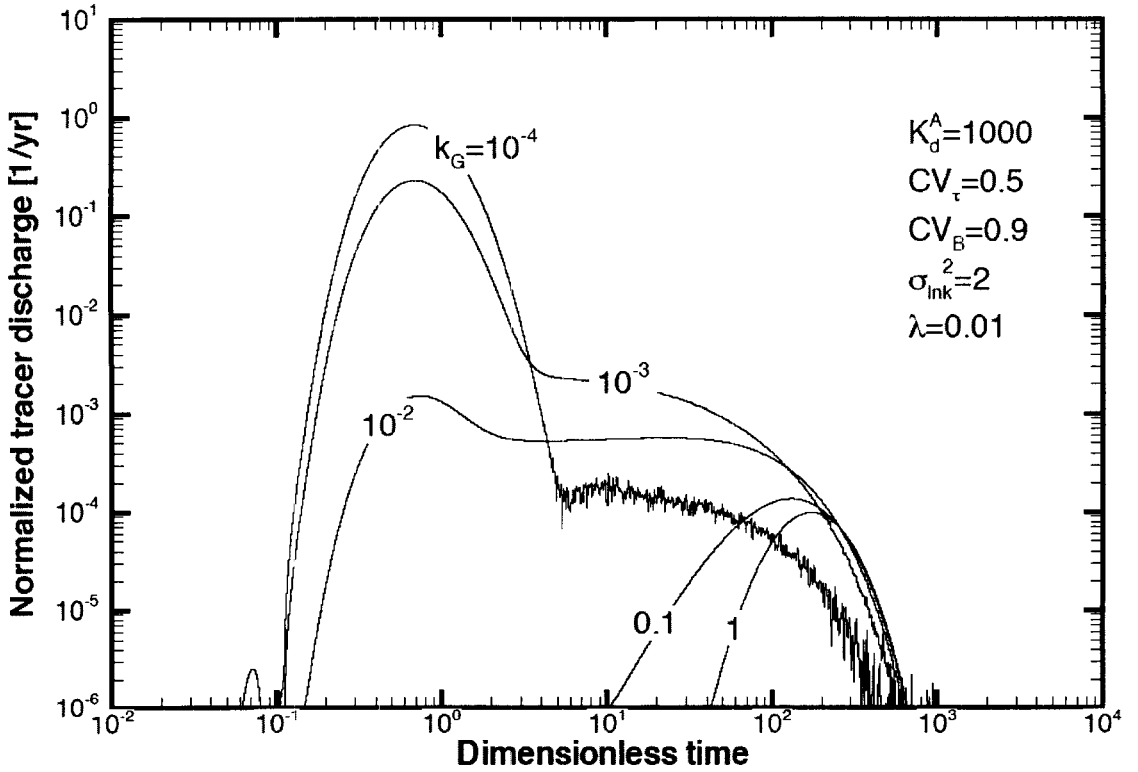


Figure 2: Expected tracer discharge across a monitoring boundary with varying Damkohler number for a strongly sorbing tracer.

ACS 1/18/2010

SN 318E Vol. 13 Pg. 48, Scott Painter

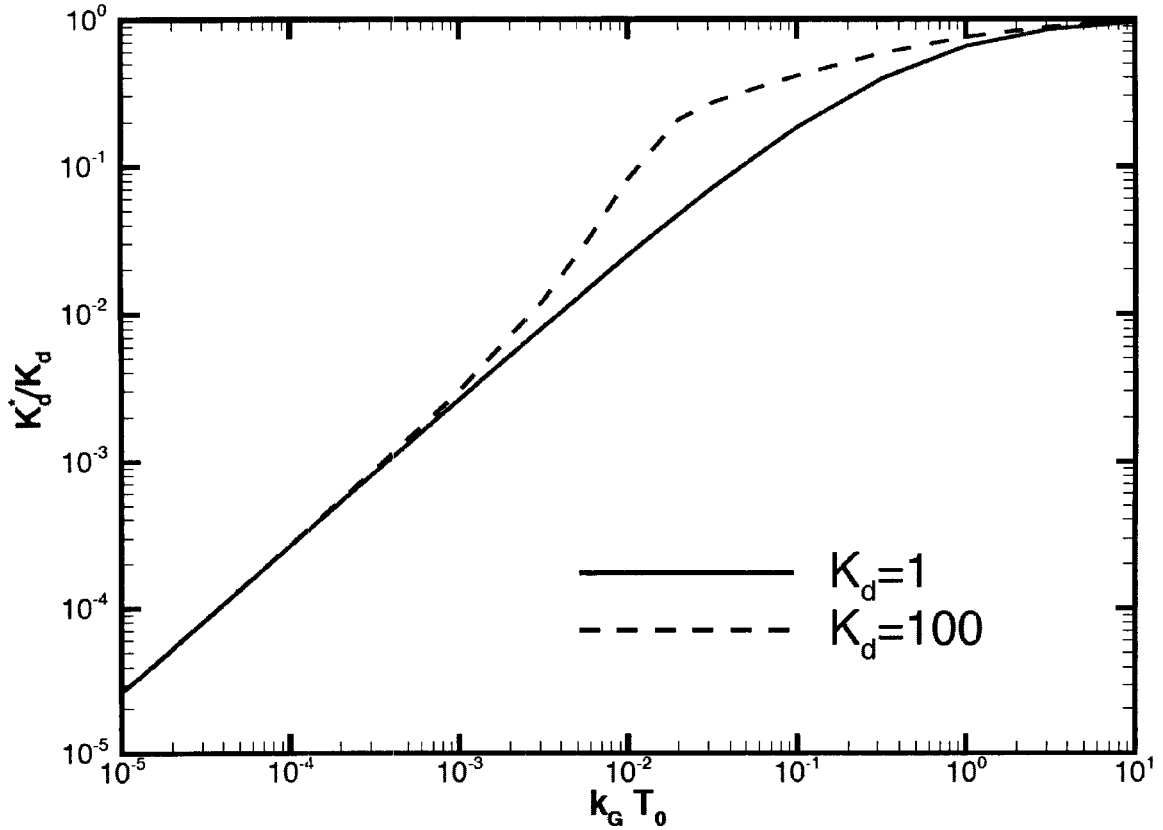


Figure 5: Naive (equilibrium) interpretation of batch tests of varying durations for two tracers. The ratio $\kappa \equiv K_d^* / K_d$ is illustrated as a function of $k_G T_0$, where T_0 is the test duration and $K_d^* = \sum_i X_i^* / X$ is the estimated sorption coefficient assuming equilibrium.

1/18/2010

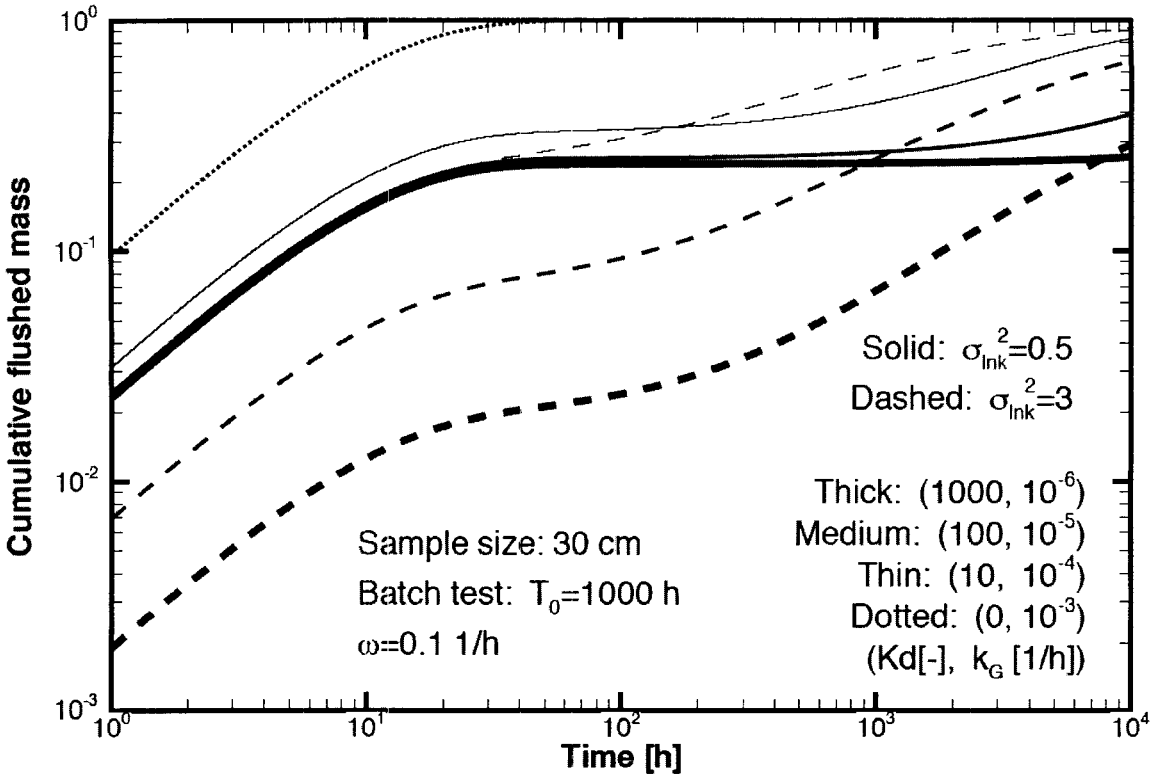
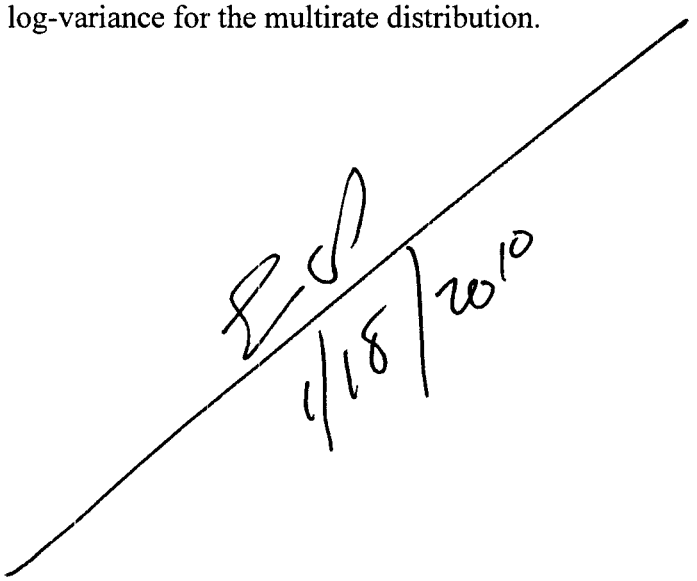
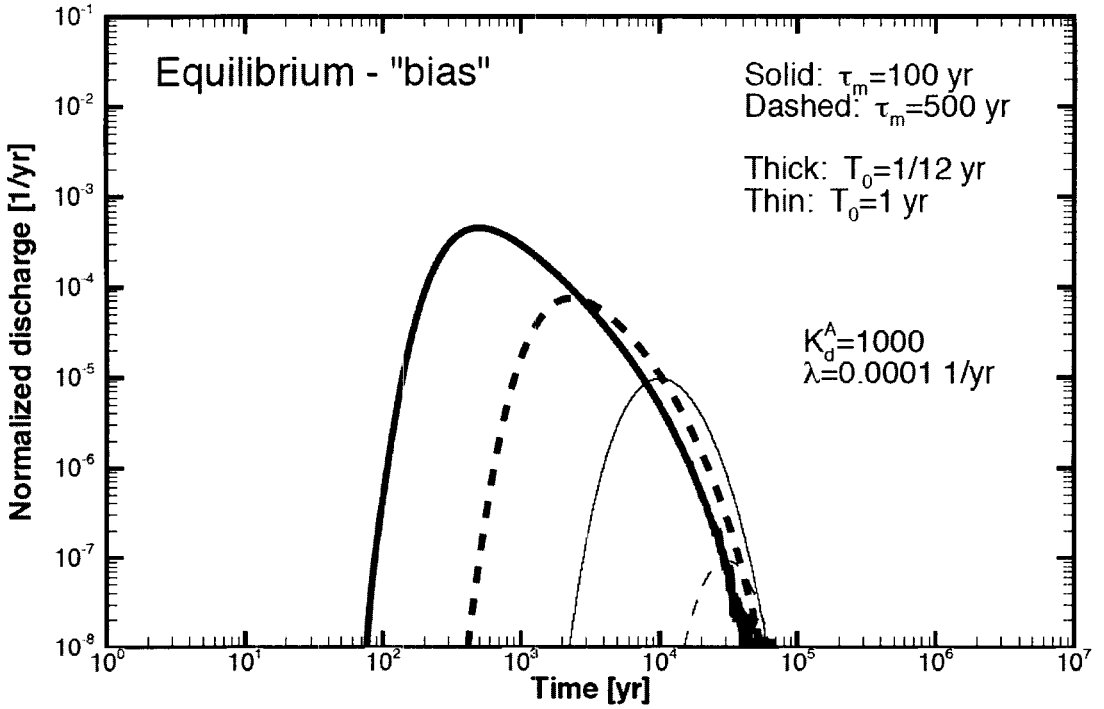


Figure 6: Cumulative flushed tracer mass in a batch-flush test; the batch test is run for 1000h and then the tracer is flushed with a turnover time of 10h. The cumulative mass is shown for four tracers with strongly varying sorption properties, with K_d between 0 and 1000. Also, we show results for two values of the log-variance for the multirate distribution.



a)


Equilibrium - "bias"
Solid: τ_m=100 yr
Dashed: τ_m=500 yr
Thick: T_0=1/12 yr
Thin: T_0=1 yr
K_d^A=1000
λ=0.0001 1/yr
Normalized discharge [1/yr]
Time [yr]

b)

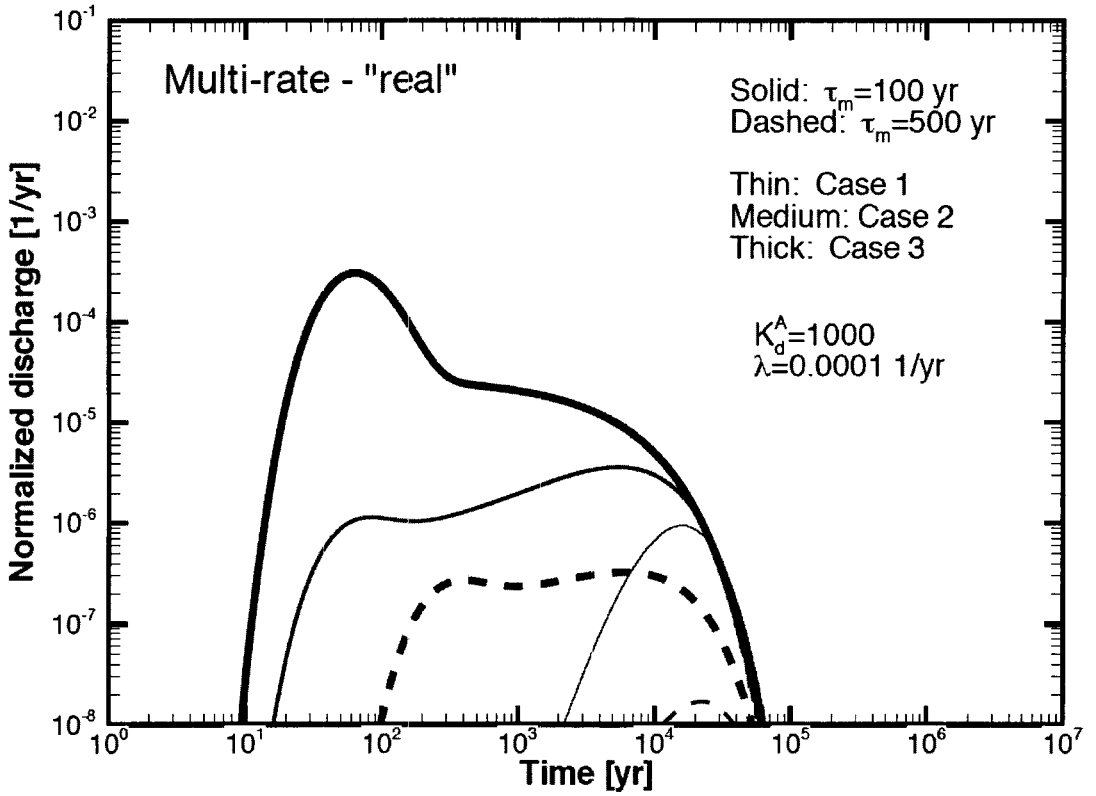

Multi-rate - "real"
Solid: τ_m=100 yr
Dashed: τ_m=500 yr
Thin: Case 1
Medium: Case 2
Thick: Case 3
K_d^A=1000
λ=0.0001 1/yr
Normalized discharge [1/yr]
Time [yr]

Figure 7: Expected tracer discharge across a monitoring boundary: a) "Biased" predictions assuming equilibrium sorption, based on biased estimates of the sorption coefficient K_d for different batch test durations, as summarized in Table 1; b) "synthetic truth" predictions for different REV scales accounting for multirate sorption; the corresponding global estimates of the multirate parameters are given in Table 2. In both cases, we compare the BTCs for mean groundwater travel times of 100 yr and 500 yr.

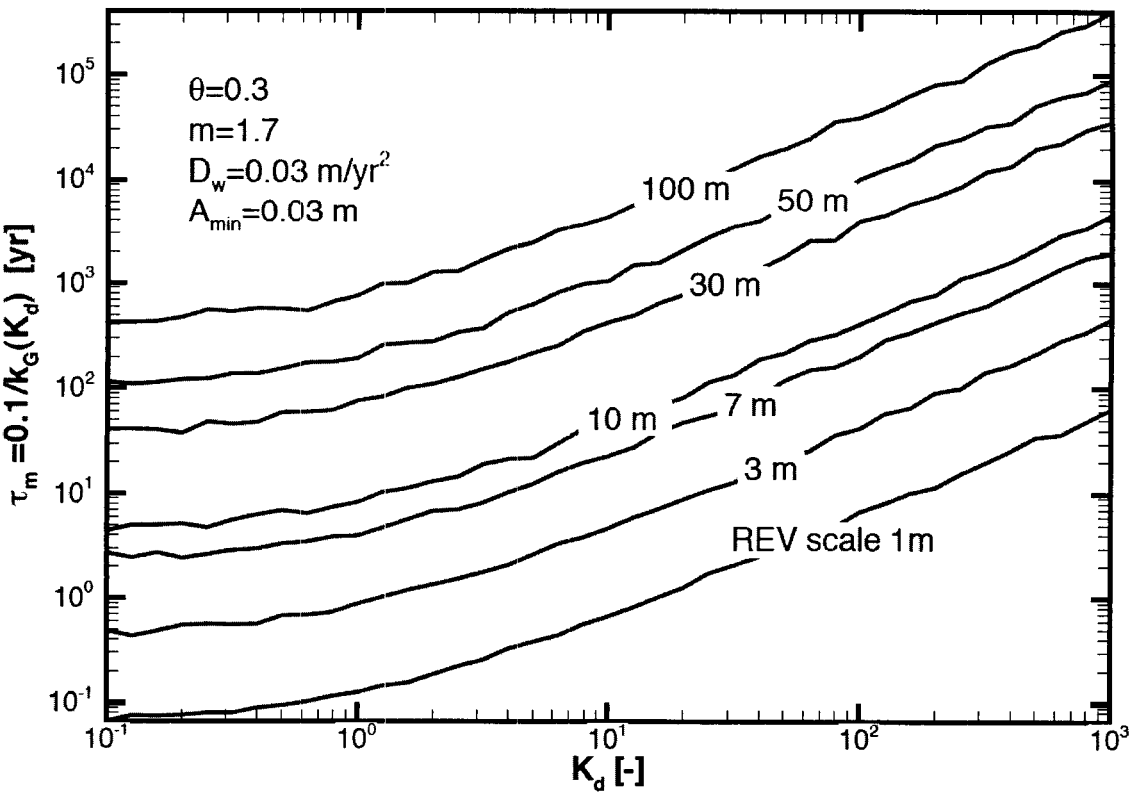


Figure 8: Type curves obtained by setting the Damkohler limit for kinetic effects as 0.1. These curves can be used for a first assessment whether equilibrium assumption is appropriate, for a given mean groundwater travel, sorption coefficient and REV scale. The curves are applicable for the specified set of physical parameters but can be easily generated for any other

set.

11/30/07 Calculations for previous entry

Calculations for the previous entry and results of those calculations are documented in Mathematica notebooks (attached). The notebooks are named by Figure (i.e. *Figure 2.nb*, etc.) and are self documenting.

The notebooks *Kd_NAIVE.nb*, *MR for sample 30 cm.nb* contain some calculations used to generate Table 1.

28 1/18/2010

Entries into Scientific Notebook No. 318E for the period 5.08.06 to 11.30.07 were made by Scott Painter. No original text entered into this scientific notebook has been removed.

Scott Painter
1-18-10

RS 1/18/2010

SCIENTIFIC NOTEBOOK

by

Scott Painter

Southwest Research Institute
Center for Nuclear Waste Regulatory Analyses
San Antonio, Texas

September 22, 2009

Table of Contents

INITIAL ENTRIES 3
Radionuclide Transport Analysis for Yucca Mountain 4
Objective: 4
5.08.06 Alternative distribution for plutonium solubility limit 5
6.5.06 Repeat of previous entry ..... **Error! Bookmark not defined.**
6.27.06 Relative affinity factors in colloid-facilitated release abstraction . **Error! Bookmark not defined.**

INITIAL ENTRIES

Scientific notebook: #318E Vol. 14
Issued to: S. L. Painter
Original Issue Date: 22-March-1999

By agreement with CNWRA QA this notebook will be printed at approximate quarterly intervals. This computerized electronic notebook is intended to address the criteria of CNWRA QAP-001.

Radionuclide Transport Analysis for Yucca Mountain

Account number: 20-06002.01.241 and 242
Description: Radionuclide Transport Analysis.
Collaborators: Drs. D. Pickett and H. Basagaoglu

Objective:
Confirmatory analyses for radionuclide transport in engineered barrier and natural systems.

RCP 1/18/2010

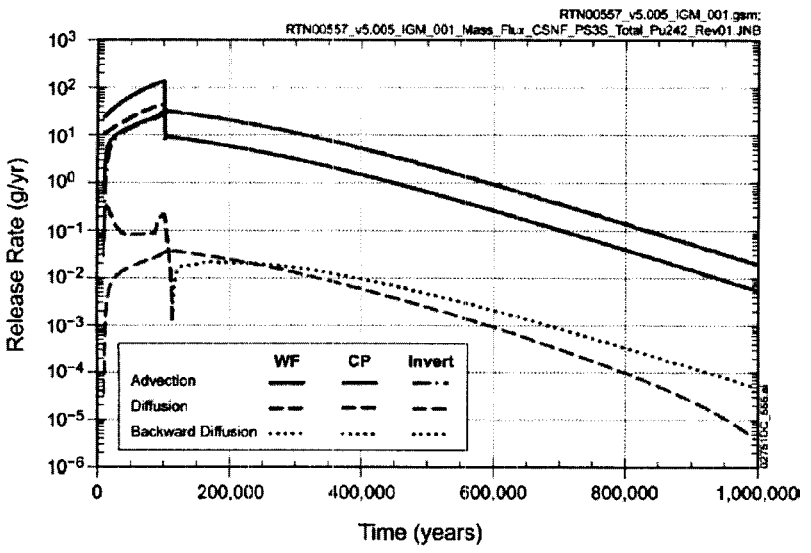
SN 318E Vol. 14 Pg. 4, Scott Painter

9.22.09 Confirmatory calculation: EBS releases of Pu-242 for Igneous Intrusion Modeling Case

This entry describes a confirmatory calculation of Pu-242 release for the igneous intrusion modeling case. In response to RAI: 3.2.2.1.3.4-3-001, DOE provided information on Pu-242 releases from the EBS system for realization #525, which uses epistemic vector #53. According to the RAI response, the realization is representative of Pu behavior in the EBS. Specifically, plutonium-rich zirconium oxide (waste form) colloid suspensions are unstable in the waste form domain and iron oxyhydroxide (corrosion product) colloid suspensions are stable in the corrosion products domain, a combination of conditions that occurs in about 62% of realizations. The results are presented for commercial SNF waste packages in the seeping environment of percolation subregion 3 following an igneous intrusion event that occurs at 10,000 year. As can be seen in Figure 2 of the RAI response, which is reproduced below, peak release of Pu-242 is limited by processes in the corrosion product domain. This confirmatory calculation, thus focuses on the corrosion product domain. Peak release is approximately 30 g/yr from 2369 failed waste packages in the seeping environment.

ENCLOSURE 1

Response Tracking Number: 00557-00-00 RAI: 3.2.2.1.3.4-3-001



NOTE: Invert advection overlays corrosion product (CP) advection, and waste form (WF) advection overlays WF backward diffusion (after 100,000 years).

Figure 2. Release Rate of Total ^{242}Pu from Each EBS Domain from Commercial SNF Waste Packages in a Seeping Environment of Percolation Subregion 3 for Realization #525 of the Igneous Intrusion Modeling Case for 1,000,000 Years after Repository Closure

For the conditions of this realization, releases are controlled by advection and sorption onto stationary corrosion products. For DOE's mixing cell model of the corrosion product domain, the rate of mass release $\frac{dM}{dt}$ [g/yr] is given by

$$\frac{dM}{dt} = V\theta R\frac{dC}{dt} = -\frac{V}{\tau}C$$

where M is total radionuclide mass in the cell (dissolved plus sorbed to stationary corrosion products, neglecting mass sorbed to mobile colloids), C [g/m^3] is dissolved concentration of radionuclide, V [m^3] is volume of the cell, θ [-] is porosity, R [-] is retardation factor, and τ is water residence time (inverse of volumetric flux rate per unit cell volume). From the equation above, we have

$$\frac{dM}{dt} = -\frac{M}{\tau\theta R}$$

DOE does not provide water residence time for realization #525. However, average water residence time (bulk volume basis) for the igneous intrusion case is given in DOE's response to RAI 3.2.2.1.3.4-2-009 as approximately 14 years.

As for the retardation factor, note that it is approximately the dimensionless distribution coefficient when the dimensionless distribution coefficient is much greater than 1. The dimensionless distribution coefficient can be estimated from Figure 7 of DOE's response to 3.2.2.1.3.4-3-001as the ratio of sorbed to dissolved mass. Using values at the time of the peak release from that figure, R is approximately 8 10^6 g/ (2.5 10^2 g) = 32000. The porosity in the corrosion products domain is 0.4 (c.f. ANL–WIS–PA–000001 REV 03, pg 8-11). At the time of the peak release, the release rate is thus
(8 10^6 g)/(14 yr * 0.4 * 32000) = 45 g/yr, compared with DOE's value of 30 g/yr.

10.8.2009 Confirmatory calculations of EBS releases for Np-237 and Pu-242

Confirmatory calculations of peak-expected release of Np-237 and Pu-242 for nominal, seismic ground motion and igneous intrusions modeling cases are documented in this entry.

The approximation

Based information provided in the SAR and in responses to RAI's, it is apparent that peak releases of Pu-242 and Np-237 are controlled by processes in the corrosion product domain and that advective releases from CSNF packages in seeping conditions dominate releases. Based on considerations of solubility limits, it is further assumed that Pu-242 is solubility limited in the

corrosion product domain. However, Np-237 concentrations are limited by strong sorption onto corrosion products.

Advective releases can be written as

$rr = N_{fp}\, qC$

where rr is radionuclide release rate in units of g/year, N_{fp} is number of failed packages experiencing flow in a seepage environment, q is water flow rate per package in units of liter/year, and C is concentration in g/liter. For the situation where concentrations are limited by sorption and not solubility,

$C = M / (\theta V R)$

Here M is radionuclide inventory in one waste package (g), V is volume of the package, θ is porosity, and R is retardation factor. The retardation factor can be written in the usual way as

$$R = 1 + \frac{K_d \rho_b}{\theta} \approx \frac{K_d \rho_b}{\theta}$$

where K_d is distribution coefficient [volume/mass], ρ_b is bulk density [mass/volume] of the corrosion products. Noting that $V \rho_b$ is the total mass of corrosion products M_{cp} in one package,

$rr = (N_{fp}\, q\, M) / (K_d\, M_{cp})$

For the situation where concentrations are limited by solubility, the release rate is simply

$rr = N_{fp}\, q\, C_{sl}$

where C_{sl} is the solubility limited concentration.

Flow rate in package

An estimate of the flow rate q is needed for each modeling case. Seepage rates are provided in RAI response 3.2.2.1.3.6-005 as follows

0.1 m3/yr = 100 liter/yr : nominal
0.43 m3/yr = 430 liter/yr : seismic ground motion
0.89 m3/yr = 890 liter/yr : igneous intrusion

These numbers are seepage flow rate, not flow through the waste package. In the TSPA AMR, MDL-WIS-PA-000005, pg 6.3.6-5, it is noted that all seepage passes through the waste package when 4% of the waste package area is breached by corrosion. Moreover, the scaling with breached area is linear. From SAR Figure 2.1-16 it is estimated that approximately 0.22% of waste package area is breached at 1 million years for the nominal and SGM modeling cases. Thus, the fraction of

seepage flow that passes through the waste package is estimate as 0.22/4 = 0.055 for theses cases. The igneous modeling case assumes 100% failure, so the flow rate q in the waste package is the same as the seepage rate in that case. Thus the flow rate q for the three modeling cases is

q = 5.5 liter/year : nominal
q = 23.6 liter/year : SGM
q = 890 liter/year : igneous

Number of waste packages experiencing flow

The number of waste packages experiencing flow due to seepage is needed. From SAR Table 2.1-7, 40% of waste packages experience seepage in the nominal modeling case. From Table 2.1-9, 69% experiences seepage in the SGM case. 100% experience seepage in the igneous case. From Table 6.3.7-1 of MDL-WIS-PA-000005, there are 8213 CSNF+naval packages. From SAR Figure 2.1-12 approximately 55% of CSNF packages are breached at 1 million years (seismic and nominal scenarios). For igneous, the number is 100% breached by the event.

Thus, the number of packages experiencing flow in a seepage environment can be calculated as

8213 * 0.55 * 0.40 = 1807 : nominal
8213 * 0.55 * 0.69 = 3117 : seismic
8213 * 1.0 * 1.0 * 0.017 = 140 : igneous

Note that for the igneous case, the probability of the event (1.7% in 1 million years) is used to get the expected value of waste packages experiencing seepage.

Inventory

From Table 6.3.7-5 of MDL-WIS-PA-000005, TSPA uses a value of 5380 g for initial Np-237 mass in one waste package. Np-237 is enhanced by the short-lived Am-241 (10200 g). Decaying all Am-241 to Np-237, the early inventory of Np-237 is 15580 g. This amount will decay to 11269 g in one million years (using a half life of 2.14 10^6 years for Np-237).

For Pu-242, the inventory modeled in TSPA is 5460. g/pkg

Mass of corrosion products

From Table 6.3-8 of EBS Radionuclide Transport Abstraction AMR, there are 2.46 10^7 g of steel corrosion products in one CSNF package.

Distribution coefficients

The applicant provided probability distributions for distribution coefficients (Kd's) (RAI: 3.2.2.1.3.4-3-003).

Mean value for neptunium is $10^{4.267}$ = 18493. ml/g = 18.493 l/g

Mean value for plutonium is $10^{3.384}$ = 2421. ml/g = 2.421 l/g

Solubility Limits

For Pu-242, solubility limit is about 0.01 mg/l = 10^{-5} g/l (ANL-WIS-MD-000010 Rev 6, Table 6.5-1).

Release rates

$rr = (N_{fp}\ q\ M) / (K_d\ M_{cp})$ when sorption limited

For Np-237:

rr = (1807 5.5 15580)/(18.493 2.46 10^7) = 0.34 g/yr : nominal case
rr = (3117 23.6 15580)/(18.493 2.46 10^7) = 2.5 g/yr : SGM case
rr = (140 890 15580)/(18.493 2.46 10^7) = 4.3 g/yr : igneous case

For Pu-242 (assuming sorption limited):

rr = (1807 5.5 5460)/(2.421 2.46 10^7) = 0.91 g/yr : nominal case
rr = (3117 23.6 5460)/(2.421 2.46 10^7) = 6.7 g/yr : SGM case
rr = (140 890 5460)/(2.421 2.46 10^7) = 11.4 g/yr : igneous case

For Pu-242 (assuming solubility limited)

rr = (1807 5.5 10^{-5}) = 0.10 g/yr : nominal case
rr = (3117 23.6 10^{-5}) = 0.73 g/yr : SGM case
rr = (140 890 10^{-5}) = 1.2 g/yr : igneous case

As the solubility limited release is smaller, release is solubility controlled for Pu-242.

10.23.2009 Confirmatory calculations of EBS releases for Tc-99

Now consider confirmatory calculations of EBS releases for Tc-99.]

The approximation for the high solubility, non-sorbing radioelements is that a single waste package failure causes the entire inventory for that package to be released in a brief pulse. Mean releases are obtained by summing a large number of such pulses with random times of occurrence. In this approximation, the fractional release of Tc-99 is just the rate at which packages fail.

We start with the seismic ground motion case.

SAR Figure 2.1-24 shows that mean releases of Tc-99 from EBS in the SGM modeling case occur relatively early, with most activity released before 100,000 years. As few of the CSNF waste packages have failed by that time, these releases must be caused by codisposal waste packages. From SAR Figure 2.1-12c, we do see that CDSP packages fail early. For example, about 1/3 fail before 100,000 years. The rate of failure on a fractional basis is estimated from Figure 2.1-12c in the range 2 – 5.6 10^{-6} per year. To convert this fractional rate to a rate in units of failed WPs /year, multiply by the number of CDSP packages. Note the high end of the range occurs around 15,000 years.

There are 3416 CDSP WPs (SAR Table 6.3.7-1), each containing 1168 g of Tc-99 (SAR Table 6.3.7-5). Thus peak mean release rate is approximately

Release rate = 1168 * 3416 * 5.6 10^{-6} g/yr = 22 g/yr

DOE has not provided mean release rates for the case of interest. From Figure 2.1-24, we estimate releases of 8-12 g/yr in the time frame 10,000 to 100,000 years.

Note also that a second broader peak caused by failures of CSNF packages is apparent around 300,000 years.

Entries into Scientific Notebook No. 318E for the period 9.22.09 to 10.23.09 were made by Scott Painter. No original text entered into this scientific notebook has been removed.

Scott Painter
1-18-10

1/18/2010

ADDITIONAL INFORMATION FOR SCIENTIFIC NOTEBOOK NO. 318E

Document Date:	09/2004
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:	09/2004
Operating System: (including version number)	Windows XP
Application Used: (including version number)	
Media Type: (CDs, 3 ½, 5 1/4 disks, etc.)	1 CD
File Types: (.exe, .bat, .zip, etc.)	ASCII Files
Remarks: (computer runs, etc.)	Notebook supplemental material

ADDITIONAL INFORMATION FOR SCIENTIFIC NOTEBOOK NO. 318E

Document Date:	3/22/1999
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:	01/19/2010
Operating System: (including version number)	Windows
Application Used: (including version number)	
Media Type: (CDs, 3 ½, 5 1/4 disks, etc.)	1 DVD
File Types: (.exe, .bat, .zip, etc.)	xls, pdf
Remarks: (computer runs, etc.)	Notebook supplemental material