

# **COLLAGE II: A Numerical Code for Radionuclide Migration Through a Fractured Geosphere in Aqueous and Colloidal Phases**

*Peter Grindrod  
Neil Cooper*

May 1993

# **SKi**

STATENS KÄRNKRAFTINSPEKTION  
SWEDISH NUCLEAR POWER INSPECTORATE

SKI TR 93:45

**COLLAGE II: A Numerical Code for Radionuclide Migration Through a Fractured Geosphere in Aqueous and Colloidal Phases**

Peter Grindrod  
Neil Cooper

Intera Information Technologies Ltd  
Chiltern House  
45 Station Road  
Henley-on-Thames  
Oxon, RG9 1AT  
U.K.

May 1993

This report concerns a study which has been conducted for the Swedish Nuclear Power Inspectorate (SKI). The conclusions and viewpoints presented in the report are those of the author(s) and do not necessarily coincide with those of the SKI.

IM2145-S  
Version 1

**COLLAGE II:  
A numerical code for  
radionuclide migration through  
a fractured geosphere in  
aqueous and colloidal phases.**

**Peter Grindrod**

**Neil C. Cooper**

**May 1993**

**Intera Information Technologies**

Chiltern House, 45 Station Road, Henley-on-Thames, Oxfordshire RG9 1AT, UK  
Tel (0491) 410474 Fax (0491) 576916



**INTERA**

**Environmental Division**

# **COLLAGE II: A numerical code for radionuclide migration through a fractured geosphere in aqueous and colloidal phases.**

**Intera Information Technologies Ltd.**

## **Summary**

In previous work, the COLLAGE code was developed to model the impact of mobile and immobile colloidal material upon the dispersal and migration of a radionuclide species within a saturated planer fracture surrounded by porous media. The adsorption of radionuclides to colloid surfaces was treated as instantaneous and reversible.

In this report we present a new version of the code, COLLAGE II. Here the adsorption of radionuclides to the colloidal material is treated via first order kinetics. The flow and geometry of the fracture remain as in the previous model.

The major effect of colloids upon the radionuclide species is to adsorb them within the fracture space and thus exclude them from the surrounding porous medium. Thus the matrix diffusion process, a strongly retarding effect, is exchanged for a colloid capture/release process by which adsorbed nuclides are also retarded. The effects of having a colloid-radionuclide kinetic interaction include the phenomena of double pulse breakthrough (the pseudo colloid population followed by the solute plume) in cases where the desorption process is slow and the pseudo colloids are highly mobile.

Some example calculations are given and some verification examples are discussed.

Finally a complete listing of the code is presented as an appendix, including the subroutines allowing for the numerical inversion of the Laplace transformed solution via Talbot's method.

# 1

## The Conceptual Model

We consider the transport of a single radionuclide species through saturated fractured rock in the presence of both mobile and immobile colloid species. Specifically we shall model a single planar fracture surrounded by a saturated porous rock mass. Within the fracture the nuclides are present in a solute phase as well as being adsorbed to both mobile and immobile colloid material. The nuclides may also gain access to the rock matrix porespace, via diffusion, where they may also become sorbed.

This is summarised schematically in figure 1.

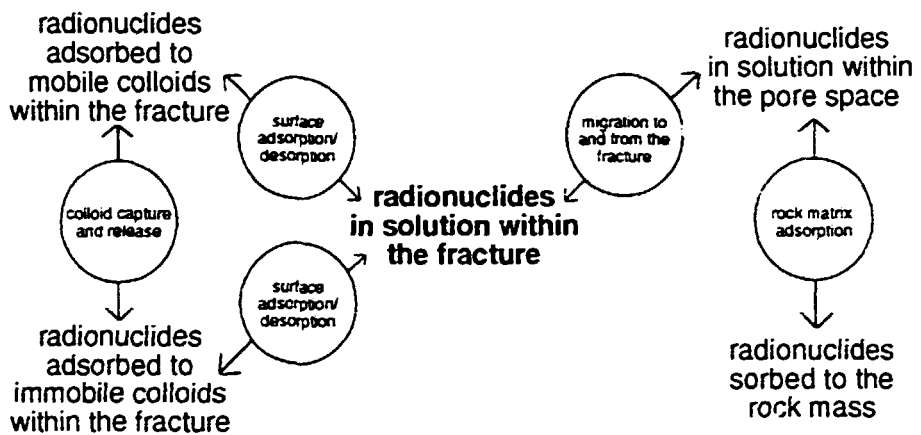


Figure 1: The partition of the radionuclide population, and exchange between the classes.

In earlier work, COLLAGE modeled the adsorption and desorption of nuclides to colloid surfaces as an instantaneous reversible reaction. In this report, COLLAGE II will be used to solve radionuclide migration problems whilst treating such adsorption and desorption of nuclides dynamically. Thus within the fracture there are three phases to be considered, with exchange of nuclides subject to rate constants. In some cases total breakthrough behaviour becomes double peaked: a phenomena previously precluded by the instantaneous modelling assumptions.

The ambient colloid distribution within the fracture space will be assumed to be in dynamic equilibrium. Thus there is a constant exchange of material between the mobile and immobile forms.

## 2

## The Mathematical Model

The model equations solved by the COLLAGE II code represent nuclide dispersal within a thin planar fracture surrounded by porous, saturated, rock matrix. As in previous studies [1] with implicit planar symmetry and cross fracture averaging, this results in a simplified problem in two spatial dimensions: the  $x$  coordinate varying along the axis of the fracture, in the direction of fluid flow, and the  $z$  coordinate defining locations within the rock matrix, increasing from zero at the fracture. Following [1], the equations for the radionuclide concentration in solution ( $w$ ), sorbed on mobile colloids ( $v$ ) and sorbed on immobile colloids ( $p$ ) are given by:

$$w_t = \bar{D}w_{xx} - \bar{u}w_x + \frac{\phi D}{b}W_z|_{z=0} - \lambda w - s_1(c_0, w) - s_2(d_0, w) + g_1(v) + g_2(p) \quad (1)$$

$$v_t = D^*v_{xx} - u^*v_x - \lambda v - \bar{q}(p, v, d_0) + s_1(c_0, w) - g_1(v) \quad (2)$$

$$p_t = -\lambda p + \bar{q}(p, v, d_0) + s_2(d_0, w) - g_2(p) \quad (3)$$

where  $s_1(c_0, w)$  and  $s_2(d_0, w)$  are the rates per unit fluid volume at which nuclides in solution become sorbed to free and immobile colloids;  $g_1(v)$  and  $g_2(p)$  are the rates of desorption from free and immobile colloids;  $\bar{q}(p, v, d_0)$  denotes the rates at which nuclides are exchanged between immobile and free colloids as a consequence of instantaneous colloidal capture or release.

In this analysis the density of free and immobile colloids is assumed fixed in dynamic equilibrium at values of  $c_0$  and  $d_0$ . Other parameters are defined in Table 1.

The concentration of nuclides in solution within the porespace is given by

$$RW_t = DW_{zz} - R\lambda W \quad (4)$$

where  $R$  is the retardation within the rock.

If it assumed that the processes of capture and release of the colloid species are rapidly equilibrated relative to the timescales associated with nuclide transport, then the nuclide concentration on immobile colloids is given by

$$p = \frac{d_0}{c_0}v = \beta v \quad (5)$$

which defines  $\beta$  as the ratio of nuclides sorbed to immobile colloidal material, to nuclide sorbed to mobile colloidal material (per unit fluid volume).

Table 1

$\bar{D}$	Dispersion coefficient of solute
$D^*$	Dispersion coefficient of free colloids
$\bar{u}$	Average groundwater flow rate
$u^*$	Average free colloid flow rate
$\phi$	Porosity of rock
$D$	Molecular diffusion coefficient of the solute within the rock matrix
$b$	Half fracture width
$\lambda$	Radioactive decay rate
$R$	Retardation of the nuclide in the porous rock
$\alpha_1$	year <sup>-1</sup> Rate of sorption/desorption to mobile colloids
$\alpha_2$	year <sup>-1</sup> Rate of sorption/desorption to immobile colloids
$\beta$	Number of immobile colloids/number of mobile colloids
$k_1$	Partition coefficient for free colloid-bound radionuclides
$k_2$	Partition coefficient for immobile colloid-bound radionuclides
$\eta$	Fraction of injection inventory in solute phase
$L$	Length modeled
$n$	Downstream boundary condition (zero concentration) located at $x = nL$ .

The adsorption and desorption of nuclides by colloids is assumed to be kinetic here, whereas in [2] it was assumed instantaneous. The relevant terms in (1) and (2) will be taken to be first order, and are given by

$$\begin{aligned} g_1 &= \alpha_1 v & s_1(c_0, w) &= \alpha_1 k_1 w \\ g_2 &= \alpha_2 p = \beta \alpha_2 v & s_2(d_0, w) &= \alpha_2 k_2 w \end{aligned} \quad (6)$$

Here the constants  $\alpha_i$  represent the kinetic disassociation rates, and the constants  $k_i \alpha_i$  represent the adsorption rates. These constants (the  $k_i$  and the  $\alpha_i$ ) implicitly depend upon the ambient densities of the colloid species, and their sorbative properties [1]. Clearly the larger values that are assumed for the  $\alpha_i$ , the faster the sorption processes are equilibrated.

Thus if the adsorption and desorption of the nuclides by colloids were ever in equilibrium, then  $v = k_1 w$ , and  $p = k_2 w$  would hold (as is assumed in [1], equation (16)).

Substituting (5) and (6) into (1), (2) and (3) gives the equations to be solved:

$$\begin{aligned} w_t &= \tilde{D} w_{xx} - \bar{u} w_x + \frac{\phi D}{b} W_z|_{z=0} - \lambda w \\ &\quad + (\alpha_1 + \beta \alpha_2) v - (\alpha_1 k_1 + \alpha_2 k_2) w \end{aligned} \quad (7)$$

$$\begin{aligned} (1 + \beta) v_t &= D^* v_{xx} - u^* v_x - \lambda(1 + \beta) v \\ &\quad - (\alpha_1 + \beta \alpha_2) v + (\alpha_1 k_1 + \alpha_2 k_2) w \end{aligned} \quad (8)$$

The boundary conditions to be imposed in solving these equations are of zero concentration a distance  $nL$  downstream; and of unit pulse injection at  $x = 0$ ,  $t = 0$ , with a fraction  $\eta$  being in the dissolved phase,  $(1 - \eta)$  being on mobile colloids.

Thus

$$w(x, 0) = v(x, 0) = 0, \quad 0 < x < nL \quad (9)$$

$$w(nL, t) = v(nL, t) = 0, \quad t \geq 0 \quad (10)$$

$$\left[ \bar{u} w - \tilde{D} \frac{\partial w}{\partial x} \right]_{x=0} = \frac{\delta(t)}{2b} \eta, \quad t \geq 0 \quad (11)$$

$$\left[ u^* v - D^* \frac{\partial v}{\partial x} \right]_{x=0} = \frac{\delta(t)}{2b} (1 - \eta), \quad t \geq 0 \quad (12)$$

The concentration in the porous rock has boundary condition:

$$W(x, z, t) \rightarrow 0 \text{ as } z \rightarrow \infty \quad (13)$$



$$W(x, 0, t) = w(x, t) \quad (14)$$

$$W(x, z, 0) = 0. \quad (15)$$

It is required to find the outflux at  $x = L$ :

$$f_w(t) = 2b \left[ \bar{u}w - \bar{D} \frac{\partial w}{\partial x} \right]_{x=L} \quad (16)$$

$$f_v(t) = 2b \left[ u^*v - D^* \frac{\partial v}{\partial x} \right]_{x=L} \quad (17)$$

Note that if  $n = 1$  the downstream boundary condition is imposed where the flux is calculated. If additionally  $\alpha_1 = \alpha_2 \gg 1$ , so that sorption becomes effectively instantaneous, then the model described in [2] is recovered (their equation (1) being  $(7) + k_1(\delta)$ ).

Collage II solves the above model using a method based upon Laplace Transforms. The details of this solution are given in Appendix A.

### 3 Model calculations

The model developed in the previous section has been coded for a 486-based PC using the Salford FTN77 compiler. A listing of the code is given in Appendix B. Using this code a series of example runs have been undertaken which show possible effects of considering dynamic sorption to and desorption from colloids. All the examples given in this section are based on the same basic model, which is now described. Parameters are selected for illustrative purposes only.

Transport of a unit injection of a non-decaying tracer is considered in a fracture 1000m long, with half-width 1 cm. The groundwater velocity is 1 m/yr, whilst the free colloids move with an average speed of 1.32 m/yr. Their respective dispersivities ( $\bar{D}, D^*$ ) are 50 and 140 m<sup>2</sup>/s. The rock matrix is porous, with porosity ( $\phi$ ) 1E-2, diffusivity ( $D$ ) 7.875E-4 and retardation ( $R$ ) 675.1. The downstream boundary condition was taken to be zero concentration at  $x=1000$ m, and the output used was the flux at this location.

Figure 2.1 shows the flux in the case of  $k_1 = 50$  and  $k_2 = 0$  (large numbers of mobile colloids and no immobile colloids) for a range of values of the rate of transfer ( $\alpha_1$ ) of the tracer between the colloid bound and the solute phases.  $\alpha_1$  is measured in units of inverse time, thus large values of  $\alpha_1$  correspond to rapid transfer. From the graph it can be seen that for

transfer rates greater than  $0.1 \text{ yr}^{-1}$  (times shorter than 10 years), the results given are independent of the reaction rate, since almost all (98%) nuclide transport is via mobile pseudocolloid form. As the value of  $\alpha$  is further decreased so changes appear, since the nuclide-colloid interactions can obtain equilibrium. For  $\alpha=1\text{E}-3$ , the peak concentration decreases by 20% whilst the concentrations at times greater than 10,000 years increases by more than a factor of 2. The changes are clearest for the curve  $\alpha = 1\text{E} - 6$ . This rate is so slow that by the time of the peak in the other cases (560 years) only a small fraction of the tracer has become bound to the colloids, most still being in the solute form (much being bound within the rock matrix). Thus a two-peak distribution is produced, the first peak being at the time of the pseudocolloid breakthrough, the later at the solute breakthrough time (as shown by the 'No Colloids' breakthrough curve). This graph highlights not only the effect of colloids on the flux of radionuclides, but also the effect of the rate of transfer of radionuclides between colloids and solute.

Figure 2.2, for the case of  $k_1 = 1$  and  $k_2 = 0$  (less mobile colloids, though still no immobile colloids), shows an even greater sensitivity to  $\alpha$ , the transfer rate. Due to the reduced number of colloids, the breakthrough time for low  $\alpha$  (rapid transfer) is now not until about 10,000 years. At shorter times, slight difference can be seen between all the curves, showing that even a transfer rate of  $1 \text{ year}^{-1}$  can produce different flux to that predicted, if the transfer is assumed to be instantaneous. The most striking component of this figure, however, is that the highest flux is predicted for a transfer rate of  $\alpha=1\text{E}-3$  (equivalent to 1000 years). In this case a significant amount of tracer transfers to the colloids but does not then transfer back to solution significantly. For slower transfer rates ( $\alpha = 1\text{E}-6$ ) the initial peak is reduced, compared to that in figure 2.1.

Effect of varying alpha on total flux, with  $K1=50$ ;  $K2=0$ .

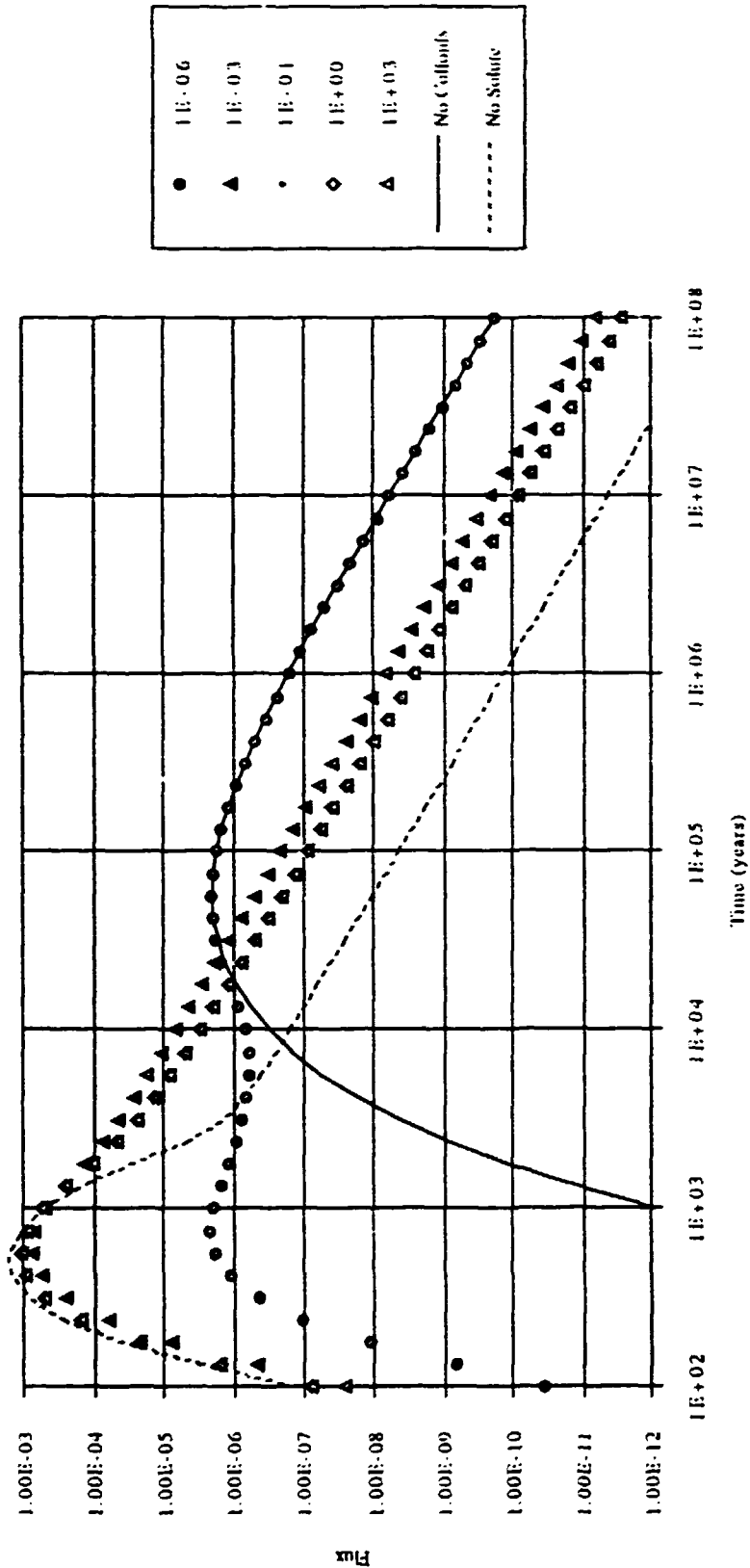


Figure 2.1

Effect of varying alpha on total flux for  $K1=1$ ;  $K2=0$ .

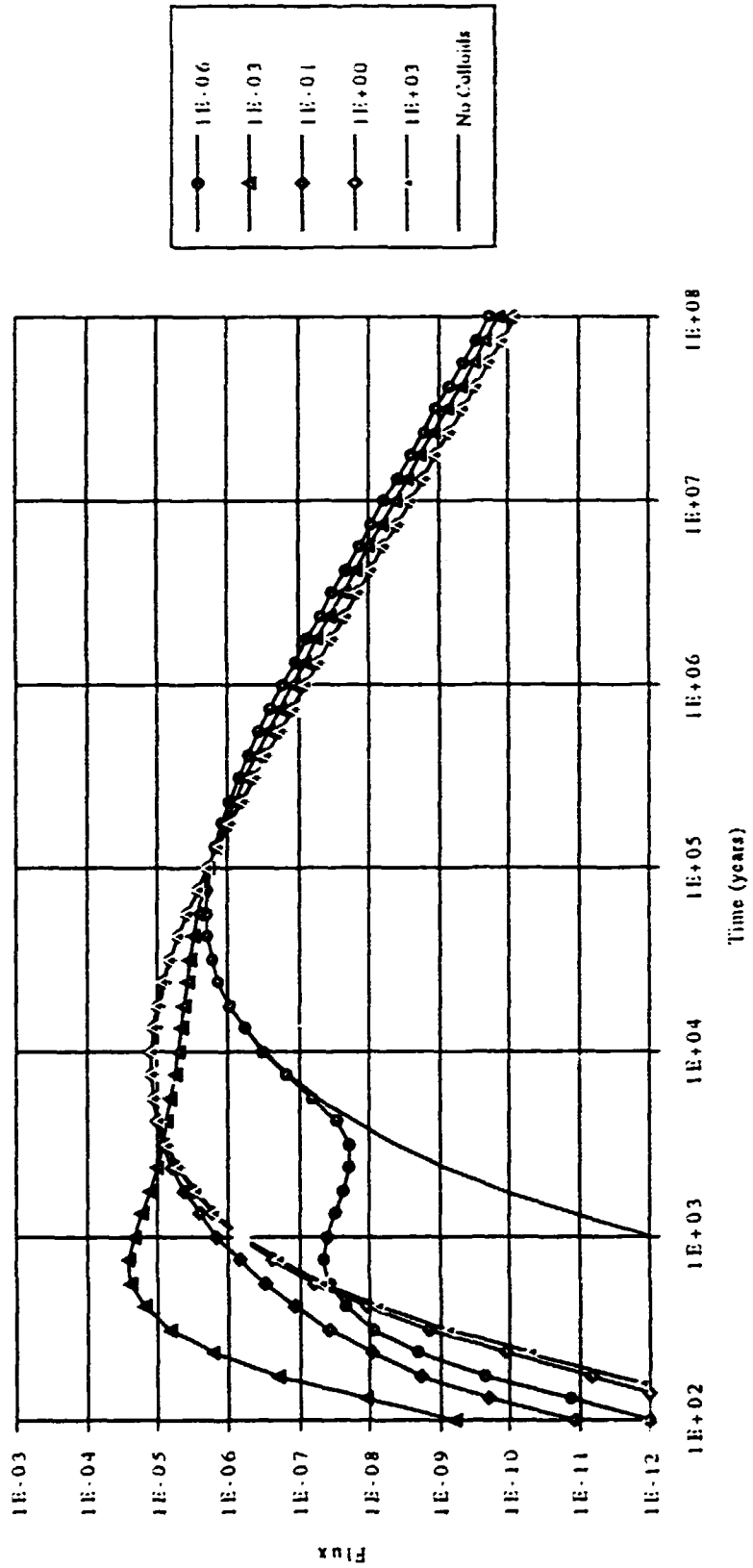


Figure 2.2

Figure 2.3 shows the split between pseudocolloids and solute transfer for two of the cases shown in figure 2.1. For the case of  $\alpha=1$  it can be seen that the transfer by colloids always dominates that by solute. This is not surprising as the transfer rate is relatively fast and the colloids have a much greater capacity than the water. For the case of  $\alpha=1E-6$  (very slow transfer rate), then before significant solute breakthrough transport by colloids dominates, while after this time transfer by solute, which is the main component of transport, dominates.

Initial tests, including both mobile and immobile colloid phases, showed that the presence of immobile colloids results in a reduced flux at the early peak (at 560 years).

In figure 2.4 we show the results for  $\alpha = 1E-6$  and  $1E-3$ , assuming  $k_1 = k_2 = 50$ , for comparison with figure 2.2. The effect of two different transfer rates ( $\alpha_1 \neq \alpha_2$ ) was that behaviour was generally determined by the faster rate (the transfer of colloids between the mobile and immobile phases having been assumed to be very rapid, in the derivation of equation (5)).

$K1=50; K2=0; \alpha = 1$

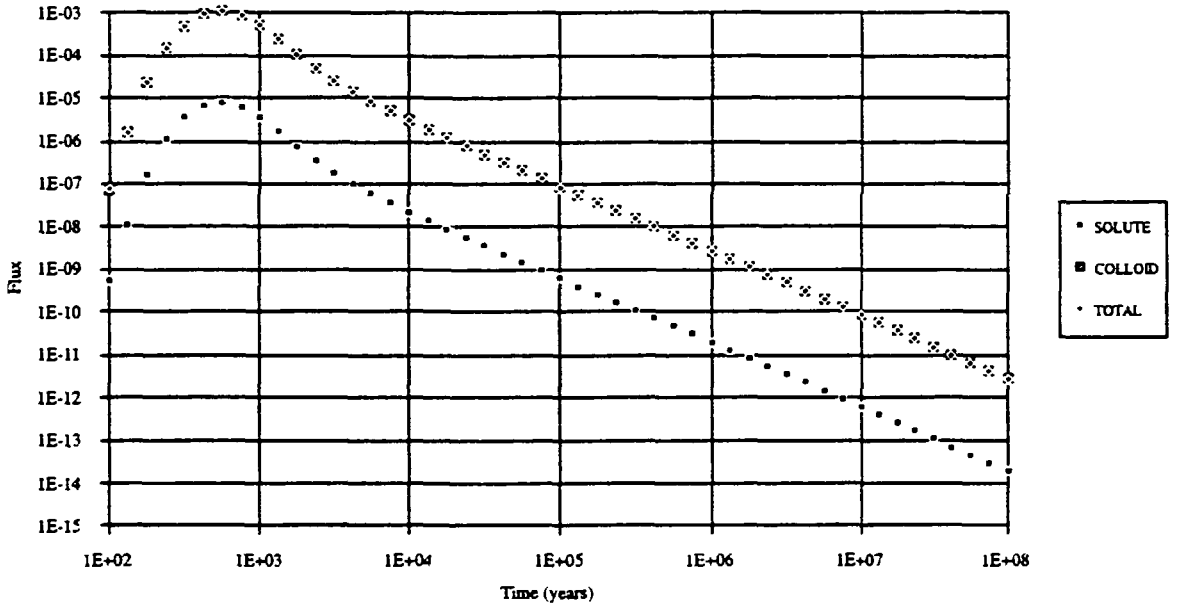


Figure 2.3a

$K1=50; K2=0; \alpha = 1E-6$

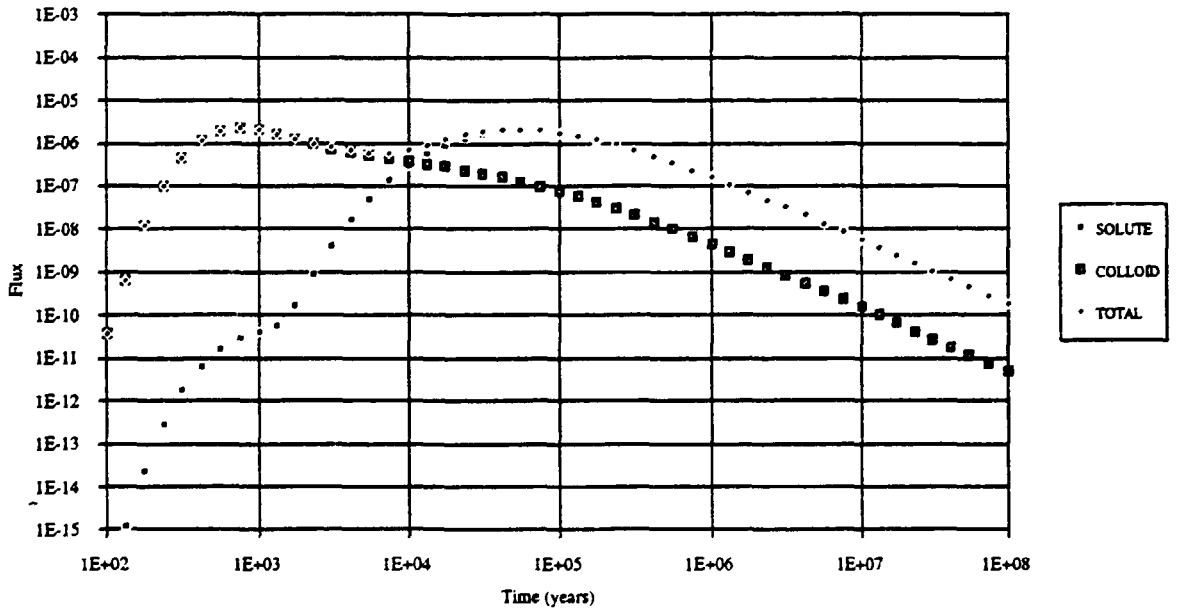


Figure 2.3b

Effect of varying alpha 1 and alpha 2 on the total flux,  $K_1=1$ ;  $K_2=1$ ;  $\text{Beta}=1$

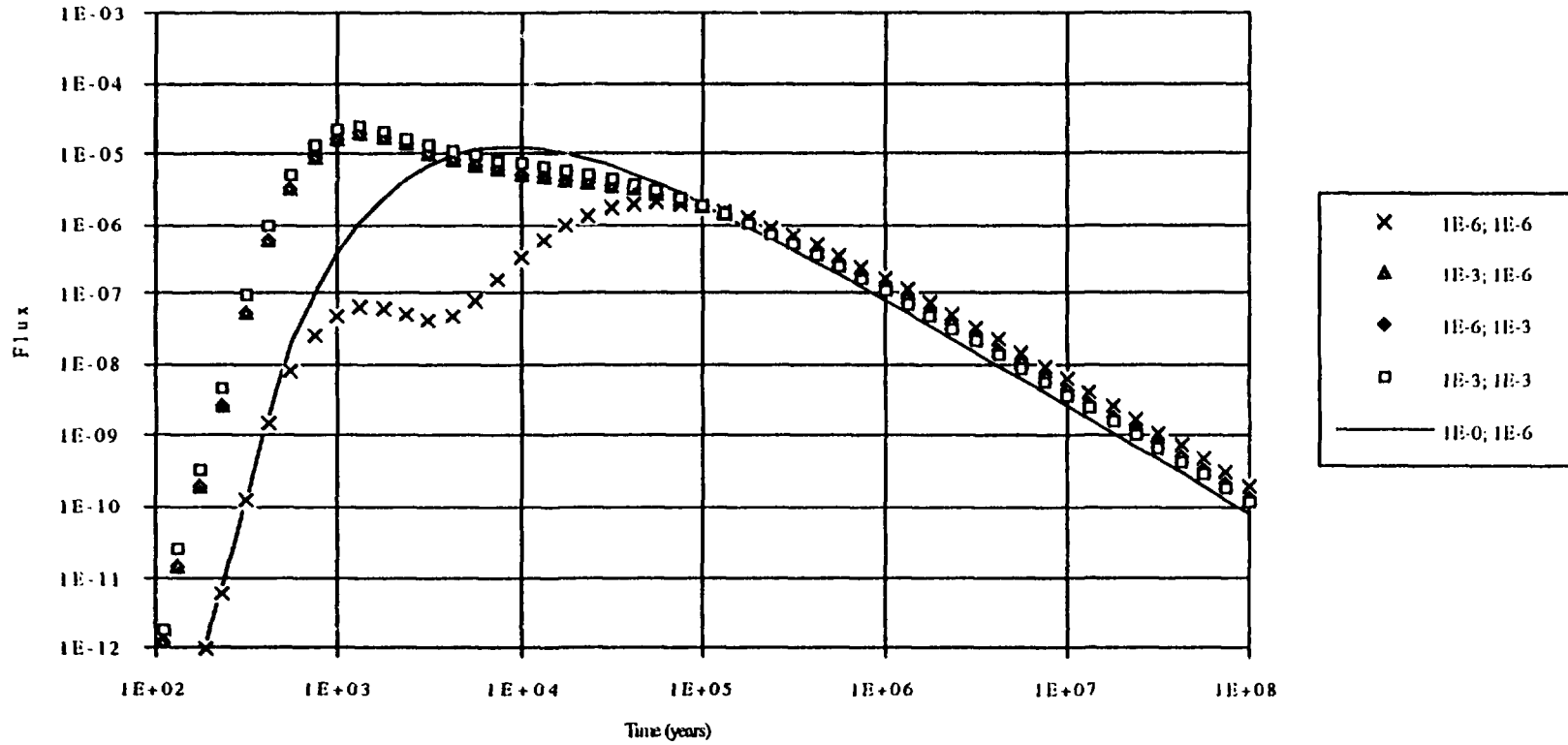


Figure 2.4

## 4 Code verification.

Various checks have been undertaken to verify that the program COLLAGE II, is an accurate coding of the mathematical model described here.

Initial tests were undertaken to confirm that the results in the case of instantaneous sorption agree with those from COLLAGE, presented in [2], by setting  $\alpha_1$  and  $\alpha_2$  to be very large. These tests showed very good agreement for the peak value and time, with values within 0.1% and at the same timestep, respectively. Away from the peak slight differences did occur, particularly at early times.

The main testing of COLLAGE II was by comparing its predictions with those of a code which models the transfer of radionuclides through porous, fractured rock with first order kinematic sorption processes included. COLLAGE II was used with no mobile colloids and an appropriate level of immobile colloids, thus allowing the basic equations for the two models to be equivalent. Tests in this case again showed very close agreement, with the values of peaks being reproduced to within 0.1% and the times predicted being at the same timestep.

In combination, these two sets of results verified the major components of COLLAGE II: the separate modelling of tracer transport by colloids from that by solute transport; and the modelling of kinematic transfer of tracer between the solute and colloidal phases.



## A Solution of the equations.

The model described in Section 2 is solved by the method used in previous studies [1, 2, 3] - that of Laplace Transforms, defined by

$$\hat{u}(s) = \int_0^{\infty} u(t)e^{-st} dt. \quad (18)$$

For the nuclide in the rock matrix, (4) and (13) are solved directly, to give

$$\hat{W}(s, x, z) = \hat{w}(s, x) \exp \left\{ -z \sqrt{\frac{R(s + \lambda)}{D}} \right\}$$

and thus

$$\hat{W}_z|_{z=0} = -\hat{w}(s, x) \sqrt{\frac{R(s + \lambda)}{D}}. \quad (19)$$

The main equations, (7) and (8), become,

$$\begin{aligned} \bar{D}\hat{w}_{xx} - \bar{u}\hat{w}_x - \hat{w}(\lambda + s + \frac{\phi D}{b} \sqrt{\frac{R(s + \lambda)}{D}} + (\alpha_1 k_1 + \alpha_2 k_2)) \\ = -(\alpha_1 + \beta \alpha_2)\hat{v} \end{aligned} \quad (20)$$

$$\begin{aligned} D^*\hat{v}_{xx} - u^*\hat{v}_x - \hat{v}\{(1 + \beta)(\lambda + s) + (\alpha_1 + \beta \alpha_2)\} \\ = -(\alpha_1 k_1 + \alpha_2 k_2)\hat{w} \end{aligned} \quad (21)$$

with boundary conditions

$$\hat{w}(nL, s) = \hat{v}(nL, s) = 0 \quad (22)$$

$$\left[ \bar{u}\hat{w} - \bar{D} \frac{\partial \hat{w}}{\partial x} \right]_{x=0} = \frac{\eta}{2b} \quad (23)$$

$$\left[ u^*\hat{v} - D^* \frac{\partial \hat{v}}{\partial x} \right]_{x=0} = \frac{(1 - \eta)}{2b}. \quad (24)$$

To solve these equations they are written in a simplified form:

$$a_1 \hat{w}_{xx} + b_1 \hat{w}_x + c_1 \hat{w} = d_1 \hat{v} \quad (25)$$

$$a_2 \hat{v}_{xx} + b_2 \hat{v}_x + c_2 \hat{v} = d_2 \hat{w} \quad (26)$$

$$\hat{w}(nL, s) = \hat{v}(nL, s) = 0$$

$$\left[ b_1 \hat{w} + a_1 \frac{\partial \hat{w}}{\partial x} \right]_{x=0} = \frac{-\eta}{2b} \quad \left[ b_2 \hat{v} + a_2 \frac{\partial \hat{v}}{\partial x} \right]_{x=0} = \frac{-(1 - \eta)}{2b} \quad (27)$$

(The definitions of  $a_i$ ,  $b_i$ ,  $c_i$  and  $d_i$  is apparent by comparing with equations (20) through (24).)

Next  $v$  is eliminated from (25) and (27), by using (26), giving

$$a_1 a_2 \hat{w}'''' + (a_1 b_2 + b_1 a_2) \hat{w}''' + (a_1 c_2 + b_1 b_2 + c_1 a_2) \hat{w}'' + (b_1 c_2 + b_2 c_1) \hat{w}' + (c_1 c_2 - d_1 d_2) \hat{w} = 0 \quad (28)$$

with boundary conditions:

$$\begin{aligned} \hat{w}(nL, s) &= 0 \\ a_1 \hat{w}''(nL, s) + b_1 \hat{w}'(nL, s) + c_1 \hat{w}(nL, s) &= 0 \\ b_1 \hat{w}(0, s) + a_1 \hat{w}'(0, s) &= -\frac{\eta}{2b} \\ a_1 a_2 \hat{w}'''(0, s) + (a_2 b_1 + b_2 a_1) \hat{w}''(0, s) + (a_2 c_1 + b_2 b_1) \hat{w}'(0, s) \\ &\quad + c_1 b_2 \hat{w}(0, s) = -\frac{d_1(1-\eta)}{2b}. \end{aligned} \quad (29)$$

Setting

$$\hat{w} = \sum_{i=1}^4 A_i e^{r_i(x-nL)}$$

in (28) gives a fourth order polynomial which can be solved by the routine ZROOTS given in [4]. This uses deflation to calculate all four roots.

Once the roots  $r_1, r_2, r_3$  and  $r_4$  are found, the  $A_i$ 's are calculated from the boundary conditions, which yield a matrix equation:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ X_1 & X_2 & X_3 & X_4 \\ Y_1 e^{-r_1 nL} & Y_2 e^{-r_2 nL} & Y_3 e^{-r_3 nL} & Y_4 e^{-r_4 nL} \\ Z_1 e^{-r_1 nL} & Z_2 e^{-r_2 nL} & Z_3 e^{-r_3 nL} & Z_4 e^{-r_4 nL} \end{pmatrix} \begin{pmatrix} A_1 \\ A_2 \\ A_3 \\ A_4 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -\frac{\eta}{2b} \\ -\frac{d_1(1-\eta)}{2b} \end{pmatrix} \quad (30)$$

where

$$\begin{aligned} X_i &= a_1 r_i^2 + b_1 r_i + c_1 \\ Y_i &= b_1 + a_1 r_i \\ Z_i &= a_1 a_2 r_i^3 + (a_2 b_1 + b_2 a_1) r_i^2 + (a_2 c_1 + b_2 b_1) r_i + c_1 b_2. \end{aligned} \quad (31)$$

The matrix equation (30) is solved by Gaussian Elimination using the routine GAUSS given in [4], to yield the  $A_i$ 's. These are then used to

calculate the Laplace Transform of the Output:

$$\begin{aligned}\hat{f}_w &= 2b\left[\bar{u}\hat{w} - \bar{D}\frac{\partial\hat{w}}{\partial x}\right]_{x=L} \quad (\text{from (16)}) \\ \hat{f}_w &= 2b\sum_{i=1}^4(\bar{u} - r_i\bar{D})\cdot A_i e^{-r_i(n-1)L}\end{aligned}\tag{32}$$

$\hat{f}_v$  is calculated in an analogous manner, beginning by eliminating  $w$  from equations (25) and (27).

The final solution step is to invert the Laplace Transforms using the Talbot method (as described in [1, 2, 3]).

## B Code Listing

```
C *****
C
C          PROGRAM QA INFORMATION
C
C          PROGRAM : COLLAGE 2
C          VERSION : 2.0.3.5
C          CREATED : NOVEMBER 1992
C          LAST EDIT : February 1993
C          AUTHORS : N S Cooper and P G Grindrod
C          Based on: COLLAGE
C
C  Compilation : FTN77 COLLAGE2 /LINK77 /DCLVAR /CHECK /DREAL
C
C DESCRIPTION:
C Collage models a one dimensional geosphere breakthrough
C problem for radionuclides in groundwater containing natural
C colloids. The breakthrough is calculated as a function of time
C by solving for Laplace transforms of the concentration
C and inverting this using Talbot's method.
C
C Version 2 includes kinematic sorption/desorption to/from
C moving and stationary colloids.
C
C *****
```

### PROGRAM COLL2

```
C Real Variables
      REAL A, A1, A2, ALP1, ALP2, B1, B2, BETA1, DROCK, DSTAR, DT,
      1      DTIL, FINT, FRAC, FV, FW, L, LAMBDA, PHI, PI, Q, R,
      2      RK1, RK2, RLAM, THETA, TIME, TSTART, TEND, USTAR, UTI
C Integer Variables
      INTEGER I, ICOUNT, IR, K, N, NL, NCURVE
C Complex arrays
      COMPLEX COEFF(4), ROOT(4), ROOT1(4), TERM(5)
C Complex Variables
      COMPLEX C1, C2, D1, D2, FHATW, FHATV, S, MATRIX (4,4),
      1      S0, SPRIME, TEMP, X(4), Y(4), Z(4)
C Character Variables
      CHARACTER*30 NAMIT, NAMTWO, FDATE
```

```

CHARACTER*80 TITLE
C Intrinsic functions
  INTRINSIC ATAN, CABS, CEXP, CMPLX, CSQRT, EXP, REAL, TAN
C External functions
  EXTERNAL GAUSSJ, ZROOTS

CALL DCLOCK(TSTART)
PI = ATAN (1.0) * 4.

c
c prompt user for i/o file names, and open files
c
  WRITE(6,*)' ENTER INPUT FILE NAME  '
  READ(5,'(A30)')NAMIT
  WRITE(6,*)' ENTER OUTPUT FILE NAME  '
  READ(5,'(A30)')NAMTWO
  OPEN(UNIT=11,FILE=NAMIT,STATUS='OLD')
  OPEN(UNIT=12,FILE=NAMTWO,STATUS='UNKNOWN')
  WRITE (12,988)NAMIT,NAMTWO
988  FORMAT (/20X,' COLLAGE version 2.0.3.5',/
1      20X,' =====',//,
2      10X,' INPUT FILE : ',A20,/
3      10X,' OUTPUT FILE : ',A20/)
  WRITE (12,989)FDATE()
989  FORMAT(10X,' PROGRAM RUN ON ',A30,/)

c
c announce, read and echo input Data
c
  READ(11,990)TITLE
  WRITE (12,991)TITLE
990  FORMAT (A80)
991  FORMAT (1X,A80/)
c mean solute and colloid advection velocities
  READ(11,*)UTIL,USTAR
  WRITE(12,*)' UTIL, USTAR =',UTIL,USTAR

c
c solute and colloid dispersion coefficients
c for representative fracture
  READ(11,*)DTIL,DSTAR
  WRITE(12,*)' DTIL, DSTAR =',DTIL,DSTAR

c
c coefficient of outflow into rock matrix
  READ(11,*)PHI
C ETA NO LONGER USED, replaced by PHI
  WRITE(12,*)' PHI =',PHI

```

```

C
C  decay rate (epsilon in report)
      READ(11,*)RLAM
      WRITE(12,*)' DECAY RATE =' ,RLAM
C
C  retardation factor
      READ(11,*)R
      WRITE(12,*)' RETARDATION FACTOR, R =' ,R
C
C  solute diffusivity within rock matrix
      READ(11,*)DROCK
      WRITE(12,*)' SOLUTE DIFFUSIVITY IN ROCK, DROCK =' ,DROCK
C
C  Fracture details
      READ(11,*)L,A,NL
      WRITE(12,*)' FRACTURE LENGTH',L,' Width', A
      WRITE(12,*)' BOUNDARY AT ',NL,' L'
C
C  Talbot inversion parameters
      READ(11,*)N,Q,LAMBDA
      WRITE(12,*)' TALBOT: N,Q,LAMBDA =' ,N,Q,LAMBDA
C
C  number of values of k1
      READ(11,*)NCURVE
      IF (NCURVE.NE.1)THEN
          PRINT *,' **WARNING **      VALUE OF NCURVE NOT 1'
          PRINT *,' **WARNING **      ONLY FIRST VALUE OF K1 USED'
      ENDIF
C      WRITE(12,*)' NUMBER OF VALUES OF K1',NCURVE
C
C  value of k1
      READ(11,*)RK1
      WRITE(12,*)' K1 =' ,RK1
C
C  value of k2
      READ(11,*)RK2
      WRITE(12,*)' K2 =' ,RK2
C
C  kinematic sorption/desorption rates
      READ (11,*)ALP1,ALP2
      WRITE (12,*)' SORPTION RATES, ALPHA1, ALPHA2 ',ALP1,ALP2
C
C  equilibrium ratio beta = p/v
      READ (11,*)BETA1

```

```

WRITE (12,*)' EQUILIBRIUM RATIO, BETA1, ',BETA1

c INJECTION FRACTION IN SOLUTION
  READ (11,*)FRAC
  WRITE (12,*)' INJECTION FRACTION IN SOLUTE',FRAC

c
c SET HEADER FOR OUTPUT OF RESULTS
c
  PRINT 86
  WRITE (12,86)
86  FORMAT (//,7X,'TIME',6X,'SOLUTE',11X,'COLLOID',8X,'TOTAL',
1    4X,'INTEGRATE FLUX'//)

C SET INTEGRATED FLUX TO ZERO AT T = 100 YEARS
  FINT = 0

c
c scan timescale
c
  DO 100 I=0,48

    TIME = 10**(2+I/8.)

c
c solve for concentration at distance NL*L and given time
c by Talbot inversion of Laplace transform
c

  DO 80 K=0,N-1

C Calculate the coefficients for the two simultaneous
c second order differential equations
c a1,b1,c1 and d1 for w eqn
c a2,b2,c2 and d2 for v eqn
  A1 = DTIL
  A2 = DSTAR
  B1 = -UTIL
  B2 = -USTAR

C calculate Talbot variables Theta, S0 and dS/dTheta
  IF (K.EQ.0) THEN
    THETA = 0
    S0 = 1
    SPRIME= Q * .5

```

```

ELSE
  THETA = K * PI / N
  S0 = THETA/TAN(THETA) + Q*CMPLX(0.,1.)*THETA
  SPRIME= (Q + CMPLX(0.,1.)*( THETA +
1      ( ((THETA/TAN(THETA))-1.)/TAN(THETA))))*0.5
ENDIF

C next apply Talbot factoring (lambda)
  S = 6. * S0 / TIME
C and calculate remaining coefficient
  C1 = -(RLAM + S + (PHI*DROCK/A) * CSQRT(R*(S+RLAM)/DROCK)+
1      (ALP1*RK1 +ALP2*RK2))
  C2 = -(((1+BETA1)*(RLAM+S) + (ALP1 +BETA1*ALP2))
c   print *, ' C1 = ',C1
  D1 = - ALP1 - BETA1*ALP2
  D2 = - ALP1*RK1 - ALP2 *RK2

C Calculate the coefficients of the fourth order
C characteristic equation
  TERM(5) = A1*A2
  TERM(4) = A1*B2 + B1*A2
  TERM(3) = A1*C2 + B1*B2 + C1*A2
  TERM(2) = B1*C2 + C1*B2
  TERM(1) = C1*C2 - D1*D2

C Find the roots of the characteristic equation, using the method
C given in Numerical Recipes
C   CALL ZROOTS(TERM,4,ROOT1,.FALSE.)
C   CALL ZROOTS(TERM,4,ROOT1,.TRUE.)

DO 30 IR = 1,4
  ROOT(IR) = ROOT1(5-IR)

C Calculate the boundary condition matrix, to invert for coefficients
  X(IR) = C1 + ROOT(IR) * (B1 + ROOT(IR) * A1)
  IF (CABS(X(IR)).LT.1.E-10)X(IR)=0.0
  Y(IR) = B1 + ROOT(IR) * A1
  Z(IR) = C1*B2 + ROOT(IR) * ((C1*A2 + B1*B2)
1      + ROOT(IR) * ((B1*A2 + A1*B2)
2      + ROOT(IR) * A1*A2 ))

C Set MATRIX for Gauss-Jordan solution
  MATRIX(1,IR) = 1
  MATRIX(2,IR) = X(IR)

```



```

C Fix to avoid numerical overflow
  TEMP = ROOT(IR)
  IF (REAL(ROOT(IR)*NL*L).LT.-600) THEN
    TEMP = CMPLX(-600./(NL*L),AIMAG(ROOT(IR)))
    IF (K.EQ.0)PRINT *,' Overflow avoided for W, I =',I
  ENDIF
  MATRIX(3,IR) = Y(IR) * CEXP(-TEMP * NL * L)
  MATRIX(4,IR) = Z(IR) * CEXP(-TEMP * NL * L)

C and right hand side
  COEFF(IR) = CMPLX(0.,0.)
  COEFF(3) = -.5*FRAC/A
  COEFF(4) = -D1*(1.-FRAC)*.5/A

30 CONTINUE

C Call Numerical Recipies GAUSSJ to solve simultaneous equations
  CALL GAUSSJ(MATRIX,4,4,COEFF,1,1)

C Calculate the Laplace Transforms of the result required
C (NB required result is flux at x=L.)
  FHATW = 0

  DO 40 IR =1,4

    FHATW = FHATW + COEFF(IR) * (UTIL - DTIL *ROOT(IR))
1      * CEXP(-ROOT(IR)*(NL-1)*L)
40 CONTINUE
  FHATW = 2. * A * FHATW

  DO 50 IR = 1,4

    ROOT(IR) = ROOT1(5-IR)

C Calculate the boundary condition matrix, to invert for coefficients
  X(IR) = C2 + ROOT(IR) * (B2 + ROOT(IR) * A2)
  IF (CABS(X(IR)).LT.1.E-10)X(IR)=0.0
  Y(IR) = B2 + ROOT(IR) * A2
  Z(IR) = C2*B1 + ROOT(IR) * ((C2*A1 + B2*B1)
1      + ROOT(IR) * ((B2*A1 + A2*B1)
2      + ROOT(IR) * A2*A1 ))

C Set MATRIX for Gauss-Jordan solution

```

```

        MATRIX(1,IR) = 1
        MATRIX(2,IR) = X(IR)
C Fix to avoid numerical overflow
        TEMP = ROOT(IR)
        IF (REAL(ROOT(IR)*NL*L).LT.-600) THEN
            TEMP = CMLX(-600./(NL*L),AIMAG(ROOT(IR)))
            IF (K.EQ.0)PRINT *,' Overflow avoided for V, I =',I
        ENDIF
        MATRIX(3,IR) = Y(IR) * CEXP(-TEMP * NL * L)
        MATRIX(4,IR) = Z(IR) * CEXP(-TEMP * NL * L)

C and right hand side
        COEFF(IR) = CMLX(0.,0.)
        COEFF(3) = -(1-FRAC)*.5/A
        COEFF(4) = -D2*FRAC*.5/A

50    CONTINUE

C Call Numerical Recipies GAUSSJ to solve simultaneous equations
        CALL GAUSSJ(MATRIX,4,4,COEFF,1,1)

C Calculate the Laplace Transforms of the result required
C (NB required result is flux at x=L.)
        FHATV = 0

        DO 60 IR =1,4
            FHATV = FHATV + COEFF(IR) * (USTAR - DSTAR *ROOT(IR))
1            * CEXP(-ROOT(IR)*(NL-1)*L)
60    CONTINUE
        FHATV = 2. * A * FHATV

C now do Talbot inversion of Laplace Transforms, FHATW and FHATV

        IF (K.EQ.0) THEN
C Used factor of 1/2 for N=0, and give SPRIME explicitly.
            FW = .5*LAMBDA * EXP(LAMBDA)*REAL(FHATW)*Q/(N*TIME)
            FV = .5*LAMBDA * EXP(LAMBDA)*REAL(FHATV)*Q/(N*TIME)
        ELSE
            FW = FW+2.*LAMBDA * REAL(CEXP(LAMBDA*SO)*SPRIME*FHATW)/N/TIME
            FV = FV+2.*LAMBDA * REAL(CEXP(LAMBDA*SO)*SPRIME*FHATV)/N/TIME
        ENDIF

80    CONTINUE

```

```

C CALCULATE THE TIME INTEGRATED FLUX.
C VALUE OF 0.3335 = 1 - 10.**(1/8)
C VALUE OF 2.3335 = 1 + 10.**(1/8)
      FINT = FINT +(FW+FV) * TIME * 0.3335 * 2.3335 *.5 /1.3335

c store results by time and k1 value
c
      PRINT 89,TIME,FW,FV,FW+FV,FINT
      WRITE (12,89)TIME,FW,FV,FW+FV,FINT
89    FORMAT (2X, 1PE10.3, 4E16.6)
100   CONTINUE

      CALL DCLOCK(TEND)
      PRINT 890,TEND-TSTART
      WRITE (12,890)TEND-TSTART
890   FORMAT(/,10X,' CPU TIME USED = ',F10.1,' SECONDS')

      STOP
      END

      SUBROUTINE ZROOTS(A,M,ROOTS,POLISH)
C REAL PARAMETERS
      REAL EPS
C INTEGER PARAMETERS
      INTEGER MAXM
      PARAMETER (EPS=1.E-6,MAXM=101)
C INTEGER VARIABLES
      INTEGER I, J, M, JJ
C EXTERNAL FUNCTIONS
      EXTERNAL LAGUER
C INTRINSIC FUNCTIONS
      INTRINSIC ABS, CMLPX, AIMAG, REAL

      COMPLEX A(*),ROOTS(M),AD(MAXM),X,B,C
      LOGICAL POLISH
      DO 11 J=1,M+1
          AD(J)=A(J)
11    CONTINUE
      DO 13 J=M,1,-1
          X=CMLPX(0.,0.)
          CALL LAGUER(AD,J,X,EPS,.FALSE.)
          IF(ABS(AIMAG(X)).LE.2.*EPS**2*ABS(REAL(X))) X=CMLPX(REAL(X),0.
          ROOTS(J)=X

```

```

        B=AD(J+1)
        DO 12 JJ=J,1,-1
            C=AD(JJ)
            AD(JJ)=B
            B=X*B+C
12     CONTINUE
13     CONTINUE
        IF (POLISH) THEN
            DO 14 J=1,M
                CALL LAGUER(A,M,ROOTS(J),EPS,.TRUE.)
14     CONTINUE
        ENDIF
        DO 16 J=2,M
            X=ROOTS(J)
            DO 15 I=J-1,1,-1
                IF (REAL(ROOTS(I)).LE.REAL(X)) GO TO 10
                ROOTS(I+1)=ROOTS(I)
15     CONTINUE
            I=0
10     ROOTS(I+1)=X
16     CONTINUE
        RETURN
        END

```

```

        SUBROUTINE LAGUER(A,M,X,EPSS,POLISH)
C REAL PARAMETERS
        REAL EPSS
C COMPLEX PARAMETERS
        COMPLEX ZERO
C INTEGER PARAMETERS
        INTEGER MAXIT

        PARAMETER (ZERO=(0.,0.),EPSS=6.E-8,MAXIT=100)
C COMPLEX VARIABLES
        COMPLEX A(*),X,DX,X1,B,D,F,G,H,SQ,GP,GM,G2
C REAL VARIABLES
        REAL DXOLD, ABX, CDX, EPS, ERR
C INTEGER VARIABLES
        INTEGER M, ITER, J
C LOGICAL VARIABLES
        LOGICAL POLISH
C INTRINSIC FUNCTIONS
        INTRINSIC CABS, CSQRT

```

```

DXOLD=CABS(X)
DO 12 ITER=1,MAXIT
  B=A(M+1)
  ERR=CABS(B)
  D=ZERO
  F=ZERO
  ABX=CABS(X)
  DO 11 J=M,1,-1
    F=X*F+D
    D=X*D+B
    B=X*B+A(J)
    ERR=CABS(B)+ABX*ERR
11  CONTINUE
    ERR=EPSS*ERR
    IF(CABS(B).LE.ERR) THEN
      DX=ZERO
      RETURN
    ELSE
      G=D/B
      G2=G*G
      H=G2-2.*F/B
      SQ=CSQRT((M-1)*(M*H-G2))
      GP=G+SQ
      GM=G-SQ
      IF(CABS(GP).LT.CABS(GM)) GP=GM
      DX=M/GP
    ENDIF
    X1=X-DX
    IF(X.EQ.X1)RETURN
    X=X1
    CDX=CABS(DX)
    IF(ITER.GT.6.AND.CDX.GE.DXOLD)RETURN
    DXOLD=CDX
    IF(.NOT.POLISH)THEN
      IF(CABS(DX).LE.EPS*CABS(X))RETURN
    ENDIF
12  CONTINUE
    PAUSE 'too many iterations'
    RETURN
  END

```

```

SUBROUTINE GAUSSJ(A,N,NP,B,M,MP)

```

```

      INTEGER NMAX
      PARAMETER (NMAX=50)
      DIMENSION A(NP,NP),B(NP,MP),IPIV(NMAX),INDXR(NMAX),INDXC(NMAX)
      COMPLEX A,B,DUM,PIVINV,BIG
      INTEGER I,IPIV,INDXR,INDXC,J,N,NP,M,MP,K,IRCW,ICOL,L,LL
      INTRINSIC CABS
      DO 11 J=1,N
        IPIV(J)=0
11     CONTINUE
      DO 22 I=1,N
        BIG=0.
        DO 13 J=1,N
          IF(IPIV(J).NE.1)THEN
            DO 12 K=1,N
              IF (IPIV(K).EQ.0) THEN
                IF (CABS(A(J,K)).GE.CABS(BIG))THEN
                  BIG=CABS(A(J,K))
                  IROW=J
                  ICOL=K
                ENDIF
              ELSE IF (IPIV(K).GT.1) THEN
                PAUSE 'Singular matrix'
              ENDIF
12          CONTINUE
            ENDIF
13        CONTINUE
          IPIV(ICOL)=IPIV(ICOL)+1
          IF (IROW.NE.ICOL) THEN
            DO 14 L=1,N
              DUM=A(IROW,L)
              A(IROW,L)=A(ICOL,L)
              A(ICOL,L)=DUM
14          CONTINUE
            DO 15 L=1,M
              DUM=B(IROW,L)
              B(IROW,L)=B(ICOL,L)
              B(ICOL,L)=DUM
15          CONTINUE
          ENDIF
          INDXR(I)=IROW
          INDXC(I)=ICOL
          IF (A(ICOL,ICOL).EQ.0.) PAUSE 'Singular matrix.'
          PIVINV=1./A(ICOL,ICOL)
          A(ICOL,ICOL)=1.

```

```

DO 16 L=1,N
  A(ICOL,L)=A(ICOL,L)*PIVINV
16 CONTINUE
DO 17 L=1,M
  B(ICOL,L)=B(ICOL,L)*PIVINV
17 CONTINUE
DO 21 LL=1,N
  IF(LL.NE.ICOL)THEN
    DUM=A(LL,ICOL)
    A(LL,ICOL)=0.
    DO 18 L=1,N
      A(LL,L)=A(LL,L)-A(ICOL,L)*DUM
18 CONTINUE
    DO 19 L=1,M
      B(LL,L)=B(LL,L)-B(ICOL,L)*DUM
19 CONTINUE
    ENDIF
21 CONTINUE
22 CONTINUE
DO 24 L=N,1,-1
  IF(INDXR(L).NE.INDXC(L))THEN
    DO 23 K=1,N
      DUM=A(K,INDXR(L))
      A(K,INDXR(L))=A(K,INDXC(L))
      A(K,INDXC(L))=DUM
23 CONTINUE
    ENDIF
24 CONTINUE
RETURN
END

```

## References

- [1] Peter Grindrod and David J. Worth, Radionuclides and colloid migration in fractured rock: model calculations, SKI Technical Report TR 91:11, June 1990.
- [2] P.Grindrod, R.C. Brown, N.D. Gealy, COLLAGE: A numerical code for radionuclide migration through a fractured geosphere in aqueous and colloidal phases, Intera report IM2421-7 version 1, 1992.
- [3] SKI Project-90, SKI Technical Report 91:23, August 1991.
- [4] W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, Numerical Recipes, The Art of Scientific Computing, P307, CUP, Cambridge, 1987.



**SKi** STATENS KÄRNKRAFTINSPEKTION  
Swedish Nuclear Power Inspectorate

Postal address

Box 27106  
S-102 52 Stockholm

Office

Sehlstedtsgratan 11

Telephone

+46-8 665 44 00

Telex

11961 SWEATOM S

Telefax

+46-8 661 90 86