Title
DEVELOPMENT OF THE NON-EQUILIBRIUM REACTIVE CHEMICAL TRANSPORT CODE CHMTRNS

Authors
Noorishad, J.
Carnahan, C.L.
Benson, L.V.

Publication Date
1987-09-01

UC-11

LBL-22361

c.1

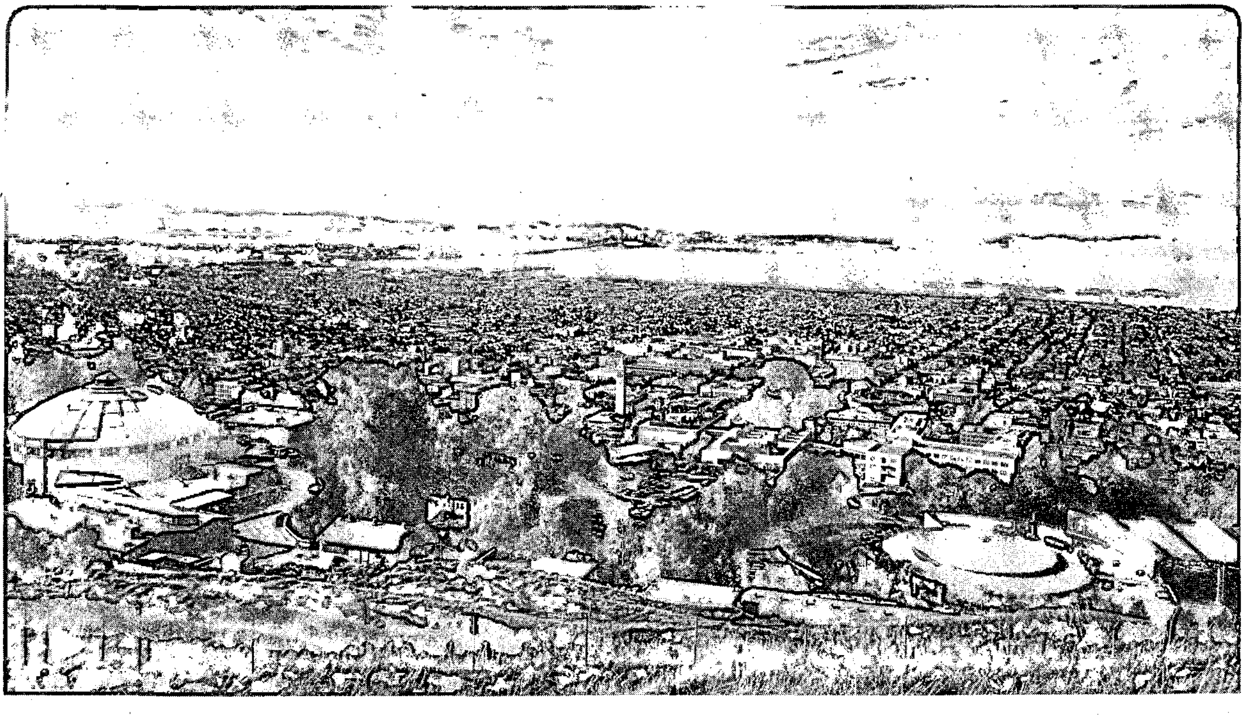
Lawrence Berkeley Laboratory
UNIVERSITY OF CALIFORNIA

EARTH SCIENCES DIVISION

Development of the Non-Equilibrium Reactive Chemical Transport Code CHMTRNS

J. Noorishad, C.L. Carnahan, and L.V. Benson

September 1987



Prepared for the U.S. Department of Energy under Contract DE-AC03-76SF00098

LBL-22361

DEVELOPMENT OF THE NON-EQUILIBRIUM REACTIVE CHEMICAL TRANSPORT CODE CHMTRNS

by

J. Noorishad and C.L. Carnahan

Earth Sciences Division
Lawrence Berkeley Laboratory
University of California
Berkeley, California 94720

and

L.V. Benson

U.S. Geological Survey
Denver, Colorado

September 1987

This work was done with support from the
U.S. Department of Energy under Contract No. DE-AC03-76SF00098

TABLE OF CONTENTS

ABSTRACT

A temperature-dependent non-equilibrium reactive chemical transport code has been developed on the basis of earlier work at Lawrence Berkeley Laboratory. The code is capable of simulating kinetic calcite and silica dissolution, irreversible glass dissolution, oxidation and reduction, and stable carbon isotope fractionation during transport. In the main part of the report necessary fundamentals and related algorithms are presented. Appendices 1–10 provide samples of inputs and outputs of solved problems. Appendices 11 and 12 include, respectively, the input preparation guide and the Fortran IV listing of the developed code CHMTRNS.

INTRODUCTION

Most reactive solute transport codes incorporate the assumption of chemical equilibrium (e.g., Miller and Benson, 1983; Kirkner et al., 1984; Cederberg, 1985; Carnahan, 1987a,b). The limits of validity of the local chemical equilibrium assumption have been explored to some degree (e.g., Palciauskas and Domenico, 1976; James and Rubin 1979), but further work is needed before the limits can be established on a firm basis. At this time, it is clear that imposing the equilibrium constraint on chemical reactions is a serious deficiency when modeling certain chemical systems. For example, comparison of theoretical computer simulations which assume chemical equilibrium with observed geothermal fluid compositions and mineral assemblages can reveal dissimilarities that are indicative of kinetic hindrance of certain reactions (Vallette-Silver et al., 1981).

For many chemical problems the slowness of reaction rates necessitates the implementation of kinetic calculations. Here, computer codes (e.g., Kirkner et al., 1985) that can simulate chemical reaction kinetics are only means to achieve realistic modeling time dependent phenomena. In the following the development of the non-isothermal, non-equilibrium, reactive chemical transport code CHMTRNS is described. This code, which is an outgrowth of the reactive chemical transport code CHEMTRN (Miller and Benson, 1983), can simulate the kinetic dissolution or precipitation of calcite and silica as well as irreversible dissolution of glass. Thermodynamic parameters are temperature dependent and are coupled to a companion heat transport simulator; thus, the effects of transient temperature conditions can be considered. Oxidation and reduction and fractionation of carbon isotopes are also options in CHMTRNS. The performance of the code CHMTRNS is illustrated in a number of simulations. The inputs and outputs, which exhibit the details of the solved cases, and at the same time serve as guides for setting up problems for solution by CHMTRNS, are given in the Appendices.

THEORETICAL BASIS

Equilibrium Formulation

The chemical transport algorithm in CHMTRNS follows that of the CHEMTRN code (Miller and Benson, 1983). However, the re-equilibration process for reversible dissolution-precipitation of solids used in CHEMTRN has been replaced by a simpler algorithm which allows determination of solid variations simultaneously with solution of the transport equations. Treatment of Neuman and Dirichlet boundary conditions and the mesh generation schemes have also been changed. An optional upstreaming capability replaces the fixed upstream algorithm. This flexibility prevents convergence problems in low ranges of Peclet number where upstreaming is not correct.

The mass balance equation for any species j can be written as

$$L\left[C_{bj} + \sum_{i=1}^{N_c} a_{ij} C_{ci}\right] + \frac{\partial}{\partial t}\left[\overline{C}_{bj} + \sum_{i=1}^{\overline{N}_c} a_{ij} \overline{C}_{ci}\right] + \sum_{i=1}^{Np} s_{ij} \frac{\partial C_{pi}}{\partial t} = 0 \quad ,$$

$$j = 1, \ldots, N_b \ , \qquad (1)$$

where the operator L is defined by

$$L(\ ) = \frac{\partial}{\partial t}(\ ) + \nabla\cdot((\ )V - D\,\nabla(\ )) \ ,$$

and

C_{bj} is the concentration of basis ion j, mol/liter

$\overline{C}_{bj}$ is the concentration of sorbed basis ion j, mol/liter of fluid phase

C_{ci} is the concentration of complex ion i, mol/liter

$\overline{C}_{ci}$ is the concentration of sorbed complex ion i, mol/liter of fluid phase

C_{pi}	is the concentration of solid i, mol/liter of fluid phase
a_{ij}	are the stoichiometric coefficients for formation of complex i
s_{ij}	are the stoichiometric coefficients for formation of solid i
V	is the fluid velocity vector equal to Darcy velocity divided by porosity ϕ
D	is the longitudinal dispersion coefficient
N_b	is the number of basis species
$\bar{N}_b$	is the number of sorbed basis species
N_C	is the number of aqueous complexes
$\bar{N}_c$	is the number of sorbed complexes
N_p	is the number of solids.

The mass action equation for the complexation reaction expressed as

$$\sum_{j=1}^{N_b} a_{ij} A_j^{n_i} \rightleftarrows B_i^{n_i} \quad , \quad i = 1,\dots,N_c \ ,$$

is

$$[B_i^{n_i}] = K_{ci} \prod_{j=1}^{N_b} [A_j^{n_j}]^{a_{ij}} \quad , \quad i = 1,\dots,N_c \ , \tag{2}$$

where $[B_i^{n_i}] = \gamma_i C_{ci}$, $[A_j^{n_j}] = \gamma_j C_{bj}$, γ_i = activity coefficient of aqueous species i calculated by the Davies equation (Stumm and Morgan, 1970), n_i and n_j are the charges of the species i, j, and K_{ci} is the equilibrium constant for formation of the complex B_i.

For reversible dissolution of a solid expressed as

$$P_i \rightleftarrows \sum_{j=1}^{N_b} s_{ij} A_j^{n_i} \ , i = 1,\dots,N_p \ ,$$

the mass action equation is

$$1 = \frac{1}{K_{Pi}} \prod_{j=1}^{N_b} [A_j^{n_j}]^{s_{ij}} \quad , \quad i = 1,\ldots,N_p \tag{3}$$

where K_{Pi} is the solubility product of the solid P_i.

For sorption by ion exchange[1] expressed as

$$n_j A_i^{n_i}(aq) + n_i \bar{A}_j \rightleftarrows n_j \bar{A}_i + n_i A_j^{n_j}(aq)$$

the algebraic equation is

$$[\bar{A}_i]^{n_j} [A_j^{n_j}]^{n_i} = K_{ij} [A_i^{n_i}]^{n_j} [\bar{A}_j]^{n_i} \tag{4}$$

where K_{ij} is the equilibrium constant, for the exchange reaction as written, A_i and A_j are activities of aqueous complexes or basis species, and $\bar{A}_i$ and $\bar{A}_j$ refer to activities of the sorbed species. The activity of a sorbed species $\bar{C}_j$ is estimated as $\bar{C}_j/\bar{C}_T$, where $\bar{C}_T$ is the total concentration of sorbed species given by

$$\bar{C}_T = \sum_{j=1}^{\bar{N}_b} \bar{C}_{bj} + \sum_{i=1}^{\bar{N}_c} \bar{C}_{ci}$$

In this sorption model it is assumed that the number of equivalents surface sites per unit volume of solution available for ion exchange, referred to as N_s, is constant. Thus,

---

[1] The algorithm for surface complexation (Davis et al., 1978) is also included in CHMTRNS in its original form taken from CHEMTRN.

$$N_s = \sum_{j=1}^{\overline{N}_b} n_j \overline{C}_{bj} + \sum_{i=1}^{\overline{N}_c} n_i \overline{C}_{ci} \ . \tag{5}$$

Variations of pH are treated by the method used in CHEMTRN. Because CHEMTRN and CHMTRNS both consider dilute solutions, it is assumed that the total mass of water (H_2O) is affected negligibly by chemical reactions, and H_2O is not included as a chemical species. However, a transport equation written for either H^+ or OH^- individually would necessarily include terms involving H_2O. To avoid this, the complete transport equation for OH^- is subtracted from the one for H^+; the result is

$$L\left[y + \sum_{i=1}^{N_c}\left(a_{i,H^+} - a_{i,OH^-}\right)C_{ci}\right] + \frac{\partial}{\partial t}\left[\overline{C}_{H^+} + \sum_{i=1}^{\overline{N}_c} a_{i,H^+}\overline{C}_{ci}\right]$$

$$+ \sum_{i=1}^{N_p}\left(s_{i,H^+} - s_{i,OH^-}\right)\frac{\partial C_{pi}}{\partial t} = 0 , \tag{6}$$

where the new variable, y, is defined by

$$y = C_{H^+} - C_{OH^-} \ .$$

The other equation needed to determine both C_{H^+} and C_{OH^-} is the mass action equation for ionization of H_2O:

$$K_w = [H^+][OH^-] \ ,$$

where K_w is the ion product H_2O, the square brackets denote activities, and the activity of H_2O is assumed equal to unity.

Non-Equilibrium Formulation

The partial differential equation of dispersive-advective mass transport does not change fundamentally for consideration of non-equilibrium chemical reactions. If basis species are produced or consumed irreversibly at a finite rate, as in kinetic dissolution of glass, a source term must be added to the transport equation for each basis species involved in the irreversible reaction. Thus, Eq. (1) becomes

$$L\left[C_{bj} + \sum_{i=1}^{N_c} a_{ij}C_{ci}\right] + \frac{\partial}{\partial t}\left[\overline{C}_{bj} + \sum_{i=1}^{\overline{N}_c} a_{ij}\overline{C}_{ci}\right]$$

$$+ \sum_{i=1}^{N_p} s_{ij}\frac{\partial C_{pi}}{\partial t} + \delta_{kj}\nu_{kb}R_b = 0, \quad j = 1,\ldots,N_b \ , \tag{8}$$

where k refers to basis species involved in the reaction and has one or more discrete values in the range $1,\ldots,N_b$, δ_{kj} is the Kronecker delta function, ν_{kb} is the stoichiometric coefficient of basis species k in the reaction, and R_b is the rate of the reaction.

On the other hand, if complexes or solids participate in non-equilibrium reactions and if no mass enters or leaves the system irreversibly, then Eq. (1) is used without modification. However, the mass action equations (2) or (3) for each participating complex or solid must be replaced by rate equations of the type

$$\frac{\partial}{\partial t}(X_i) = -R_{X_i} \ , \tag{9}$$

where X_i represents the concentration of a complex or a solid and R_{X_i} is the appropriate rate. The mass action equations for non-participating complexes and solids remain unchanged.

Oxidation-Reduction Reactions

Oxidation-reduction reactions are treated by defining a hypothetical electron activity as a basis species subject to transport as are other aqueous basis species (Carnahan, 1987b). For multivalent elements, a species in the highest oxidation state is chosen to be the basis species. Reduction to a lower oxidation state is described formally by a half-cell reaction in which the higher valent species "reacts" with hypothetical electrons to form the lower-valent species. The general form of the half-cell reaction with transfer of n electrons is

$$OX + ne^{-} = RED \ .$$

The corresponding mass action equation is

$$[RED] = K[OX]a^{n}_{e^{-}} \ ,$$

where K is the equilibrium constant, square brackets denote activities, and $a_{e^{-}}$ is the hypothetical electron activity. This process is analogous to the process by which aqueous complexes are formed by reactions between basis species. Thus, the reduced species are treated as though they are complexes of oxidized species with electrons. The electron activity is related to Eh, the oxidation potential on the hydrogen scale, by activity is related to Eh, the oxidation potential on the hydrogen scale, by

$$a_{e^-} = \exp\left(-\frac{F\ Eh}{RT}\right) ,$$

where F is the Faraday constant, R is the gas constant, and T is absolute temperature. It is noted that the electron activity is an artifice that allows treatment of reduction in the same manner as complexation; there is no implication that free electrons exist in aqueous solution.

Carbon Isotope Fractionation

Chemical reactions involving ^{13}C-rich carbonate rocks will change the relative amounts of the isotopes ^{13}C and ^{12}C within individual carbonate aqueous species. These changes are expressed as changes of the isotopic ratio, R_i, in species i, defined by

$$R_i = \frac{^{13}C_i}{^{12}C_i} ,$$

where the notation $^{m}C_i$ denote the concentration of that fraction of species i containing the isotope of mass m. Changes of R_i are measured by the isotopic shift, δ_i, defined by

$$\delta_i = 10^3\left(\frac{R_i}{R_{PDB}} - 1\right) ,$$

where R_{PDB} is the isotopic ratio of a standard.

To formulate the equations of change of isotopic composition during transport, we consider the aqueous species CO_3^{2-}, HCO_3^-, and $H_2CO_3^0$ and the solid $CaCO_3(c)$ (calcite), and denote their respective concentrations by

C_2, C_1, C_0, and C_s. The transport equation for these species is then

$$L(C_0 + C_1 + C_2) + \frac{\partial C_s}{\partial t} = 0$$

or

$$L(C_T) + \frac{\partial C_s}{\partial t} = 0 \ , \tag{11}$$

where $C_T = C_0 + C_1 + C_2$. We can write also,

$$L(^{13}C_0 + {}^{13}C_1 + {}^{13}C_2) + \frac{\partial\, {}^{13}C_s}{\partial t} = 0 \ . \tag{12}$$

Because $^{13}C_i << {}^{12}C_i$, we have a good approximation

$$^{13}C_i = R_i C_i \quad , \quad i = 0,1,2,s \ . \tag{13}$$

Also,

$$R_i = R_{PDB}(10^{-3}\ \delta_i + 1) \tag{14}$$

The net isotopic shift of the fluid, δ_{sol}, is defined by

$$\sum_{i=0}^{2} C_i \delta_i = C_T \delta_{sol} \tag{15}$$

We adopt the calcite precipitation mechanism proposed by Plummer et al., (1978), in which $CaCO_3$(c) precipitates directly from HCO_3^-(aq). Then, substituting Eqs. (13) and (14) into (12) and using Eqs. (11) and (15) gives

$$L(C_T \delta_{sol}) + \frac{\partial}{\partial t} (\delta_x C_s) = 0 \quad , \tag{16}$$

where

$$\delta_x = \delta_s \quad \text{if} \quad \frac{\partial C_s}{\partial t} < 0 \qquad \text{(dissolution case)} \ ,$$

$$\delta_x = \delta_1 \quad \text{if} \quad \frac{\partial C_s}{\partial t} > 0 \qquad \text{(precipitation case)} \ .$$

In the case of dissolution where carbonate from the rock is added to the solution Eq. (16) becomes

$$L(C_T \delta_{sol}) + \delta_s \frac{\partial C_s}{\partial t} > 0 \ . \tag{17}$$

Combining Eqs. (17) and (11) we obtain

$$L(\theta_d) = 0 \tag{18}$$

where

$$\theta_d = (\delta_{sol} - \delta_s)\ C_T$$

In the case of precipitation we can expand Eq. (16) to

$$L(C_T \delta_{sol}) + \delta_1 \frac{\partial C_s}{\partial t} + C_s \frac{\partial \delta_1}{\partial t} = 0 \ . \tag{19}$$

The shift of the bicarbonate species, δ_1, originating from fractionation during dissolution of CO_2(gas) in water, can be referenced to δ_{CO_2} in the following manner. We have

$$R_{CO_2} = R_{PDB}(10^{-3}\ \delta_{CO_2} + 1) \ .$$

The partition coefficients, α_i, are defined by

$$\alpha_i = \frac{R_i}{R_{CO_2}} \quad , \quad i = 0,1,2 \ ,$$

where i refers to $H_2CO_3^0$, HCO_3^- and CO_3^{2-} respectively. Then, using Eq. (14),

$$\delta_i = \alpha_i \delta_{CO_2} + 10^3(\alpha_i - 1) \quad , \quad i=0,1,2 \ . \tag{20}$$

We can also reference the aqueous carbonate species to each other and eliminate δ_{CO_2} from Eq. (20). If we choose $H_2CO_3^0$ (i=0) as the reference species we get

$$\delta_1 = \frac{\alpha_1}{\alpha_0} \delta_0 + 10^3 \left(\frac{\alpha_1}{\alpha_0} - 1\right) = A_1\delta_0 + A_2 \ , \tag{21a}$$

$$\delta_2 = \frac{\alpha_2}{\alpha_0} \delta_0 + 10^3 \left(\frac{\alpha_2}{\alpha_0} - 1\right) = A_3\delta_0 + A_4 \ . \tag{21b}$$

Substituting Eqs. (21a) and (21b) into Eq. (15)and solving for δ_0 gives

$$\delta_0 = \frac{C_T\delta_{sol} - 10^3\left[C_1\left(\frac{\alpha_1}{\alpha_0} - 1\right) + C_2\left(\frac{\alpha_2}{\alpha_0} - 1\right)\right]}{C_0 + \left(\frac{\alpha_1}{\alpha_0}\right) C_1 + \left(\frac{\alpha_2}{\alpha_0}\right) C_2} = \frac{C_T\delta_{sol} - A_5}{A_6} \tag{22}$$

Defining $\theta_p = C_T\delta_{sol}$ and substituting Eq. (22) into Eq. (21a) we obtain

$$\delta_1 = \frac{A_1}{A_6} \theta_p - \left(\frac{A_1A_5}{A_6} - A_2\right) = A_7\theta_p + A_8 \tag{23}$$

where all A_i parameters are defined in terms of concentrations of carbonate species and α_i i = 0,1,2. The latter partition coefficients are given (Deines et al., 1974) as functions of absolute temperature, T, by

$$\ln \alpha_0 = -0.91 \times 10^{-3} + 6.3/T^2$$

$$\ln \alpha_1 = -4541 \times 10^{-3} + 1099/T^2$$

$$\ln \alpha_2 = -3.4 \times 10^{-3} + 870/T^2$$

Substituting Eq. (23) into Eq. (19) results in

$$L(\theta_p) = (A_7\theta_p + A_8) \frac{\delta C_s}{\delta t} + C_s \frac{\delta}{\delta t} (A_7\theta_p + A_8) \tag{24}$$

Equation (18) and (24) are partial differential equations of transport for the new variables θ_d and θ_p that keep track of $\delta^{13}C$ in the solution, in cases of dissolution or precipitation of calcite respectively. Unlike Eq. (18), Eq. (24) (which can be further rearranged by explicit writing of the L operator) is a nonlinear equation coupled to the other reactive transport equations through the dependence of the "A_i" parameters on the concentrations of carbonate species. However, the solution of Eqs. (18) and (24) poses no difficulty and has been implemented along the same lines of approach explained in the section on the numerical solution of the reactive transport problem.

Heat Transport Formulation

The formulations presented so far are generally based on the assumption of isothermal conditions. However, considering the role of variation of temperature in many geochemical processes, it is necessary to include thermal effects in chemical reactions and to formulate heat transport as a companion to mass transport. Removing the restriction of constancy of temperature does not alter the mass transport formulation in any way. The only necessary addition is to formulate the functional relationship between the thermodynamic constants and temperature. The relationship used has the thermodynamic equation of Clarke and Glew (1966):

$$\log_{10}K = A + \frac{B}{T} + C\log_{10}T + DT$$

where K is the thermodynamic equilibrium constant used in any mass action relation, A, B, C and D are parameters that vary for different chemical reactions, and T is the absolute temperature.

The governing equation of heat transport is

$$(\rho C)_M \frac{\partial T}{\partial t} + \phi\rho_\ell C_{v\ell}\, V\cdot\nabla\, T - \nabla\cdot(K_M\, \nabla\, T) + Q = 0 \ , \tag{25}$$

where

$$(\rho C)_M = \phi\rho_\ell C_{v\ell} + (1 - \phi)\ \rho_s C_{vs} \ ,$$

$$K_M = \phi K_\ell + (1 - \phi)\ K_s \ ,$$

$$\rho_\ell = \rho_o + \beta_t(T - T_0) \ ,$$

in which $C_{v\ell}$ and C_{vs} are fluid and solid specific heat capacities at constant volume, ρ_ℓ is the liquid mass density, K_ℓ and K_s are the thermal conductivities of the fluid and the solid, β_t is the fluid thermal expansion coefficient, ϕ is the porosity of the medium, and ρ_o and T_o are the initial fluid density and temperature, and v is the pore fluid velocity.

SOLUTION APPROACH

Differential Eqs. (8) for mass balance and aqueous species transport, rate Eq. (9) for non-equilibrium chemical reactions, differential Eq. (6) for transport of hydrogen and hydroxide ions, differential Eqs. (18) or (24) for isotopic fractionation tracking during transport, algebraic Eqs. (2)-(5) for equilibrium complexation, dissolution or precipitation, and ion exchange, algebraic Eq. (10) for oxidation-reduction reactions, and finally Eq. (25) for dispersive-conductive heat transport, provide the complete set of differential-algebraic equations for the coupled processes of non-isothermal equilibrium and non-equilibrium reactive chemical transport with oxidation-reduction and carbon isotope tracking options. Because of its complexity, the set of equations is only amenable to numerical solution. One-way coupling between the energy equation and the other equations makes an interlaced solution approach possible. With this approach, the differential-algebraic equations of chemical transport are solved separately from the heat equation. Temperatures calculated from the solution of the heat equation at each time level are used to update all temperature-dependent parameters for the next time step, and this scheme continues through the solution. The usually large time constant of the heat equation eliminates the need for iteration between the heat and transport equations.

Detailed one-dimensional space and time discretization of the reactive chemical transport equations will be addressed in this section. Because the heat transport equation is very similar to the mass transport equation, with the exception of parameters, no separate explanation is offered. However, as will be discused later, the solution method for the heat equation is much simpler because of the justifiable assumption of its linearity.

Space and Time Discretization

For a one-dimensional finite difference numerical solution we can choose either a uniform or nonuniform spatial discretization of the flow path. For uniform discretization, the location, x_i, of finite difference node i is computed from

$$x_i = \frac{i}{N-1} L_P \quad , \quad i = 1,\ldots,N \ ,$$

while for nonuniform discretization x_i is computed from

$$x_i = \frac{d^{\left(\frac{i-1}{N-1}\right)} - 1}{d-1} L_P \quad , \quad i = 1,\ldots,N \ ,$$

where L_P is the flow path length, N is the number of finite difference nodes, and d is an arbitrary number. If d is chosen to be greater than one, the interval $x_i - x_{i-1}$ increases as x_i increases.

The reactive mass transport equations are approximated by replacing first the spatial derivatives with finite difference approximations. Using

$$W_{j,n} = \left[C_{bj} + \sum_{i=1}^{N_c} a_{ij} C_{ci} \right]_n \tag{26}$$

$$\overline{W}_{j,n} = \left[\overline{C}_{bj} + \sum_{i=1}^{\overline{N}_c} a_{ij} \overline{C}_{ci} \right]_n \tag{27}$$

to represent respectively the total masses of aqueous and adsorbed species at grid point n, the transport Eq. (8) is rewritten as

$$\frac{d}{dt} W_{j,n} + \frac{d}{dt} \overline{W}_{j,n} + \frac{d}{dt} \sum_{i=1}^{N_P} s_{ij} C_{Pi,n}$$

$$+ V\left[\left((1-\gamma)\frac{x_n - x_{n-1}}{x_{n+1} - x_{n-1}}\right)\left(\frac{C_n^{\ell+1} - C_{n-1}^{\ell+1}}{x_n - x_{n-1}}\right)\right.$$

$$\left. + \left(\gamma \frac{x_n - x_{n-1}}{x_n - x_{n-1}}\right)\left(\frac{C_{n+1}^{\ell+1} - C_n^{\ell+1}}{x_{n+1} - x_n}\right)\right]$$

$$+ \frac{D}{0.5*(x_{n+1} - x_{n-1})}\left[\left(\frac{W_{j,n+1}^{\ell+1} - W_{j,n}^{\ell+1}}{x_{n+1} - x_n}\right) - \left(\frac{W_{j,n}^{\ell+1} - W_{j,n-1}^{\ell+1}}{x_n - x_{n-1}}\right)\right]$$

$$+ \delta_{kj}\nu_{kb}R_b = 0 \quad , \quad j = 1,\ldots,N_b, \quad n = 1,\ldots,N \, , \tag{28}$$

where $\ell + 1$ refers to the time level in which we seek a solution and γ is an upwind parameter between zero and one. Equations (2)-(5) and (9) remain basically the same except that they are referred to grid points:

$$[B_{i,n}^{n_i}] = K_{C_{i,n}} \prod_{j=1}^{N_b} [A_{j,n}^{n_j}]^{a_{ij}} \quad , \quad i = 1,\ldots,N_c \quad , \quad n = 1,\ldots,N \, , \tag{29}$$

$$1 = \frac{1}{K_{Pi,n}} \prod_{j=1}^{N_b} [A_{j,n}^{n_j}]^{s_{ij}} \quad , \quad i = 1,\ldots,N_P \quad , \quad n = 1,\ldots,N \, , \tag{30}$$

$$[\bar{A}_{i,n}]^{n_j}\,[A_{j,n}^{n_j}]^{n_i} = K_{ij,n}[A_{i,n}^{n_i}]^{n_j}\,[\bar{A}_{j,n}]^{n_i} \qquad n = 1,\ldots,N\ , \tag{31}$$

$$N_s = \sum_{j=1}^{\bar{N}_b} n_j \bar{C}_{bj,n} + \sum_{i=1}^{\bar{N}_c} n_i \bar{C}_{ci,n} \qquad n = 1,\ldots,N\ , \tag{32}$$

$$\frac{d}{dt}\,(X_{i,n}) = -R_{X_{i,n}}\ , \qquad n=1,\ldots,N\ . \tag{33}$$

For the sake of clarity we omit writing discretized forms of Eqs. (7) and (10), which will be similar to Eqs. (30) and (29), respectively. Because of their temperature dependence, the equilibrium constants have also been indexed. Among the above equations, only Eqs. (28) and (33) will require time discretization as well. Using a backward Euler's method (fully implicit) the time derivatives are approximated as follows:

$$\frac{d}{dt}\,W_{j,n} \approx \frac{W_{j,n}^{\ell+1} - W_{j,n}^{\ell}}{\Delta t}$$

$$\frac{d}{dt}\,\bar{W}_{j,n} \approx \frac{\bar{W}_{j,n}^{\ell+1} - \bar{W}_{j,n}^{\ell}}{\Delta t}$$

$$\frac{d}{dt}\,C_{Pi,n} \approx \frac{C_{Pi,n}^{\ell+1} - C_{Pi,n}^{\ell}}{\Delta t}$$

$$\frac{d}{dt}\,X_{i,n} = \frac{X_{i,n}^{\ell+1} - X_{i,n}^{\ell}}{\Delta t} \tag{34}$$

Equations (28)-(33) in conjunction with the time discretization expressions in Eq. (34) are a set of strongly nonlinear simultaneous algebraic equations that can be written in matrix form as

$$F(\underline{X}) = A(\underline{X})\ \underline{X} - B(\underline{X}) = 0 \tag{35}$$

where F is the residue vector, $\underline{x}$ is the vector of unknows, A and B are coefficient matrices, the number of unknowns, N_x, is equal to

$$N_x = (N_b + N_c + \overline{N}_b + \overline{N}_c + N_p) * N$$

and N is the number of grid points.

In the above discretization we used Cartesian coordinates. However, discretization for axially symmetric space can be achieved simply by replacing x_n, V and D in Eq. (28) with r_n, $V_0 \frac{R}{r_n}$ and $D = D(r_n)$ respectively where V_0 is the pore fluid velocity at radius R. The final discretized form of equations [i.e., Eq. (35)] will not be affected. Discretization of the heat equation can be performed similarly, leading to a set of simultaneous equations of the general form of Eq. (35). For the small temperature variations considered here the extent of nonlinearity in the heat equations is small. Hence, we assume linearity of Eq. (35) for heat transport.

Implementation of Boundary Conditions

In dealing with boundary conditions the main concern is proper treatment of the spatial derivatives. In the following derivations we illustrate the method by using a simple form of Eq. (1) for a single species of concentration C and consider a backward time difference scheme.

Constant flux boundary condition (Crank, 1975) - Consider a boundary B at x=0 where the following governing equation of transport and flux boundary condition hold:

$$\frac{\partial C}{\partial t} - D \frac{\partial^2 C}{\partial x^2} + V \frac{\partial C}{\partial x} = 0 \tag{36}$$

$$\left[- D \frac{\partial C}{\partial x} + VC \right]_{x=0} = F \tag{37}$$

where F is the prescribed flux at the boundary. Using the difference approximations in the above equations at the boundary B results in

$$\frac{C_0^\ell - C_0^{\ell-1}}{\Delta t} - \frac{D}{(\Delta x)^2} (C_{-1}^\ell - 2C_0^\ell + C_1^\ell) + \frac{V}{2\Delta x} (C_1^\ell - C_{-1}^\ell) = 0 \quad . \tag{38}$$

$$C_{-1}^\ell = C_1^\ell + \frac{2\Delta x}{D} (F - VC_0^\ell) \tag{39}$$

where indices -1, 0 and 1 designate an image node, the boundary node, and the first internal node, and Δx is the discretization distance, assumed uniform in this illustration. Combining Eqs. (38) and (39) to eliminate C_{-1}^ℓ and rearranging we obtain

$$\frac{C_0^\ell - C_0^{\ell-1}}{\Delta t} - 2D\left[\frac{C_1^\ell - C_0^\ell}{(\Delta x)^2}\right] + \left[\frac{VC_0^\ell - F}{\Delta x}\right] + \left(2 + \frac{V\Delta x}{D}\right) = 0 \quad . \tag{40}$$

Eq. (40) is the discretized form of the transport equation at the boundary B. For axially symmetric radial mass transport, we use the same arrangement for the imaginary point except that Δr replaces Δx. In this case the governing equation and boundary flux condition are given in the following form

$$\frac{\partial C}{\partial t} - D(r)\frac{\partial^2 C}{\partial r^2} + \left[V(r) - \frac{D_d}{r}\right]\frac{\partial C}{\partial r} = 0 \ , \tag{41}$$

$$\left[- D(r)\frac{\partial C}{\partial r} + V_c\right]_{r=R} = F \ , \tag{42}$$

where R is the radial coordinate of the boundary point B, D(r) is equal to $\partial_L V(r) + D_d$, ∂L is longitudinal dispersivity, D_d is the molecular diffusion coefficient, V(r) is equal to $V_0 \frac{R}{r}$, and V_0 is the pore fluid velocity at the boundary. Again using difference approximations in Eqs. (41) and (42) and eliminating the image concentration value we obtain

$$\frac{C_0^{\ell} - C_0^{\ell-1}}{\Delta t} - 2D(R)\left(\frac{C_1^{\ell} - C_0^{\ell}}{(\Delta r)^2}\right) + \left(\frac{VC_0^{\ell} - F(R)}{\Delta r}\right)\left(2 + \frac{V_0\Delta r}{D(R)} - \frac{D_d\Delta r}{RD(R)}\right) = 0 \ , \tag{43}$$

which is the discretized form of the governing equation at a flux boundary point. With exception of the contribution of D_d in the last term, the Eqs. (43) and (40) have the same form. Either of these equations replace the part of Eq. (28) related to mobile species, with C_0^{ℓ} and C_1^{ℓ} being replaced by $W_{j,1}^{\ell+1}$ and $W_{j,2}^{\ell+1}$ respectively, for boundary condition implementation.

<u>Constant concentration boundary conditions</u> - Implementation of this condition requires only simple manipulation of the matrix equations, Eq. (35). The equations for the known concentration boundary conditions are simply decoupled from those for the neighboring points. In the direct solution method used here it is enough to make all the

nondiagonal terms very small and the diagonal terms in the A matrix equal to unity. In the Newton-Raphson method, discussed later, we reduce both the residue for the corresponding boundary equations and the nondiagonal terms to very small quantities, thus assuring a correction closer to zero for the known quantities.

Solution Methods

As discussed earlier, time and space discretization of the governing equations for both reactive chemical transport and heat transport lead to matrix equations of the form presented in Eq. (35). For heat transport we have used the Crank-Nicholson approximation rather than fully implicit methods; however, the final form is the same. The almost linear form of the equations for heat transport, due to small variations of A and B during a time step, makes them amenable to a one-step direct solution method. We used an LU decomposition and backward substitution solution procedure in conjunction with the tridiagonal form of these equations.

In case of the reactive chemical transport equations, the strong nonlinearity of Eq. (35) makes the use of an iterative scheme imperative. We have used a generalized Newton-Raphson method which leads to following solution algorithm for Eq. (35):

$$\underline{X}_i = \underline{X}_{i-1} - \Delta\underline{X}_i \tag{44}$$

$$\underline{J}\ (\underline{X}_{i-1})\ \Delta\underline{X}_i = \underline{F}\ (\underline{X}_{i-1}) \tag{45}$$

$$J(\underline{X}_{i-1}) = \left(\frac{\partial}{\partial\underline{X}}\ F(\underline{x})\right)_{i-1} \tag{46}$$

where i represents the iteration level and the Jacobian $\underline{J}$ is a function of both time and space. In the CHMTRNS approach $\underline{J}$ is evaluated numerically at each time step. Equation (45) is solved by LU decomposition and backward substitution in conjunction with its block tridiagonal nature (Miller and Benson, 1983). The convergence criterion for this iterative scheme is set by

$$||\delta Q|| \leq 10^{-4} ,$$

where the norm Q is defined as

$$||\delta Q|| = \left[\sum_{n=1}^{N} \sum_{j=1}^{N_T} \left(\frac{C_{xj,n}^{\ell+1,k+1} - C_{xj,n}^{\ell+1,k}}{\mathrm{max}j C_{xj,n}^{\ell+1,k+1}} \right)^2 \right]^{1/2} ,$$

N_T is the total number of degrees of freedom at each node, equal to $N_b + N_c + \bar{N}_b + \bar{N}_c + N_p$, N is the number of mesh points, and k represents the iteration level. As in CHEMTRN (Miller and Benson, 1983), the elements of the Jacobian matrix are calculated numerically at each step by calculating the ratios of the variation of each residue F to ΔC, a small, arbitrary change of concentration. However, to save computation time during the solution process in CHMTRNS, two measures are taken. (1) The spatial part of the Jacobian is only calculated once and is stored for repeated use. (2) The Jacobian is not recomputed at each iteration. If a solution is not obtained within a specified number of iterations, then a new Jacobian is calculated and the process is repeated.

Treatment of Dissolution and Precipitation

Among the set of algebraic equations to be solved in the N-R numerical scheme, the solubility product equation does not include any direct representation of the solid for which it is formulated. Therefore, the diagonal elements of the Jacobians of such equations are not automatically determined in the numerical evaluation of the Jacobian matrix. In the cases of supersaturation or undersaturation when solid exists in contact with the solution, no numerical problem is posed. However, in the case of undersaturation and lack of enough available solid for dissolution, the terms of the Jacobian for the solubility (solid) equations must be modified. In this case the goal is to preserve the undersaturated state while not creating negative solid concentrations. To achieve this, we put all the non-diagonal terms of the Jacobian matrix, for that solid, equal to zero and assign a value of unity to the diagonal term. This uncouples the solid equation from the undersaturated aqueous species equations and prevents creation of negative concentrations of solids.

APPLICATIONS

We have used CHMTRNS, to simulate a number of problems in kinetics, oxidation-reduction, and ^{13}C-fractionation in the process of transport along fluid-flow paths in geologic systems. The main purpose has been to test the performance of the CHMTRNS code and verify it whenever possible. However, due to unavailability of other comparable methods, as is often the case for problems of the degree of complexity addressed here, the soundness of the code in such cases can only be judged through its performance in various trials. The performance of the CHMTRNS thus far has created grounds for a high level of confidence in its soundness.

Simulation of Reversible Non-equilibrium Dissolution of Calcite in CO_2-Water Systems

Formulation - Plummer et al. (1978) suggest a net rate of calcite dissolution (R) of the following form

$$R = \frac{dC_{Ca^{2+}}}{dt} = k_1 a_{H^+} + k_2 a_{H_2CO_3} + k_3 a_{H_2O} - k_4 a_{Ca^{2+}} a_{HCO_3^-} \tag{47}$$

where k_1, k_2 and k_3 are first order rate constants dependent on temperature, k_4 is a function of temperature and P_{CO_2}, and R is in units of mmol of calcite per cm^2 per second. The following three mechanisms occur simultaneously during dissolution

$$CaCO_3 + H^+ \rightleftarrows Ca^{2+} + HCO_3^-$$

$$CaCO_3 + H_2CO_3 \rightleftarrows Ca^{2+} + 2HCO_3^-$$

$$CaCO_3 + H_2O \rightleftarrows Ca^{2+} + HCO_3^- + OH^-$$

The constants in the rate equation are given as follows

$$\log k_1 = 0.198 - 444./T \ ,$$

$$\log k_2 = 2.84 - 2177./T \ ,$$

$$\log k_3 = -5.86 - 317/T \quad \text{for} \quad 278 \text{ K} \le T < 298 \text{ K} \ ,$$

$$\log k_3 = -1.10 - 1737./T \quad \text{for} \quad 298 \text{ K} \le T \le 321 \text{ K} \ ,$$

$$k_4 = \frac{K_2}{K_C}\left[15\ K_1 + \frac{1}{a_{H^+_{(s)}}}\left(k_2 a_{H_2CO_3} + k_3 a_{H_2O}\right)\right] \ ,$$

$$a_{H^+_{(s)}} = \left(\frac{K_1 K_2 a_{Ca^{2+}} a_{H_2CO_3}}{K_c}\right)^{1/2} \ ,$$

$$\log K_1 = -40.366 + 5576.57/T + 11.331 \log T + 0.034655\ T \ ,$$

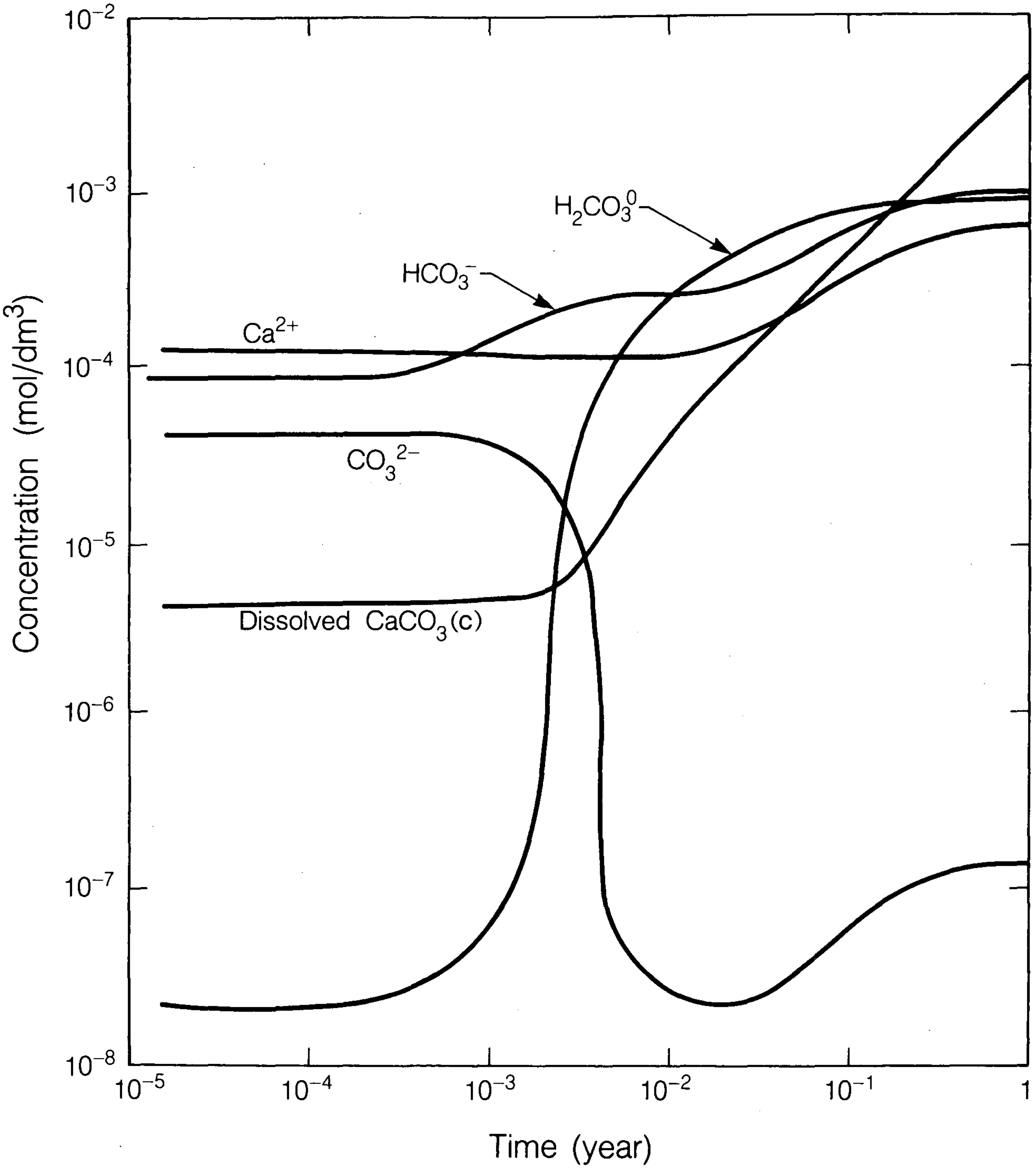
$$\log K_2 = -77.3723 + 4593.31/T + 29.042 \log T + 5.9438 \times 10^{-4}\ T \ ,$$

$$\log K_c = -8.446 \ ,$$

$$\log K_w = -80.3723 - 1395.46/T + 32.5514 \log T - 0.0318065\ T \ .$$

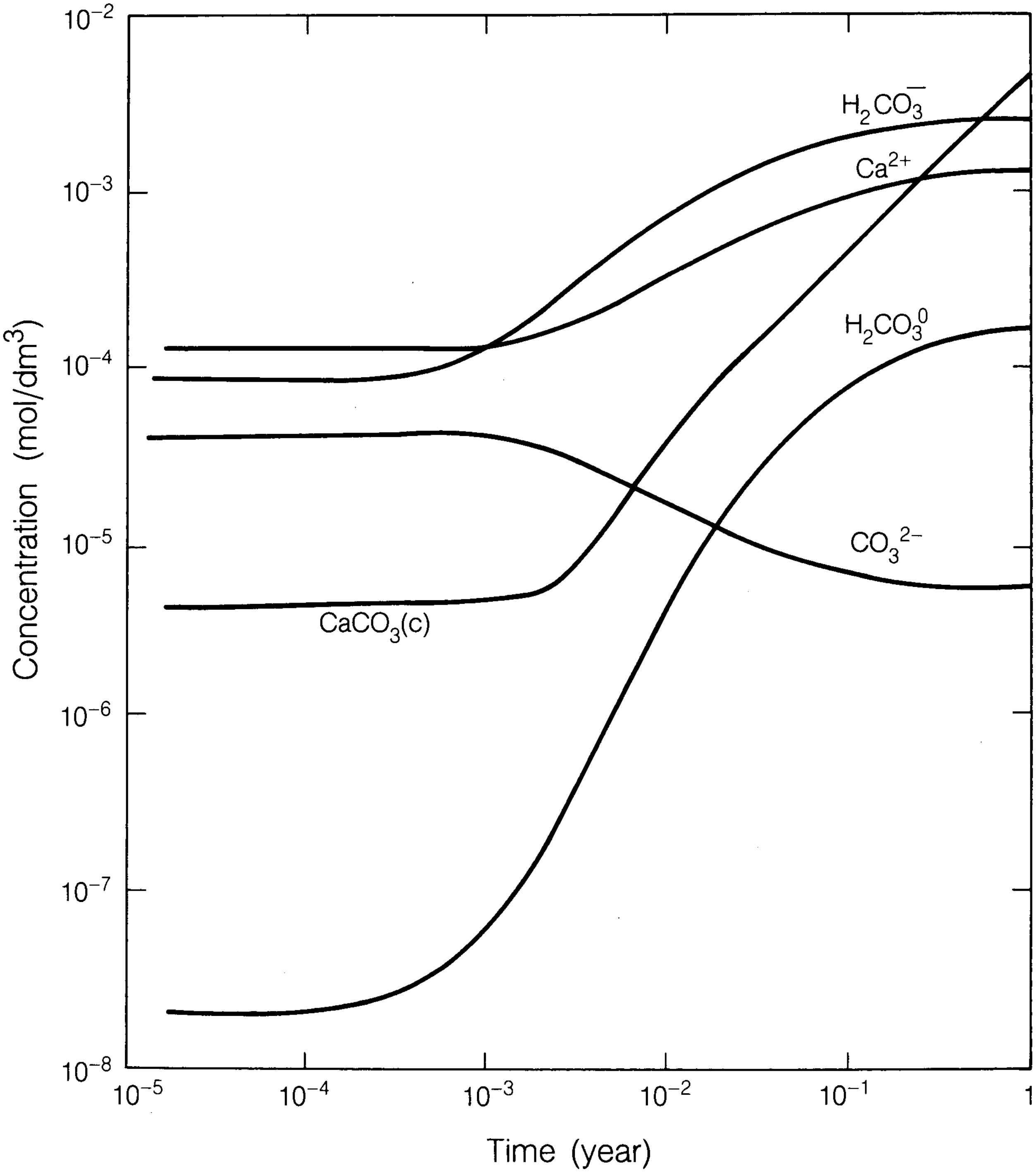
With the above parameters defined, the rate term R_{X_i} in Eq. (8) is also defined. In the code, the rate (R) is quantified in subroutine SRATE and its value is fed to the right-hand side of Eq. (35).

Dissolution in a calcite lined tube - In this problem we assumed that recharge water, with pH of 5, at equilibrium with soil-zone CO_2 ($P_{CO_2} = 3.1623 \times 10^{-2}$ atm) flows through a calcite-lined tube. A preprocessing calculation, using CHMTRNS in static mode, provided the composition of this water (see Appendix 1 for input and output). The fluid, with the calculated concentrations $C_{H_2CO_3} = 1.473 \times 10^{-3}\underline{M}$, $C_{HCO_3^-} = 5.637 \times 10^{-5}\underline{M}$ and $C_{CO_3^{2-}} = 1.865 \times 10^{-10}\underline{M}$, enters the calcite-lined tube with a velocity of 10^{-3} m/year. The tube length is 10 meters, variably discretized in 20 segments. The simulation time is 1 year. The input for this problem along with the output are included in Appendix 2. For performance demonstration purposes the results at a point near the entry of the tube are presented, where events can be observed for long periods without interference by effects due to the exit boundary. For the simulation time used here no appreciable exit boundary changes were observed, so an infinite tube-like behavior was obtained. Figure 1 shows the variation of concentrations of the aqueous species and the dissolved calcite with time. Equilibrium is reached at 1 year. However, the solid keeps dissolving continuously. To appreciate the role of kinetics in transport we solved the same transport problem with the assumption of total equilibrium. The results in Fig. 2 show the dramatic difference between the two cases. We also preformed a batch (no transport) non-equilibrium



XBL 8610-12658

Fig. 1. Breakthrough of the aqueous species at a point near the inflow boundary during kinetic transport.



XBL 8610-12659

Fig. 2. Breakthrough of the aqueous species at a point near the inflow boundary during equilibrium transport.

calcite dissolution in pure water of pH = 5, which is shown in Fig. 3. Figures 4-6 compare the three mentioned cases for various species. As expected, the results of the equilibrium transport case always converge to those of the batch kinetic dissolution case. The inputs and outputs of the latter problems are given in Appendices 3 and 4.

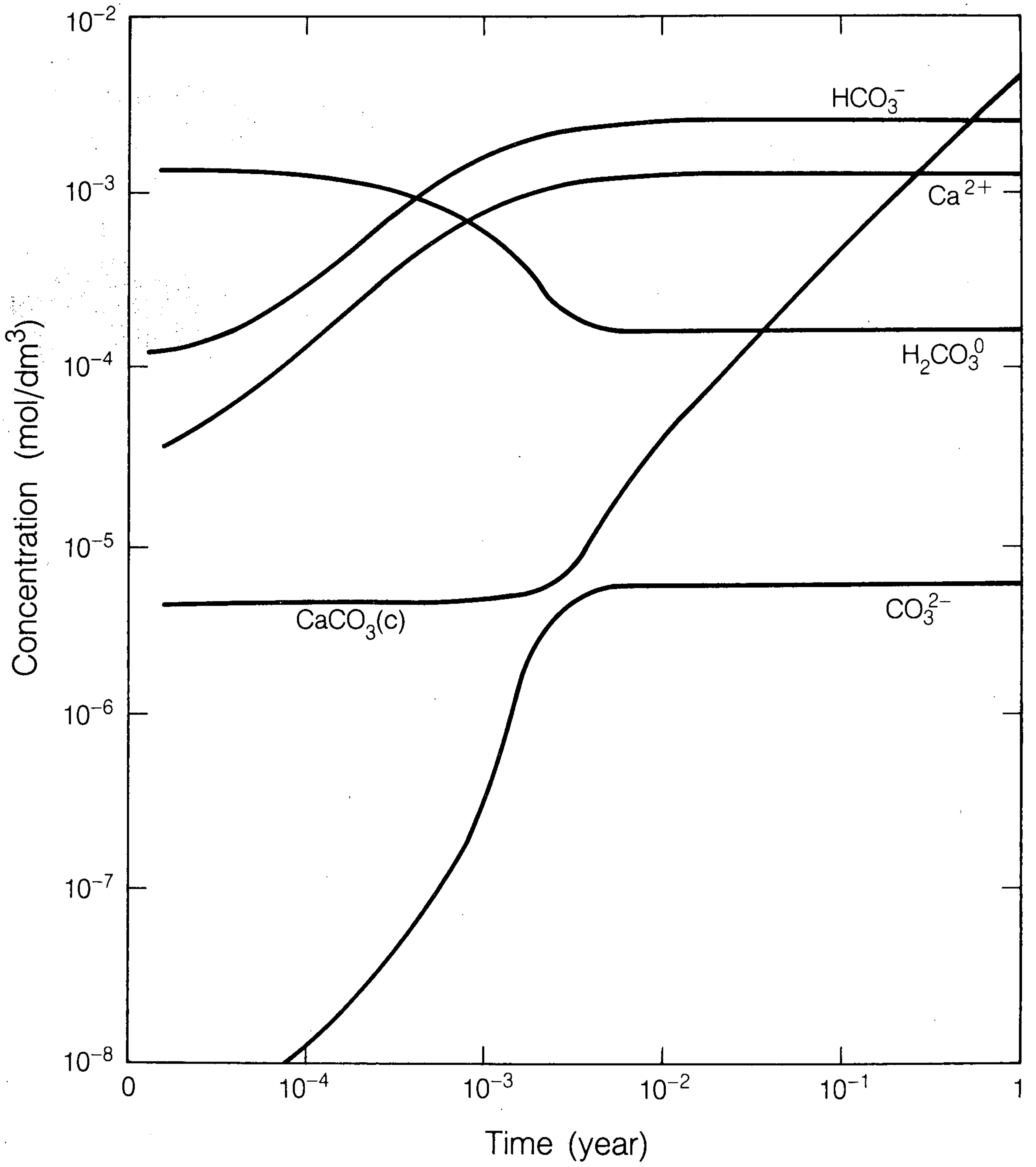
<u>Precipitation in a smectite-lined tube</u> - In this problem a water saturated with Ca^{2+} and also rich in Na^+ flows into a long smectite tube where ion exchange between the lining and the solutes can occur. The smectite initially is loaded with calcium which is then displaced by Na^+ in the influent water. Simulation time is 0.5 year. Input and output to this problem are given in Appendix 5. Figure 7 presents results for various species during the dissolution process. As can be observed in this figure, the solid builds up due to the release of Ca^{2+} from the $\overline{Ca}$ sites, and its subsequent kinetic precipitation (curve labeled $\Delta CaCO_3$ (c). The curve labeled $CaCO_3$ (c), shows the build up of the total amount of calcite in the tube.

<u>Simulation of Kinetic Silica-Water Reactions</u>

<u>Formulation</u> - Rimstidt and Barnes (1980) present a differential rate equation for silica-water reactions from 0–30°C of the following form:

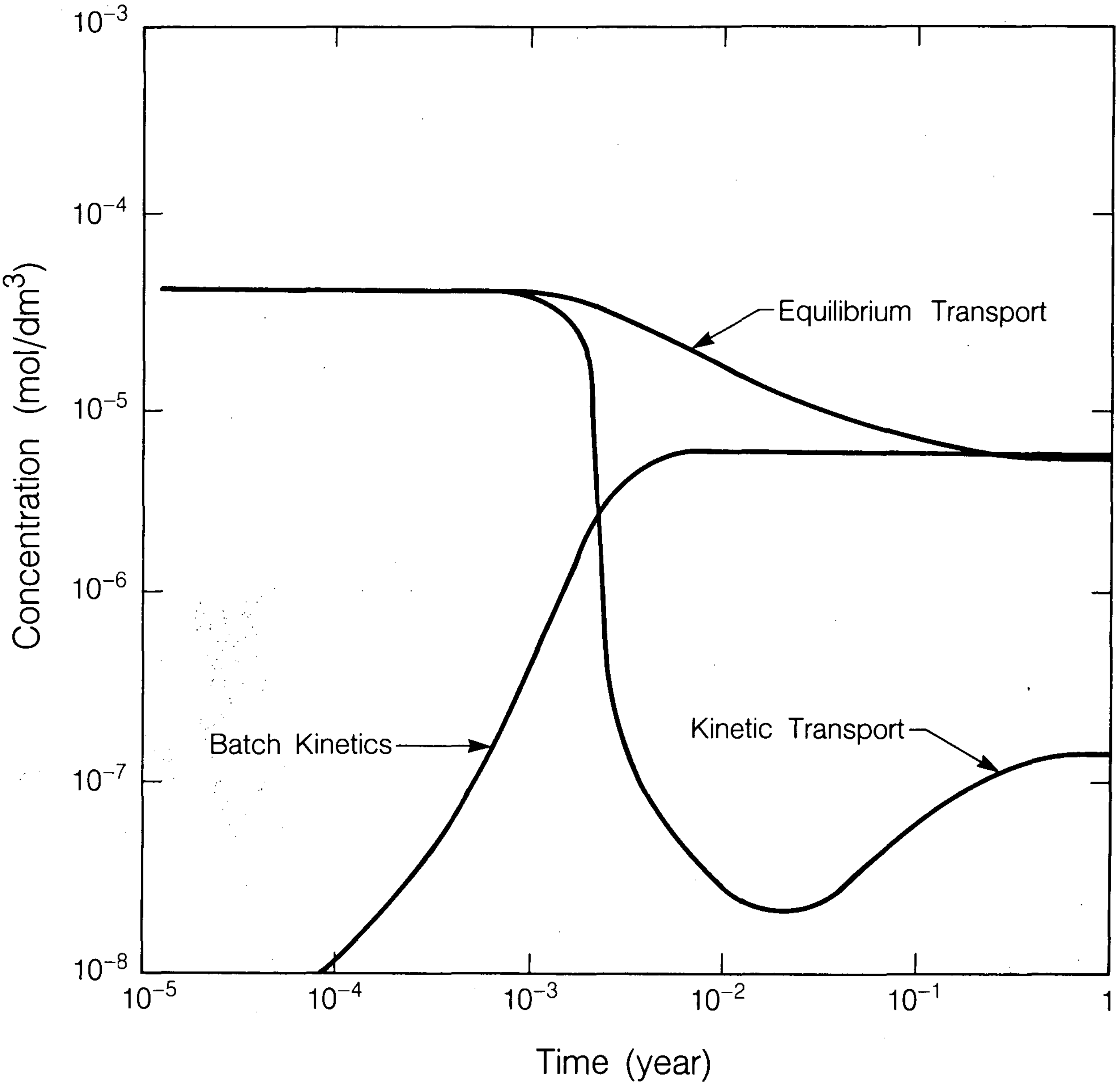
$$R = \frac{da_{H_4SiO_4}}{dt} = (A/M)(\gamma_{H_4SiO_4})(k_+ a_{SiO_2} a^2_{H_2O} - k_- a_{H_4SiO_4}) \qquad (47)$$

where (A/M) is the ratio of interfacial area between the solid and aqueous phases to the mass of water in the system, $\gamma_{H_4SiO_4}$ is the activity of silica and k_+ and k_- are the rate constants for dissolution and



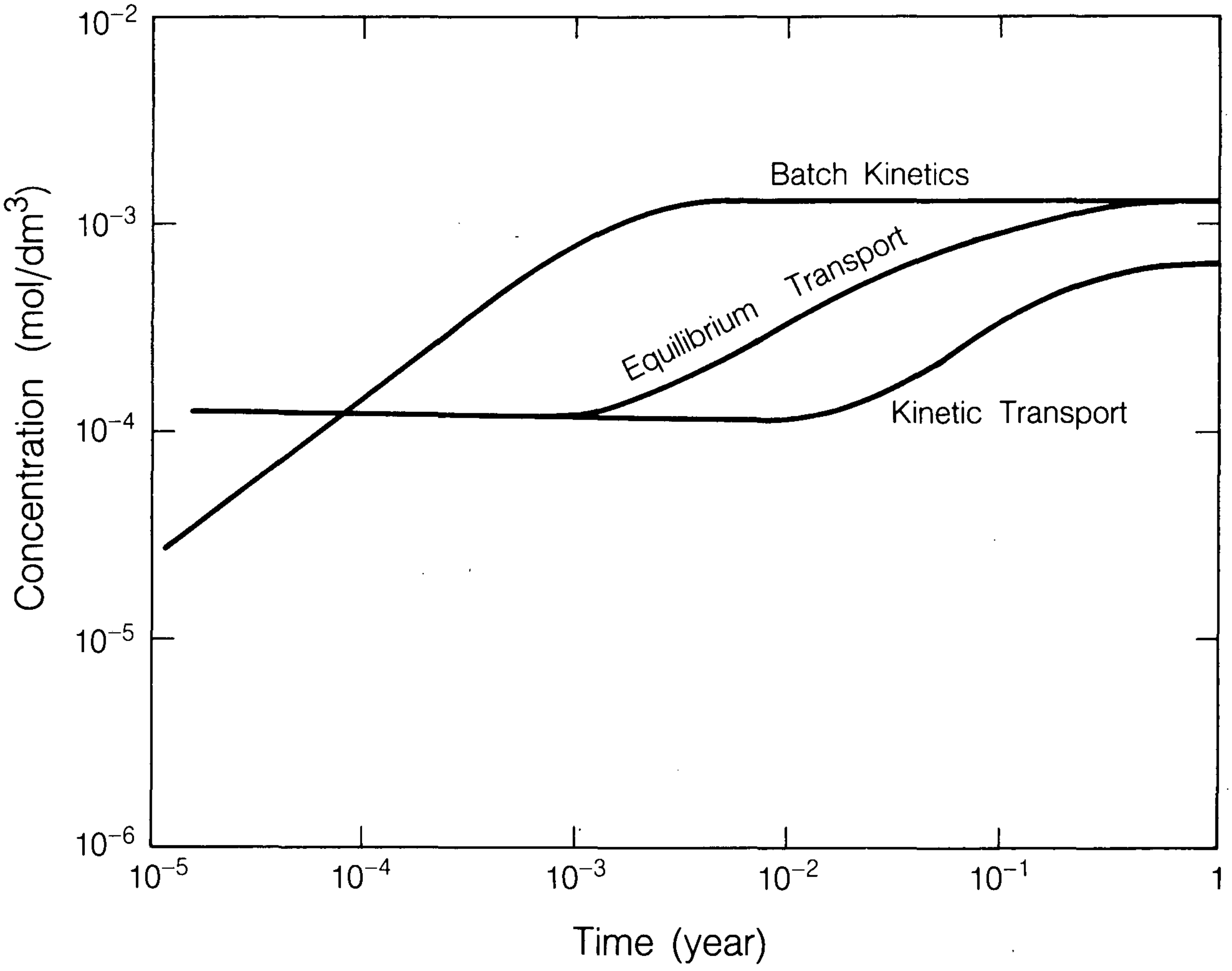
XBL 8610-12660

Fig. 3. Variation of concentration of aqueous species vs. time in batch kinetic dissolution of calcite.



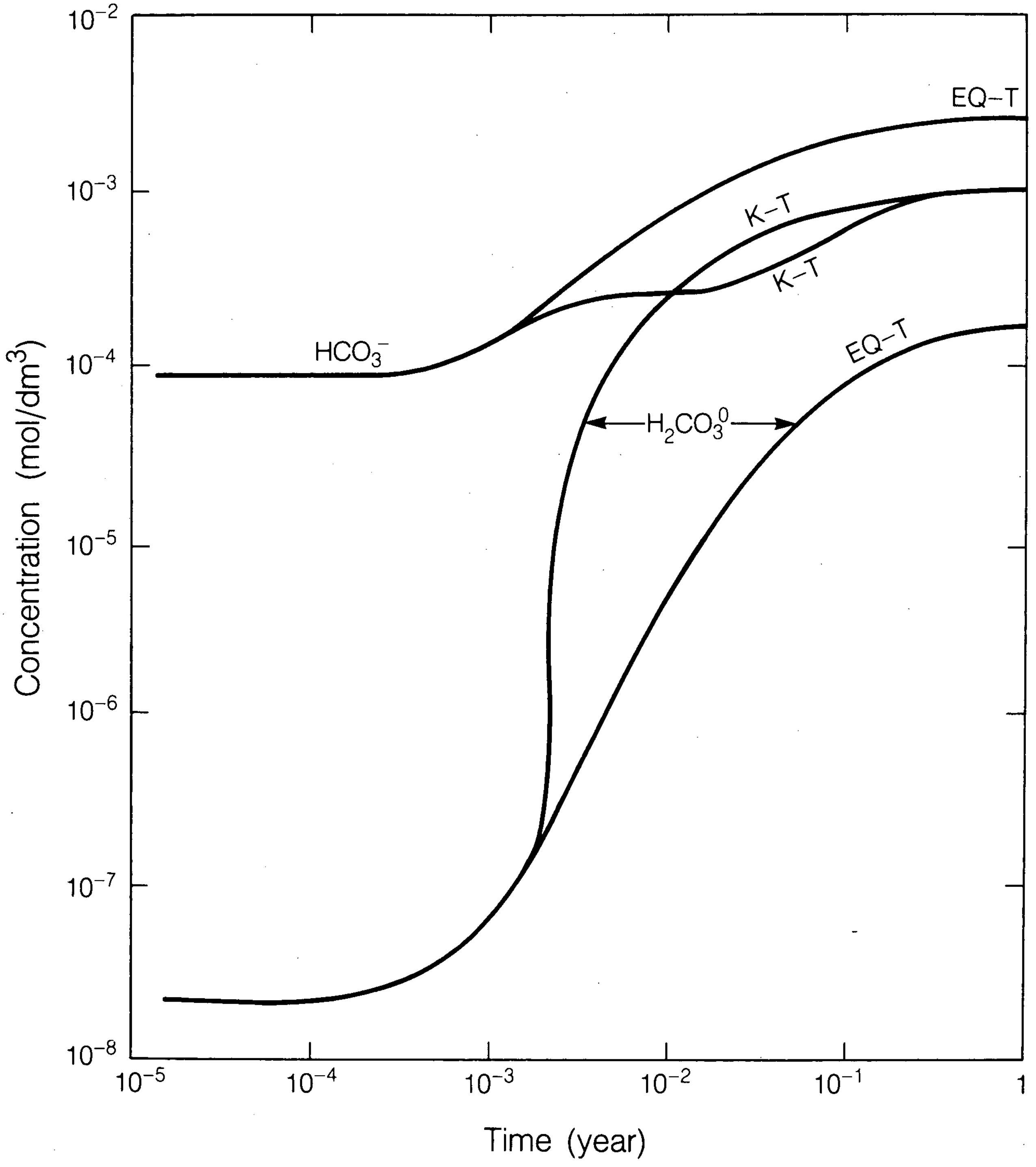
XBL 8610-12650

Fig. 4. Variation of CO_3^{2-} concentration vs. time in batch kinetics, equilibrium transport and kinetic transport scenarios.



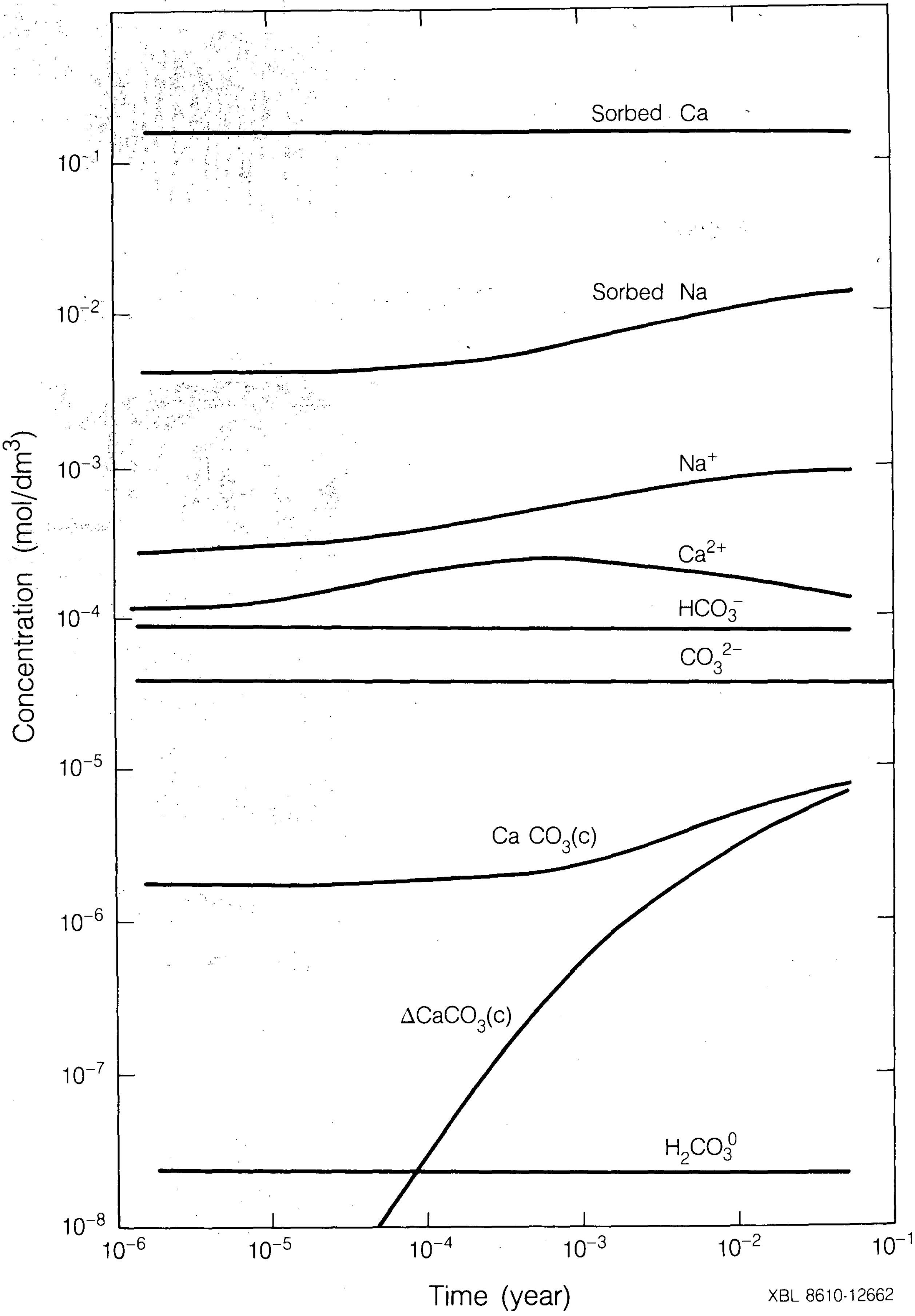
XBL 8610-12652

Fig. 5. Ca^{2+} variations vs. time in batch kinetics, equilibrium and kinetic transport.



XBL 8610-12661

Fig. 6. HCO_3^- and H_2CO_3 breakthrough curves in kinetic transport (K-T) and equilibrium transport (EQ-T).



XBL 8610-12662

Fig. 7. Variation of concentration of aqueous species, sorbed ions and solids vs. time.

precipitation respectively. The chemical reaction between silica and water is expressed as

$$SiO_2(s) + 2\ H_2O(\ell) = H_4SiO_4(aq)$$

The rate constants are given by

$$\log k_- = -707 - 2598/T$$

and

$$\log k_+ = A + BT + C/T$$

where T is absolute temperature and

$A = 1.174$, $B = -2.028 \times 10^3$, $C = -4158$ for quartz ;

$A = -0.739$, $B = 0$, $C = -3586$ for α-cristobalite ;

$A = -0.936$, $B = 0$, $C = -3392$ for β-cristobalite ;

$A = -0.936$, $B = -7.890 \times 10^{-4}$, $C = -3438$ for amorphous silica.

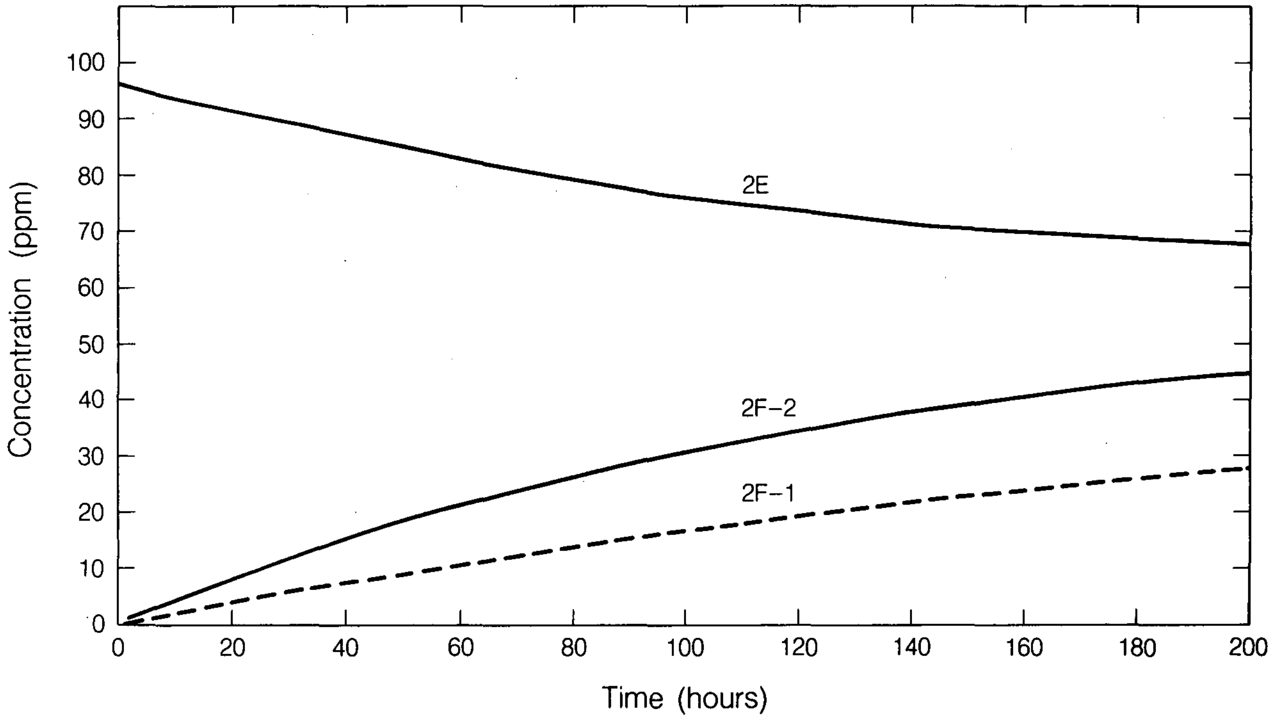
With the above parameters given, the rate equation is completely defined. A subsection of subroutine SRATE implements the needed algorithm and calculates the right hand side of Eq. (35).

Precipitation of silica - In this problem we simulated the precipitation experiment of Rimstidt and Barnes (1980). The chemistry in this problem is kept quite simple by considering H_4SiO_4 as the only aqueous species and SiO_2 as the only solid that can form. The constants of the rate equation in this problem are those of the curve 2E in the experiment. Simulation starts with a supersaturated solution of $10^{-3}\underline{M}$ of silica at 105°C. To achieve this initial condition the equilibrium solubility of the SiO_2 (Quartz) is set equal to the assumed initial concentration. The input and output to this simulation are given in Appendix 6. The results of this run are shown by the curve labeled 2E in Fig. 8. This curve practically duplicates the results of the Rimstidt and Barnes (1980) experiment.

Dissolution of silica - In this problem we simulated the dissolution part of the Rimstidt and Barnes (1980) experiment. The chemistry in this case is identical to the precipitation case. However, here we start with an undersaturated solution of $10^{-20}\underline{M}$ of silica. In this problem we made two different runs using different sets of rate constants. In the first run, we used the k_+ and k_- values of the curve 2F of the experiment which resulted in the dashed curve 2F-1 in Fig. 8. Obviously, this result varied largely from those of the experiment. However, when we used the data of the precipitation experiment (curve 2E), we obtained a perfect match for the dissolution experiment, as shown by curve 2F-2 in Fig. 8. We cannot explain the reason for failure of simulation when correct data is used and its unexpected success otherwise. However, we do not attribute this behavior to lack of soundness of the algorithm. The input and output for this problem are provided in Appendix 6.

Simulation of Non-Reversible Kinetic Dissolution of Glass

Formulation - Claassen and White (1979) suggest an empirical kinetic formula for the dissolution mechanism of glass in tuffs. This



XBL 8610-12648

Fig. 8. Variation of silica concentration vs. time.

expresses the time-dependent release of various cations into an aqueous solution containing dissolved CO_2. The formula has the form:

$$Q_{M,t} = Q_o + k_M t^{1/2} \quad , \tag{48}$$

where $Q_{M,t}$ is the total mass of cation M transferred from solid to solution in time t, Q_o is the mass transferred from solid to solution by ion exchange at the fresh rock surface, and k_M is the rate constant associated with the transfer of cation M at a specific pH. The rate of dissolution, R_M, is therefore given by

$$R_M = \sigma \frac{dQ_{M,t}}{dt} = 0.5 \, \sigma \, k_M t^{-1/2} \quad ,$$

where σ is aquifer surface area in contact with a unit volume of solution; σ has units of cm^2/liter . The above derivation neglects the secondary effects of pH variations on k_M.

The reaction in this dissolution mechanism is assumed to occur in the following steps. The recharging water dissolves soil-zone CO_2 to provide the aqueous carbonate constituents of the infiltrating water. Details of CO_2 dissolution are provided in Appendix 1. The distribution of the carbonate species in this water is governed by the equilibrium between HCO_3^- and $H_2CO_3^0$ at any pH. The $H_2CO_3^0$ of the groundwater reacts with the silicate rock leading to the release of various cations. These steps can be written as

$$[CO_3^{2-}] + [HCO_3^-] + [H_2CO_3^0] = \text{constant} \; ;$$

$$HCO_3^- + H^+ = H_2CO_3 \quad , K_1 \; ;$$

$$(\text{silicate rock})(M^{n+}) + H_2CO_3^0 \rightarrow (\text{silicate rock})(H^+) + M^{n+} + H_2CO_3^0 \quad , \quad k_{M^{n+}} \ .$$

The rate constants were calculated from data given by Claassen and White (1979), and are

$$\log k_{Na^+} = -0.17(pH - 5.75) + 13.73 \ ,$$

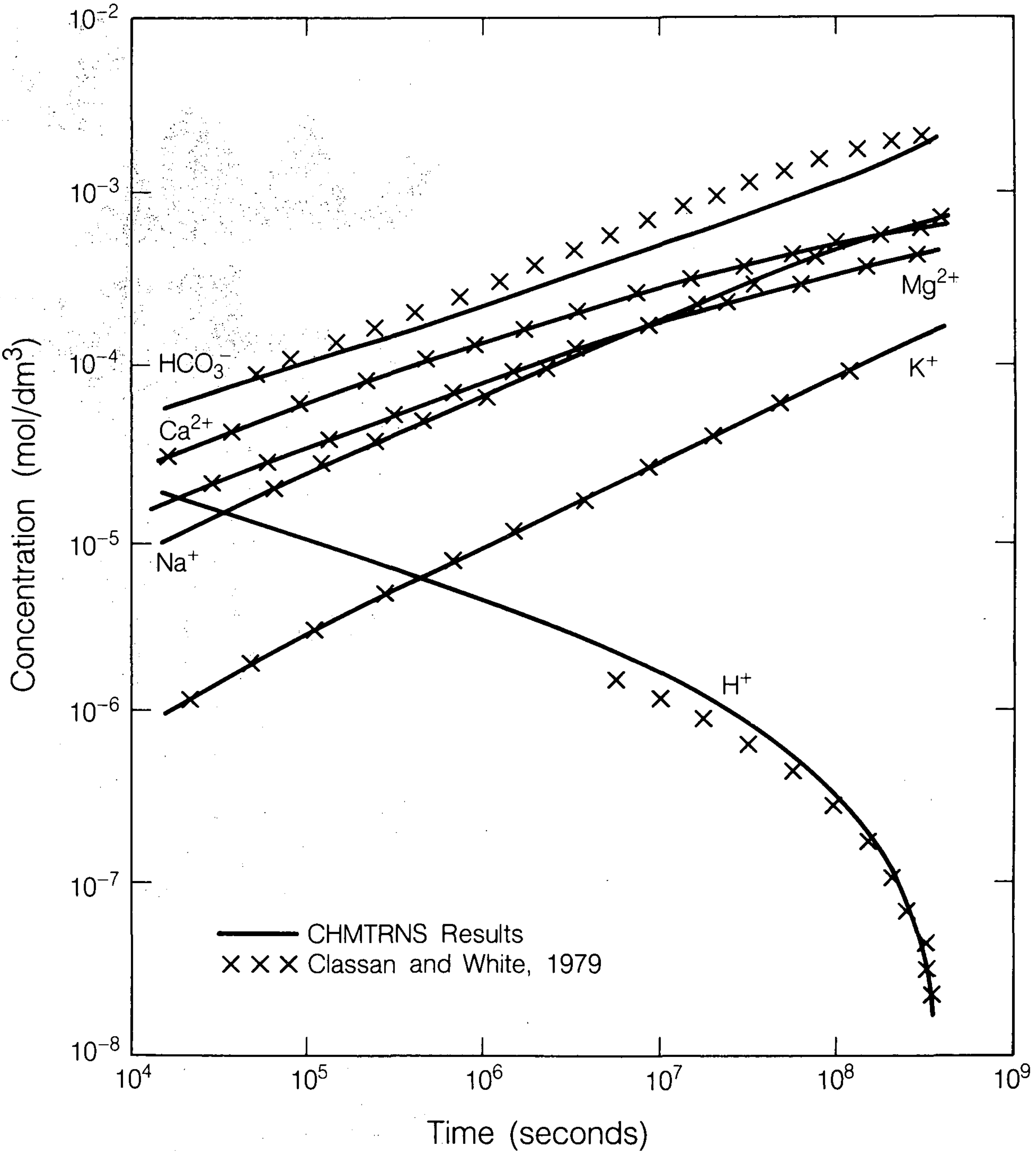
$$\log k_{K^+} = -14.54 \ ,$$

$$\log k_{Ca^{2+}} = -0.5(pH - 6.25) + 14.15 \ ,$$

$$\log k_{Mg^{2+}} = -0.4(pH - 5.25) + 13.88 \ ,$$

The algorithm for these empirical rate formulas has been implemented in the subroutine SRATE.

Dissolution of silicate rocks - In this section we repeat the calculations of Claassen and White (1979) using CHMTRNS. In this problem we assumed $\sigma = 1 \times 10^6$ cm^2/liter and the total carbonate concentration of 2.16×10^{-3} mol/liter at pH = 4.5. The total simulated time was 3.52×10^8 seconds. The results of our calculation along with those of Claassen and White (1979) are shown in Fig. 9. With the exception of the pH and HCO_3^- results, which are systematically overestimated, the calculations agree almost exactly. The mentioned discrepancies are due to linearization of the rate constants in our calculations. In another set of calculations, we included the contribution of the time variation of the rate coefficients in the rate formula. As a result the rate formula became



XBL 8610-12656

Fig. 9. Dissolution of various cations and HCO_3^- and H^+ in time.

$$R_M = 0.5\ \sigma\ k_M t^{-1/2} + \left[\frac{dk_M}{dpH} \cdot \frac{dpH}{dt}\right] \sigma t^{1/2} \qquad (49)$$

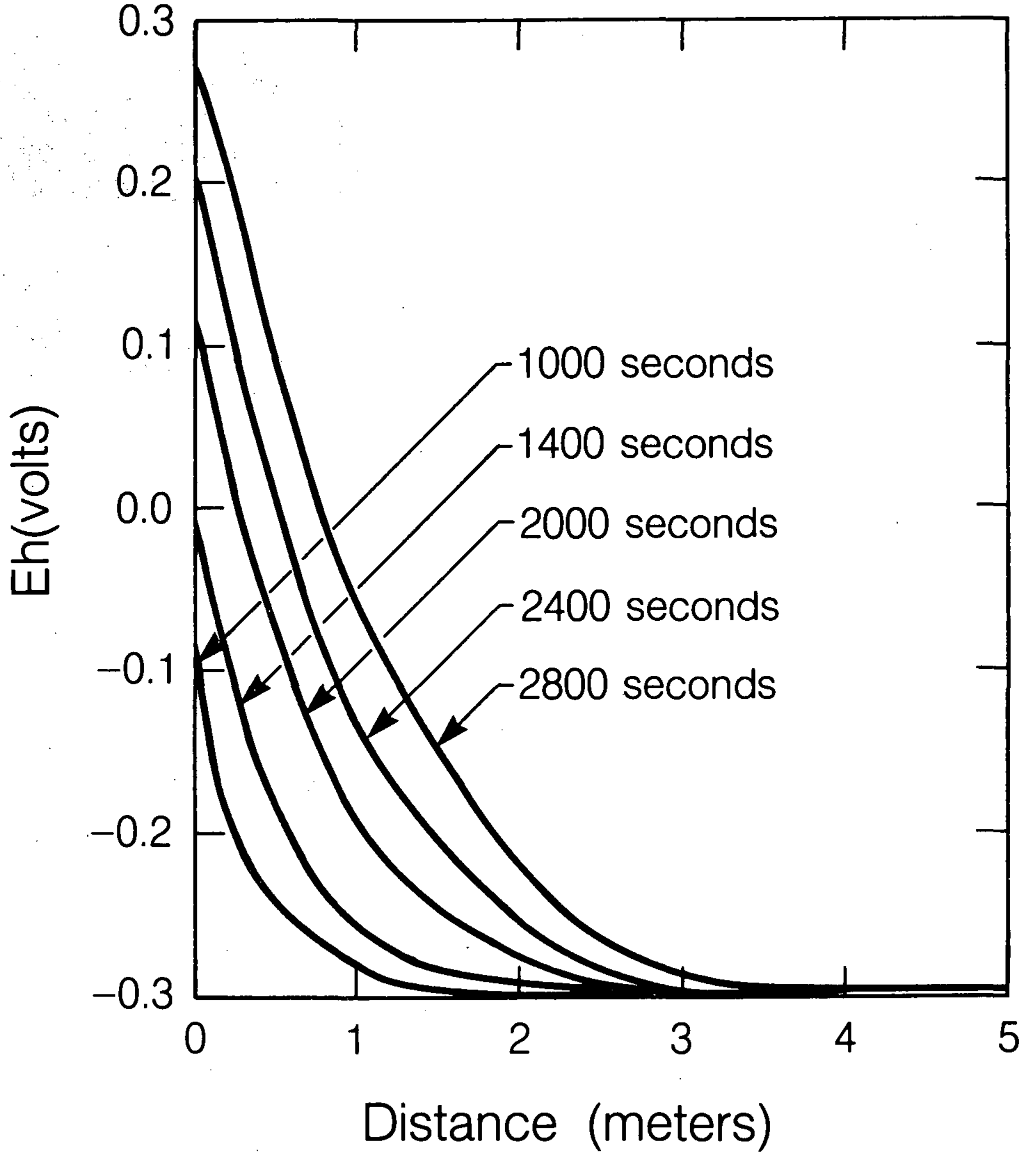
However, the apparent improvement of the rate formula caused deterioration of the results. Considering the simple nature of the algorithms, such deviations are unexpected. Further examination seems to be necessary. The input and output for these simulations are included in Appendix 7.

Simulation of Non-Isothermal Oxidation-Reduction During Transport

In this simulation an effort was made to test both the oxidation-reduction and the heat transport algorithms. The problem was that of 150°C water with Eh = 0.3 v and pH = 4 entering a long tube initially containing 50°C water with Eh = -0.3 v and pH = 10*. Convective-dispersive heat transport along with reactive solute transport are considered. The chemical reactions considered were redox, complexation, precipitation and dissolution. The tube length was taken to be 10 meters discretized into 40 sections. Fluid velocity was 10^{-3} m/second. The results of this simulation for 2800 seconds are shown in Figs. 10-14. The figures depict the evolution of Eh, aqueous species $H_2CO_3^0$ and UO_2^{+2}, precipitates $UO_2(c)$ and $UO_2(OH)_2(c)$ and the temperature along the flow path. Since the above problem had originally been simulated with the code THCC(Carnahan, 1976), we could compare the results of both simulations. In spite of the perfect agreement of the results in their evolutionary trend, there are discrepancies in the magnitudes of various species. However, these differences are attributed to variations in the numerical approaches. The general performance agreement of the two

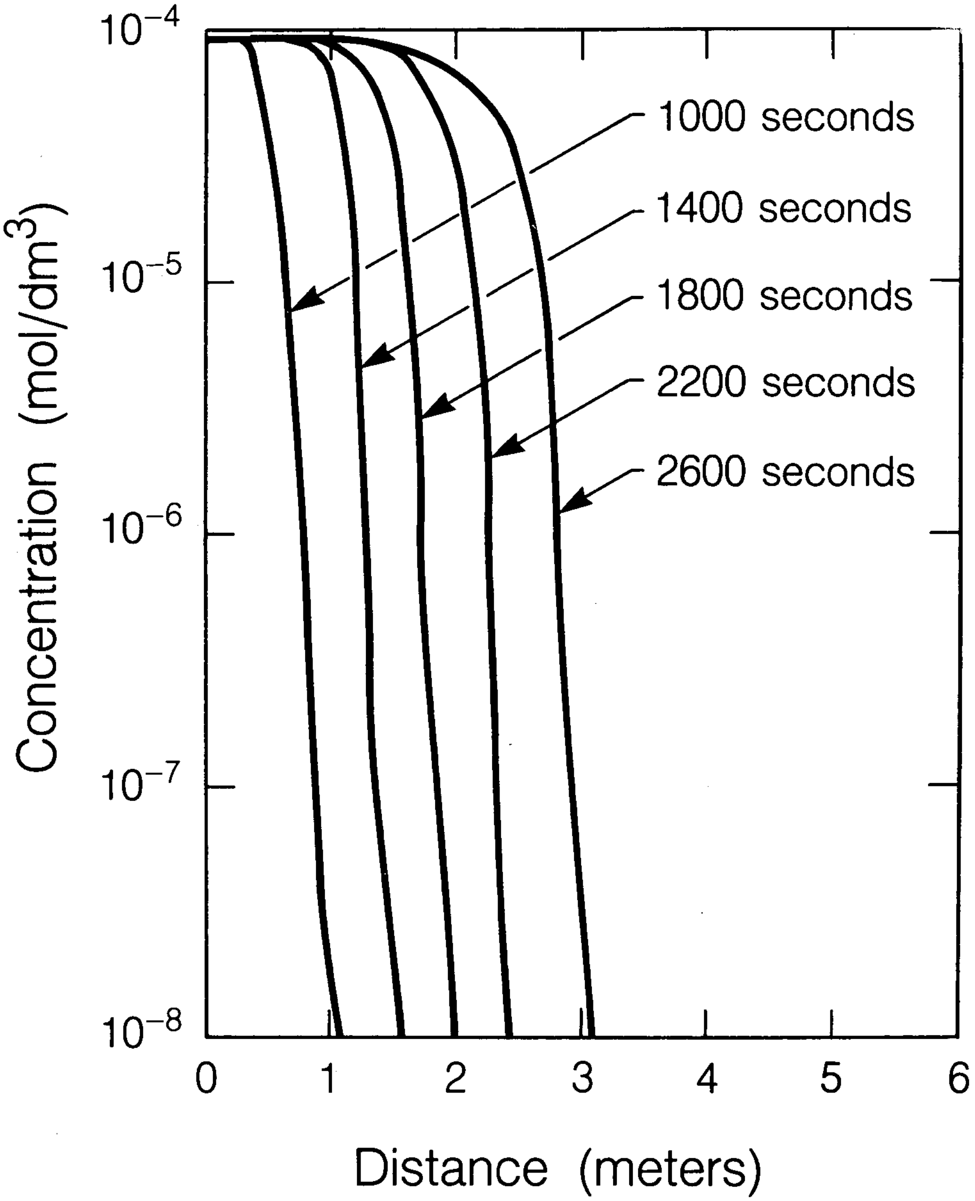
---

*See Appendix 8 for input details



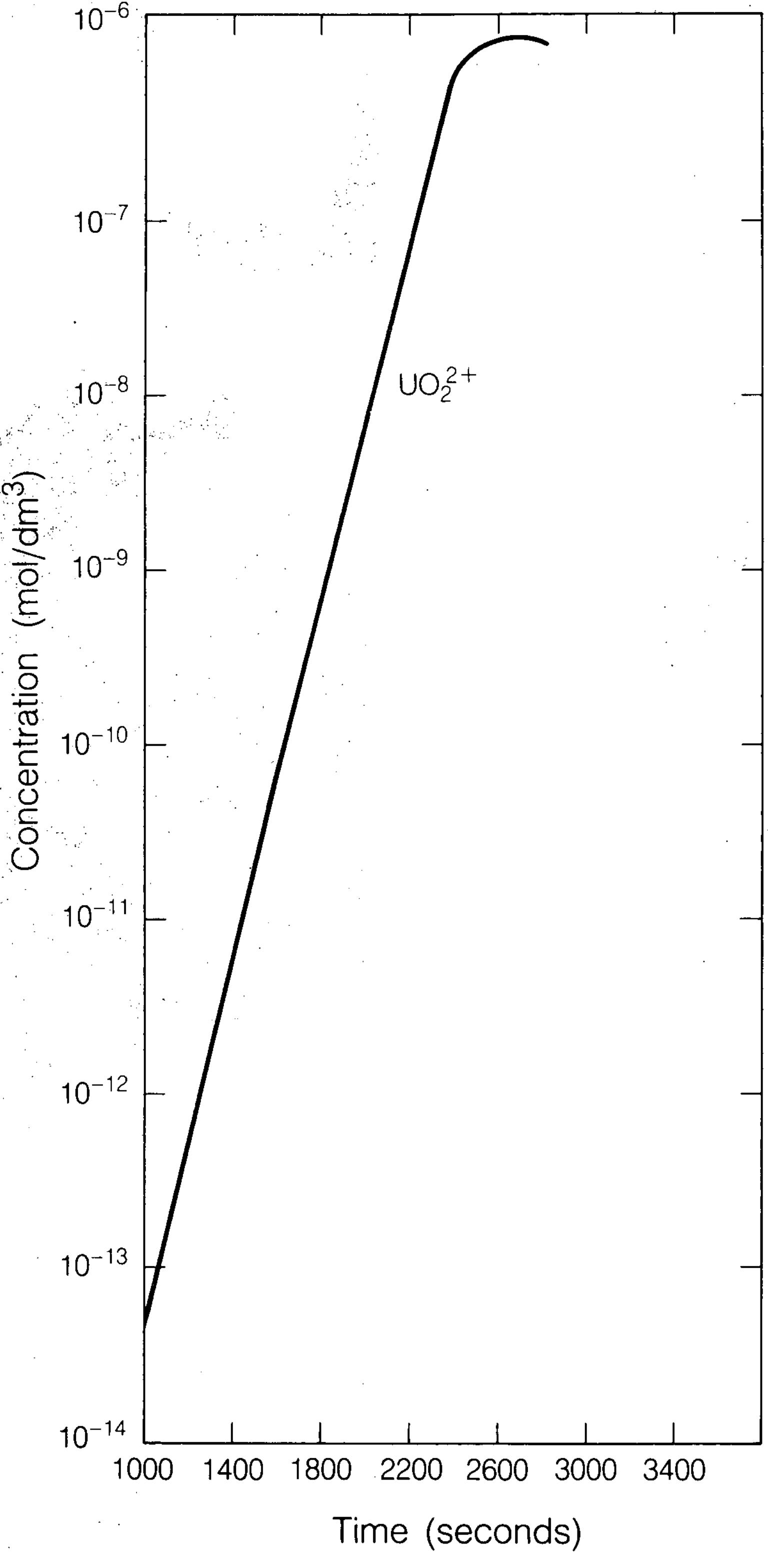
XBL 8610-12642

Fig. 10. Eh profiles along the flow path at various times.



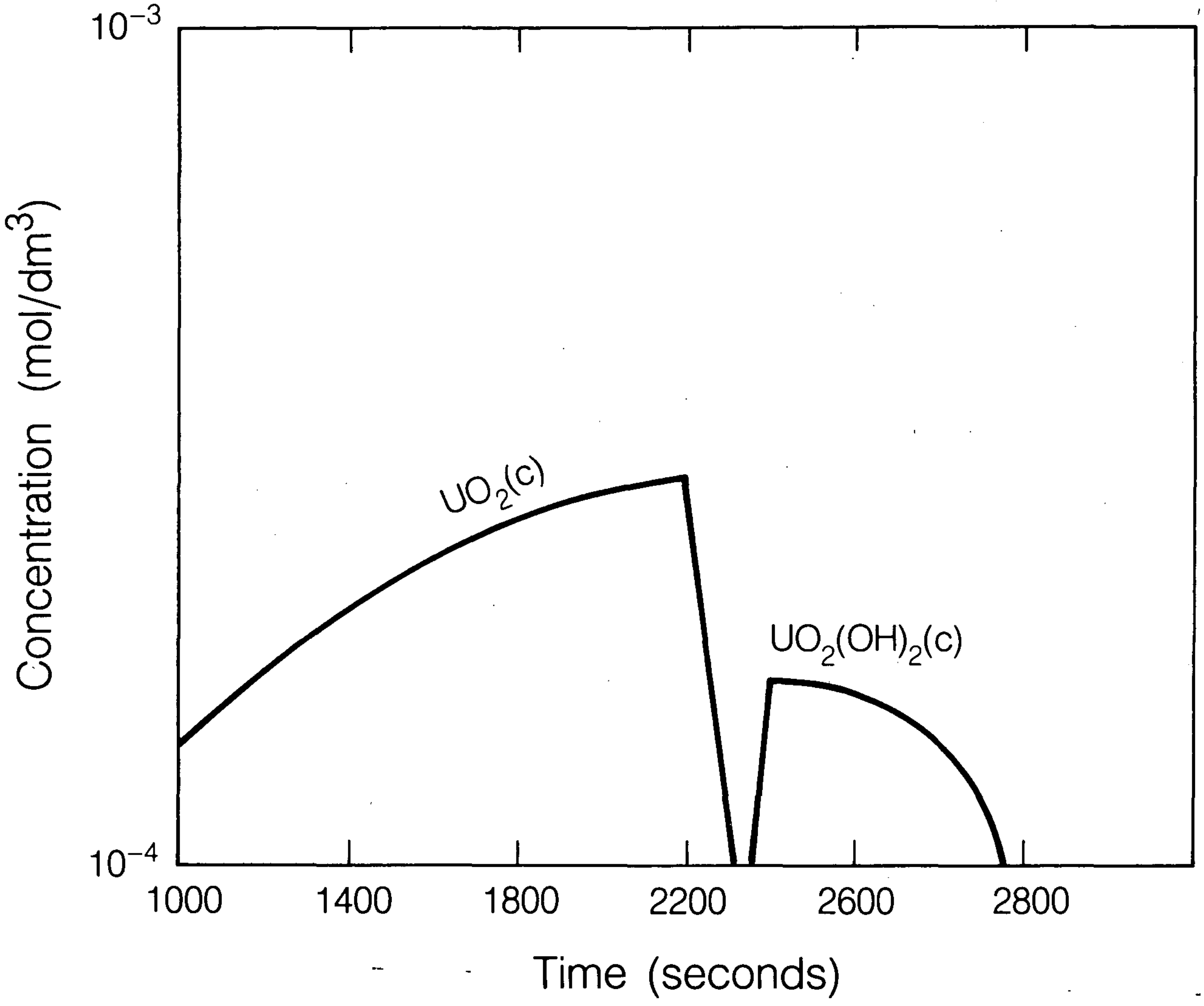
XBL 8610-12654

Fig. 11. Carbonic acid profile along the flow path at various times.



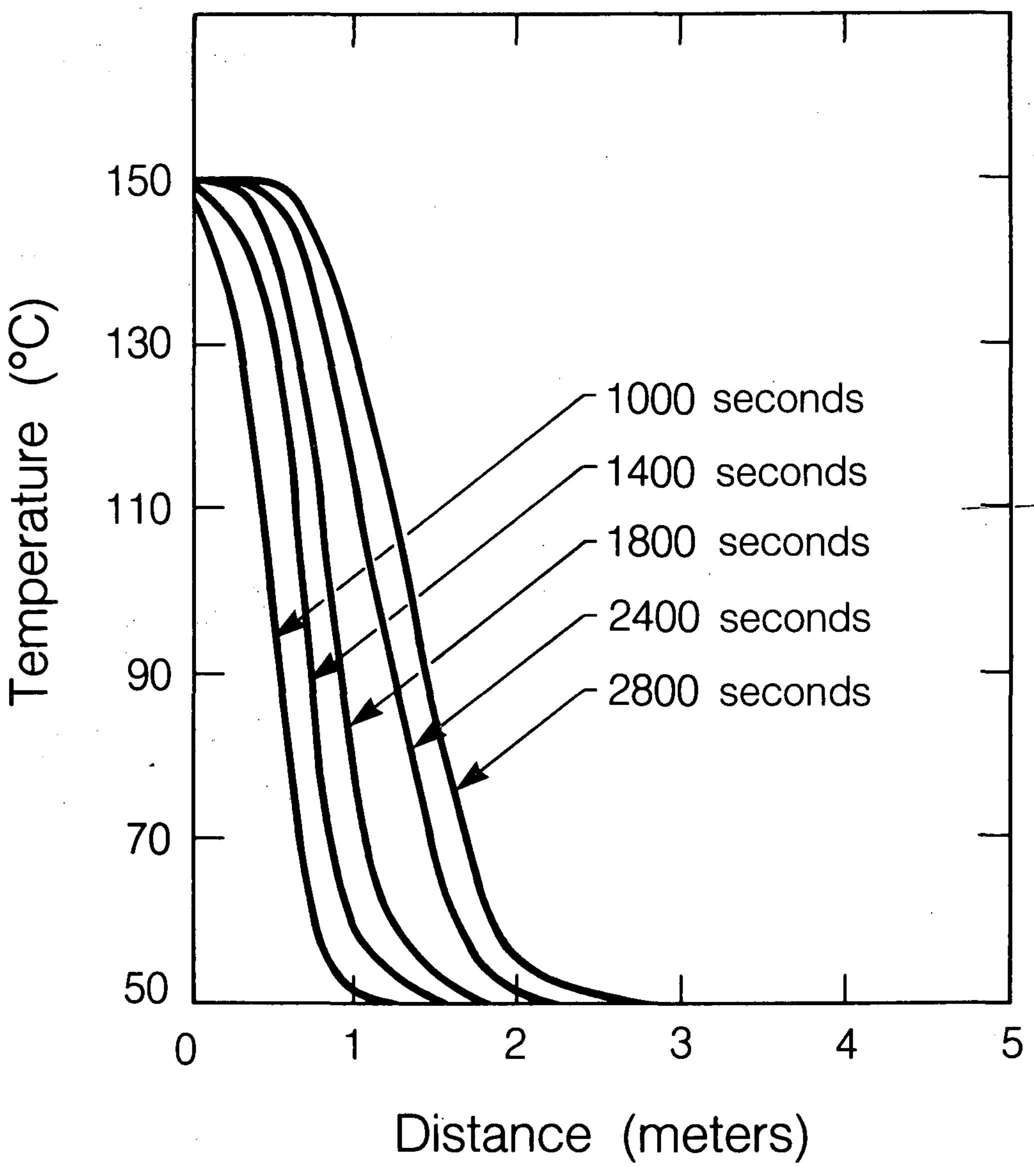
XBL 8610-12653

Fig. 12. Build up of UO_2^{2+} as a function of time near the inflow boundary.



XBL 8610-12644

Fig. 13. Precipitation and dissolution of $UO_2(C)$ and $UO_2(OH)_2C$ in time.



XBL 8610-12643

Fig. 14. Temperature profile along flow path at various times.

codes provides a strong basis for algorithm soundness. The input and output to this problem are provided in Appendix 8.

Simulation of ^{13}C Fractionation During Non-Equilibrium Reactive Solute Transport in Geologic Systems

Major issues regarding the evolution of groundwater are the composition of recharge waters, travel times to and from and within aquifers, and chemical processes that have occurred during the transport. Traditionally, studies of groundwater ages have been based on the assumption of dilution of carbon from various sources. In this approach the ^{13}C content of the water sample is measured by (Pearson and Hanshaw, 1970)

$$^{13}\delta_{sol_{sm}} = \frac{\sum {}^{13}\delta_{solj}\, m_{cj}}{\sum m_{cj}} \tag{50}$$

and the activity of ^{14}C is similarly

$$A_{sm}(^{14}C) = \frac{\sum m_{cj} A_j^0 e^{-\lambda t_j}}{\sum m_{cj}} \tag{51}$$

where m_{cj} denotes the concentration of stable carbon from source j,t represents time and A^0 is taken as equal to close to 100% modern ^{14}C. Considering only two sources of carbon, one belonging to the recharge water and the other to that of a rock enriched in ^{13}C, Eq. (50) becomes

$$^{13}\delta_{sol_{sm}} = \frac{^{13}\delta_a\, m_{Ca} + {}^{13}\delta_s\, m_{C_s}}{m_{C_a} + m_{C_s}}$$

where the subscripts a and s refer to the recharge water (air) and the rock. After simple manipulations of the above relationship we get

$$\frac{m_{C_a}}{m_{C_a} + m_{C_s}} = \frac{{}^{13}\delta_{sol_{sm}} - {}^{13}\delta_s}{{}^{13}\delta_a - {}^{13}\delta_s} \tag{52}$$

Equation (51), therefore, changes to

$$A_{sm}({}^{14}C) = \frac{m_{C_a} A_o e^{-\lambda t}}{m_{C_a} + m_{C_s}} \tag{53}$$

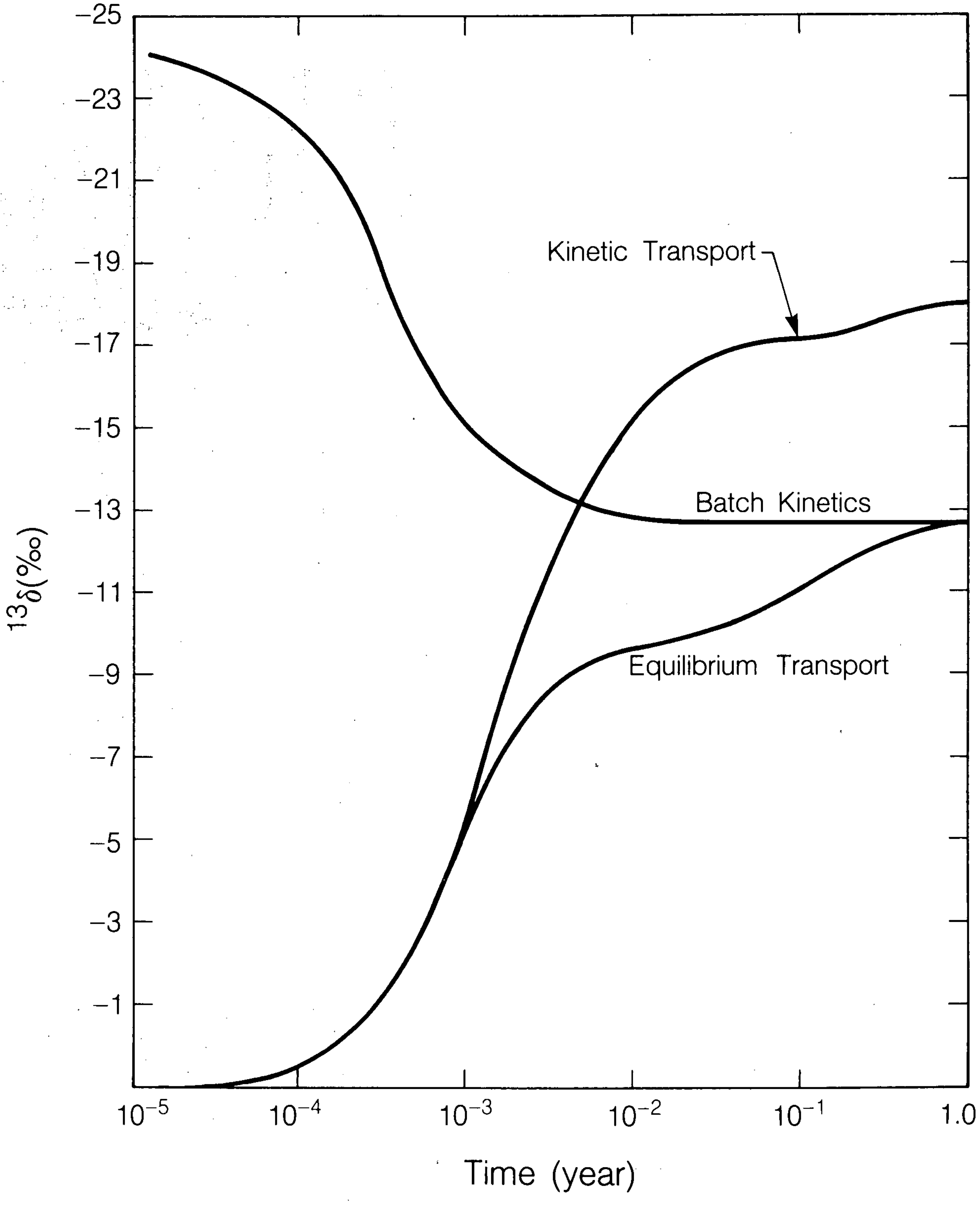
Assuming reasonable values for the parameters in Eq. (52) makes it possible to obtain a correct age for the waters. As mentioned earlier, derivations such as the above (e.g., Wigley, 1975), are based on simple batch mixing assumptions. However, the evolution of groundwater may have involved complex non-equilibrium chemical processes during transport through the earth material. CHMTRNS can serve as a basis for testing the validity of the simple dilution assumption for specific problems. Also, it can be employed for general investigation of ground water evolution processes. In the following we provide accounts of such attempts.

Generic simulations - In this problem we investigated the ${}^{13}C$ fractionation of recharge waters as they percolate through heavy carbonate rocks. It is reported (Deines et al., 1974) that for most soil waters $\delta^{13}C \approx -25$o/oo and P_{CO_2} ranges from $10^{-0.5}$ atm. to $10^{-3.5}$ atm. and pH ranges from 4.0 to 7.5 and ${}^{14}C = 100\%$ modern. Based on the above data, a typical water of this nature entering marine carbonate rocks that have $\delta^{13}C = 0$ o/oo and ${}^{14}C = 0\%$ will tend to have $\delta^{13}C \approx -12.5$o/oo and ${}^{14}C = 50\%$ modern, after equilibrium has been reached in the dissolution reaction

$$CO_2 + H_2O + CaCO_3 \rightarrow Ca^{2+} + 2HCO_3^- \; . \tag{54}$$

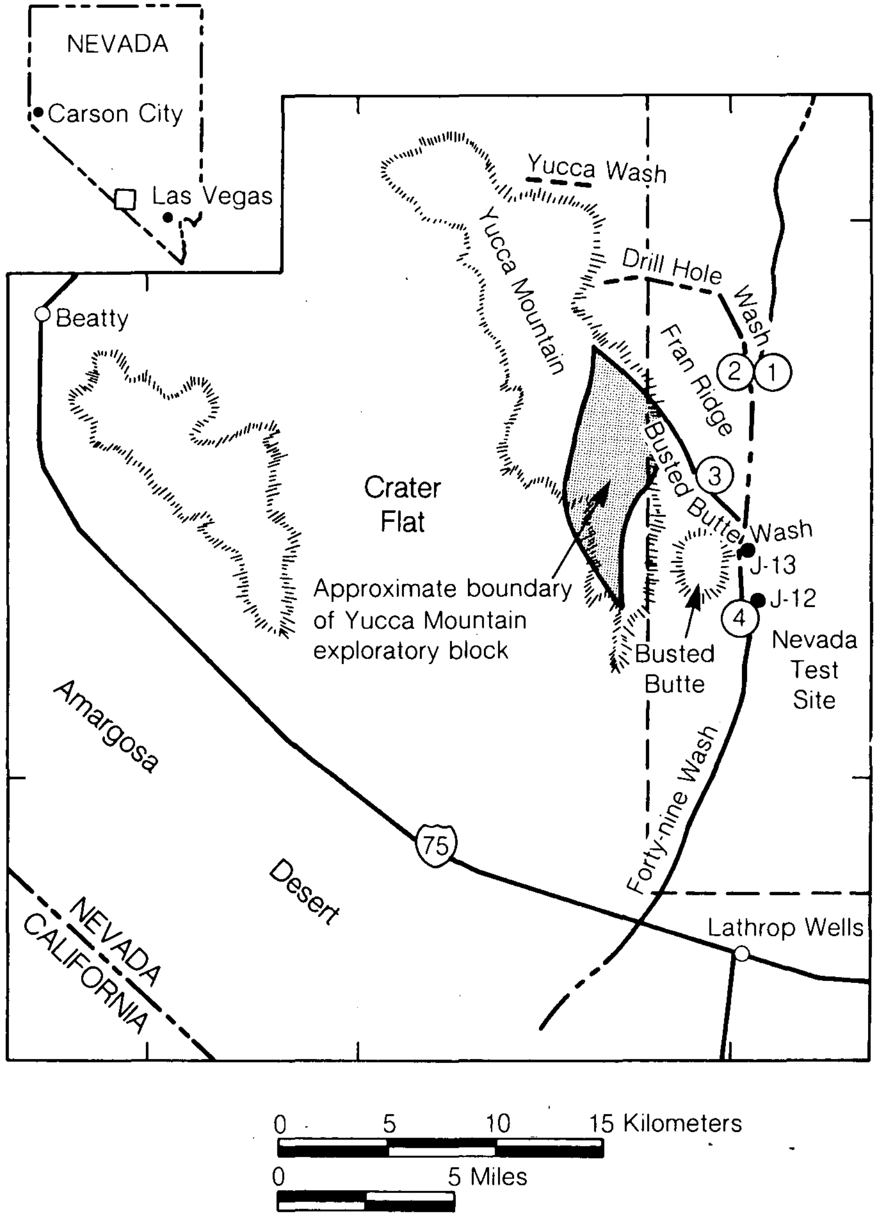
The reason is evident from the mass balance of Eq. (54). However, if this reaction does not go to completion, because of kinetic effects, reactive transport effects, or paucity of solid carbon in the flow path, the dilution by the dead rock would lead to $\delta^{13}C$ between −25o/oo to −12.5o/oo. Some of the above conditions are simulated in three hypothetical cases we have solved. Before discussing these results, attention is drawn to the fact that for all cases initial equilibrium calculations were based on CO_2 dissolution in water with pH = 5 in order to obtain the recharge water composition. These calculations are provided in Appendix 9. For initial conditions in the carbonate rock we assumed the composition of pure water at equilibrium with the rock. The flow path is a one-dimensional discretized space. The three runs simulate batch kinetic calcite dissolution, equilibrium dissolution of calcite with transport, and kinetic dissolution with transport. The results are exhibited in Fig. 15. While the first two simulations dilute the ^{13}C in the water by 50% and reduce $\delta^{13}C$ to 13o/oo, the kinetic transport is effective only by about 30%, even after a year. The input and output to these problems, which were also used in an other context, are given in Appendix 2.

Scoping calculations for Yucca Mountain (Nevada) groundwater - In this example the effect of kinetic dissolution during transport at a particular site is considered. In this first-order study of the Yucca Mountain groundwater evolution process, we used the typical composition of surface waters of the Busted Butte site [see Fig. (16) for site map], shown in Table 1, for the recharge water (Benson, 1986). In this



XBL 8610-12649

Fig. 15. Time variation of $^{13}\delta_{sol}$ as a result of dilution of the carbonate species by marine calcium carbonate in various scenarios.



① Forty mile Wash above Drill Hole Wash on Rd. 'H'.

② Drill Hole Wash at the mouth.

③ Busted Butte Wash

④ Forty mile Wash at Well J-12

XBL 8610-12663

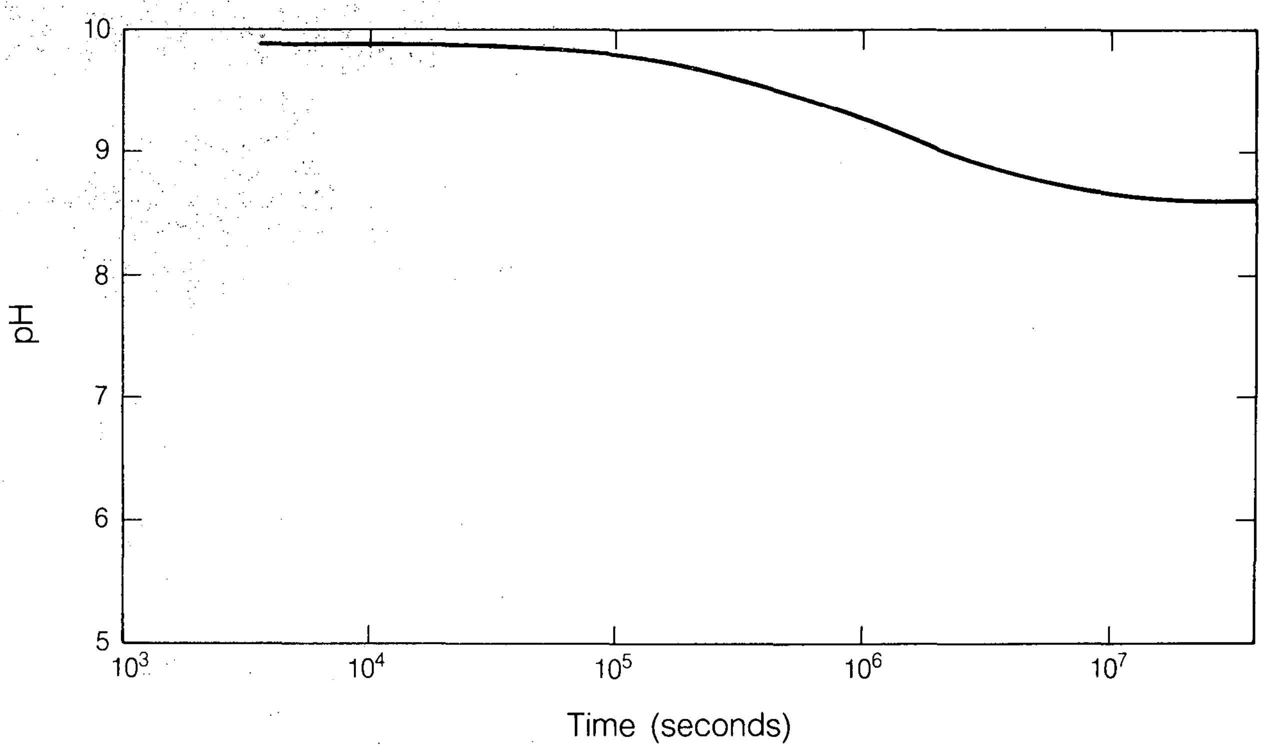
Fig. 16. Location of sampling sites near Yacca mountain.

Table 1. Chemical analysis from Forty-Mile Wash surface water samples.
1 to 4 Forty-Mile Wash above Drill Hole Wash
5 and 6 Drill Hole Wash
7 and 8 Busted Butte
9 Forty-Mile Wash near Well J-12

	STATION	DATE	TIME	Na mg/c	Ca mg/c	Mg	K	HCO_3 field	HCO_3 lab	SO_4	Cl	SiO_2	Li μg/L	Sr μg/L	F	Dissolved Solids	pH field	pH lab	SC field	SC lab	δD	δO^{18}	C^{14} yrs BP	C 13/12 ratio
	1	7/22/84	1925	9.0	27	3.6	8.9	110	83	11	4.5	26		120	0.2	140	7.6	7.2	198	212	---	---	---	---
	2	7/24/84	2005	8.1	24	3.3	7.8	106	125.6	10	3.7	25	6	100	0.2	145	7.3	7.4	182	201	−89	−12.3		
40 mile	3	8/14/84	1505	4.1	8.1	0.90	5.6	---	43.9	6.2	1.3	8.7	6	34	---	57	8.4	7.5	70	198	−60	−9.0	---	---
	4	3/15/84	2230	8.2	21	2.9	9.1	---	91.5	10 0.69	1.4 0.316	24	7	100	0.2 0.038	122	8.0	7.4	170	259	−96	−13.7	0	−13.8
	5	8/14/84	1510	8.6	9.5	1.3	7.4	---	51.2	12	2.2	20	14	66	0.3	92	8.3	7.8	100	218	−57.3	−8.9	---	−12.7
Drill Hole	6	7/19/85	1900	11	19	2.6	10	---	103.7	10	3.2	26	14	120	0.3	134	---	7.9	---	193	---	---	---	---
	7	8/14/84	1530	7.0	12	1.8	8.1	---	57.5	7.9	1.7	23	17	86	0.3	100	8.3	7.8	120	217	−69.5	−10.1	+846	−16.1
Busted Butte	8	8/15/84	0230	6.9	9.8	1.5	5.8	---	52.4	4.9 0.69	1.2 0.?16	16	14	67	0.2 0.038	73	8.3	8.0	90	184	−70.5	−10.5	---	−17.4
40 m J-12	9	8/14/84	1600	2.4	6.7	0.7	6.3	---	31.7	6.3	2.0	4.5	5	31	---	45	8.2	7.0	59	100	−64.5	−9.5	---	−16.1

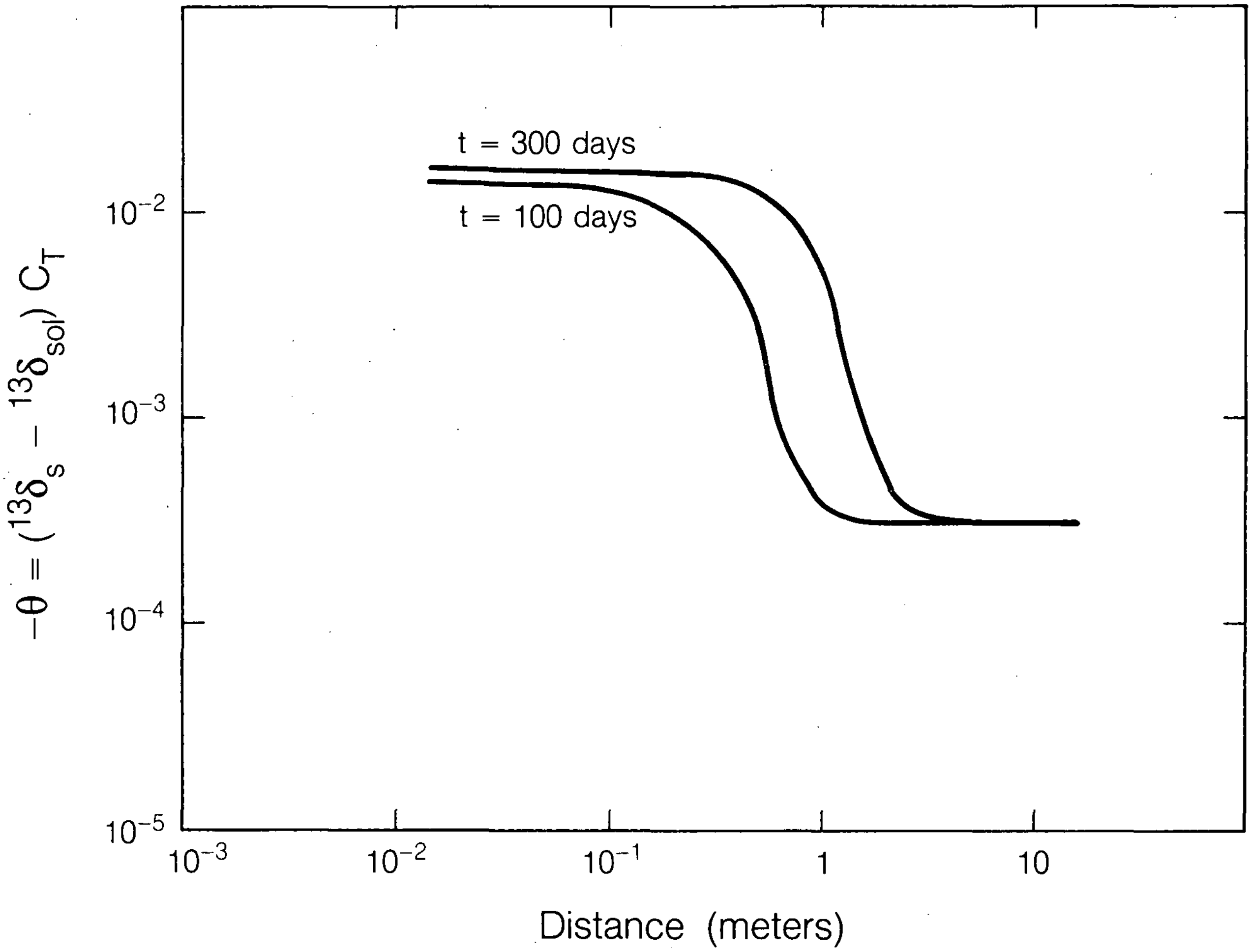
Reference: Benson, L.V., unpublished U.S.G.S. Report.

simulation a constant flux of surface water was input at the boundary of a one-dimensional column of tuff saturated with pure water. We allowed reversible kinetic dissolution of calcite and silica and non-reversible kinetic dissolution of glass. This study, having a scoping nature, precluded any other complex reactions which might be of relevance, and also used additional simplifications. In order to reduce simulation time, the percolation velocity was increased from the actual order of cm/year to m/year, and also the flow path was reduced to a 10-meter distance discretized into 20 segments. As has been pointed out before, the latter assumption posed no drawback because we considered the events which were not affected by exit boundary effects during the observation time. That is, we only analyzed near-entrance events that stabilized before the far boundary sensed any change. The results of this simulation are not very dramatic because of the very low dissolution capability of the water. Therefore, expectedly, a front with the composition of the flux water swept through the column. No appreciable change of $\delta^{13}C$ was observed due to the fact that little dilution took place. The results of this simulation are shown in Figs. 17–23. Figure 17 exhibits the small pH variations. Figure 18 shows the advance of the θ_d front and consequently, as seen in Fig. 19, a very sharp front of $\delta^{13}C$ of about –16 o/oo develops at around 100 days into the simulation. At the end of the one-year simulation $\delta^{13}C$ in the upstream part of the flow path approaches the value for the recharge water, as shown in Fig. 20. Figure 21 demonstrates the variations of the concentrations of carbonate species at a point near the entrance as a function of time. Dissolution of glass is very low because of the high pH of both the in situ water and the percolating water. It is therefore overshadowed by higher



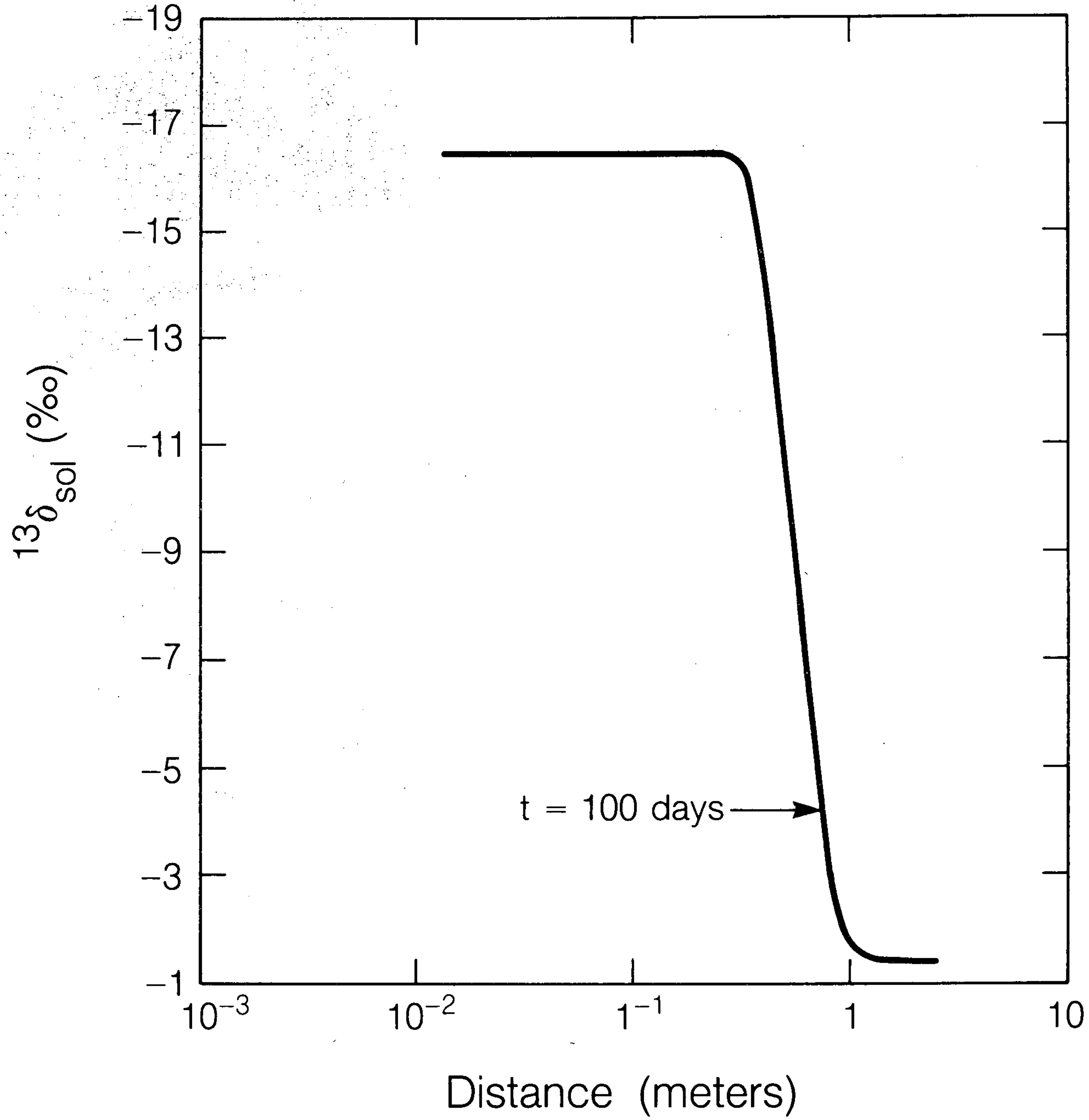
XBL 8610-12645

Fig. 17. pH variations vs. time.



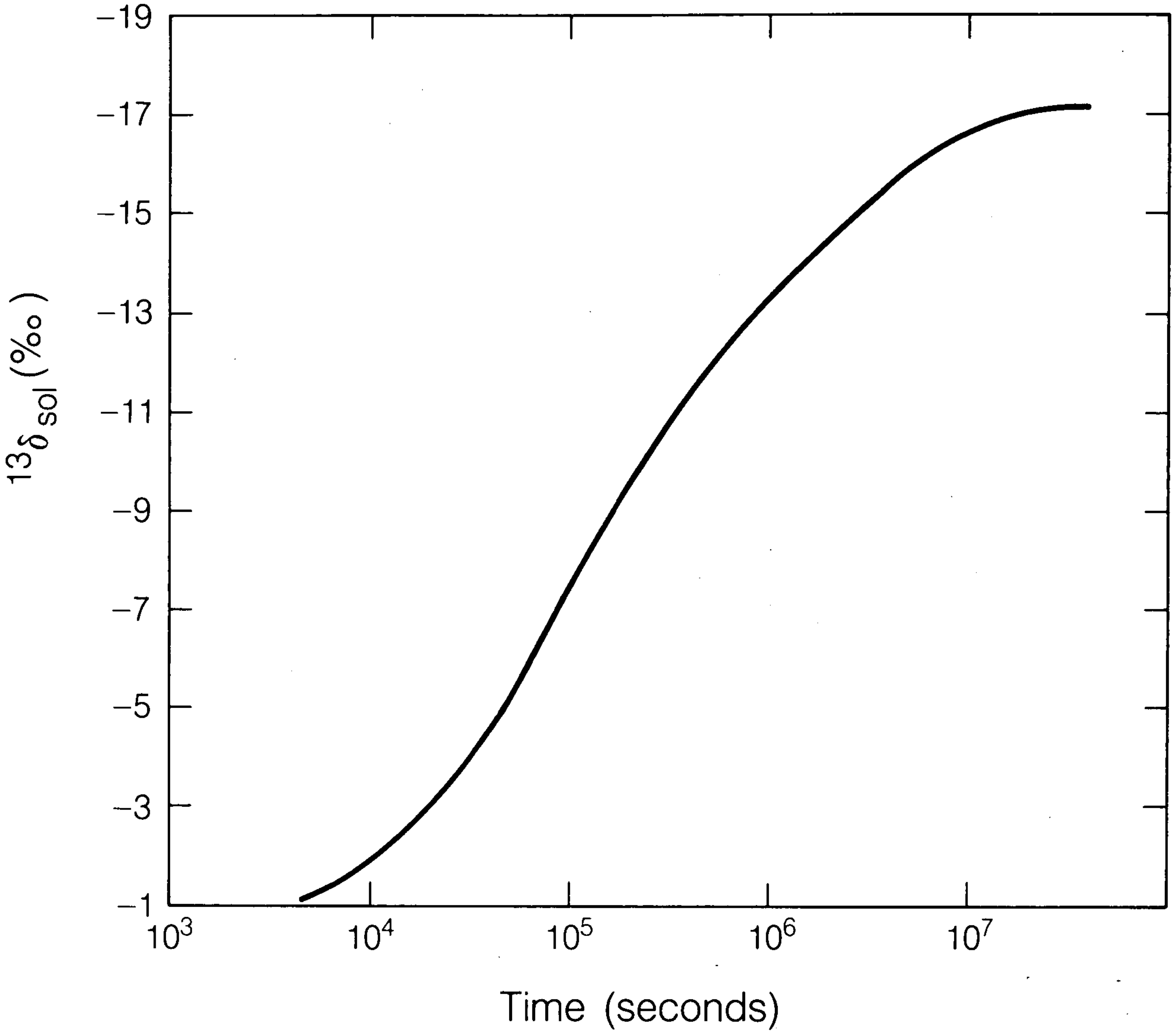
XBL 8610-12641

Fig. 18. Profile of θ_d along the flow path at two different times, $^{13}\delta_s$, $^{13}\delta_{sol}$ and c_T describe $^{13}\delta$ of rocks, $^{13}\delta$ of the infiltrating water and the total carbonate concentration of the solute.



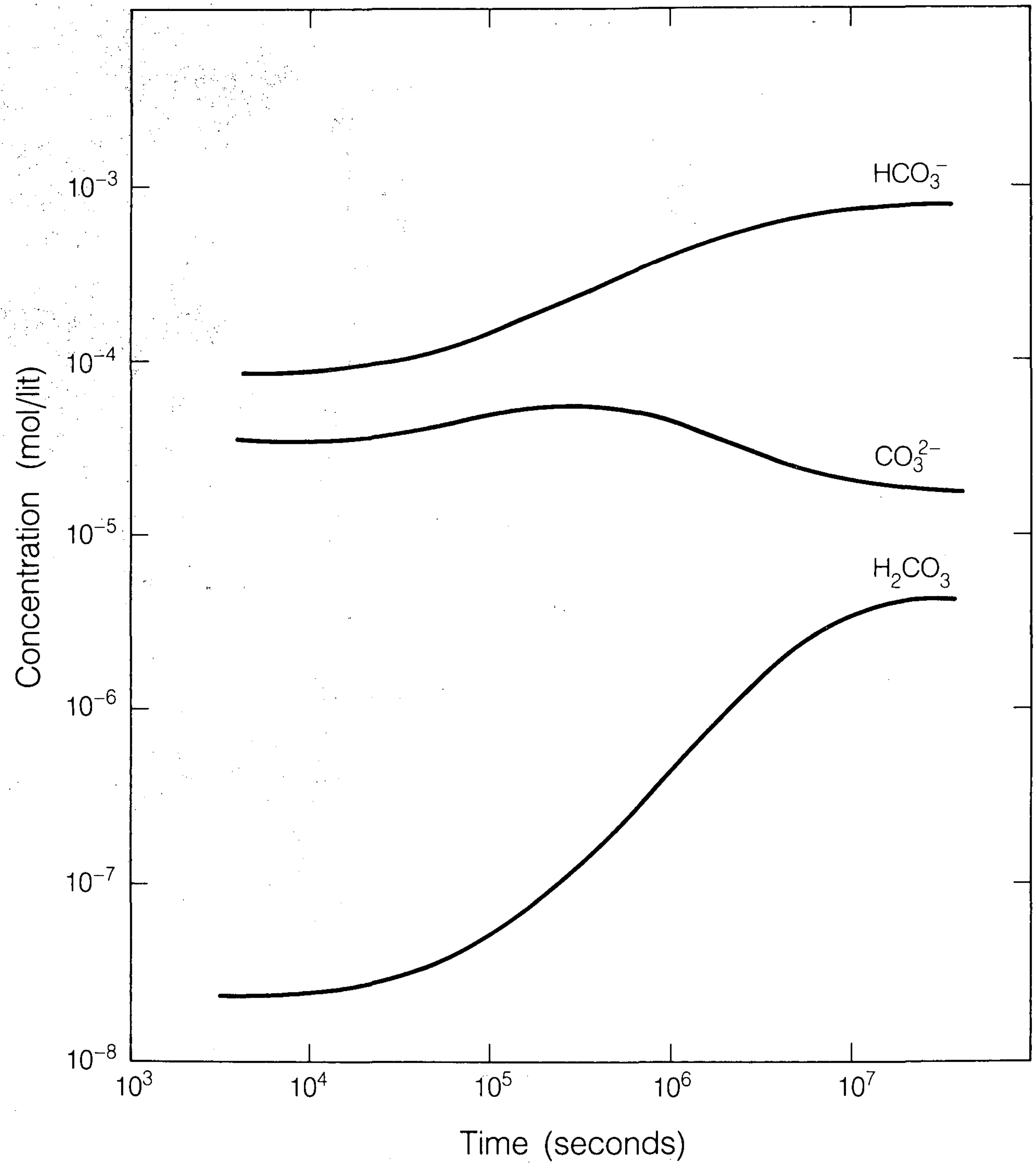
XBL 8610-12646

Fig. 19. Profile of $^{13}\delta_{sol}$ along the flow path for 100 days.



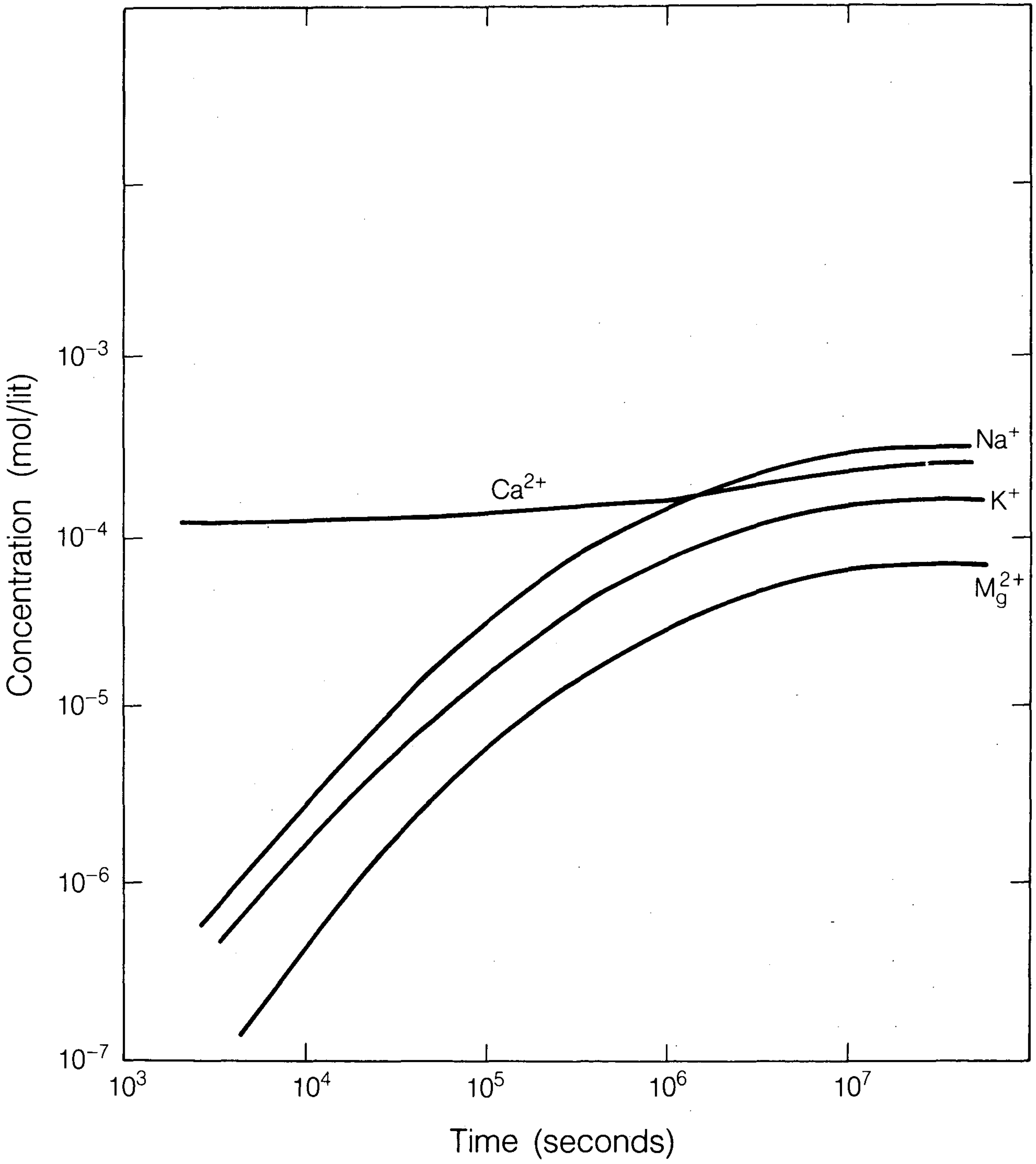
XBL 8610-12647

Fig. 20. Breakthrough of the solution shift in the percolating water.



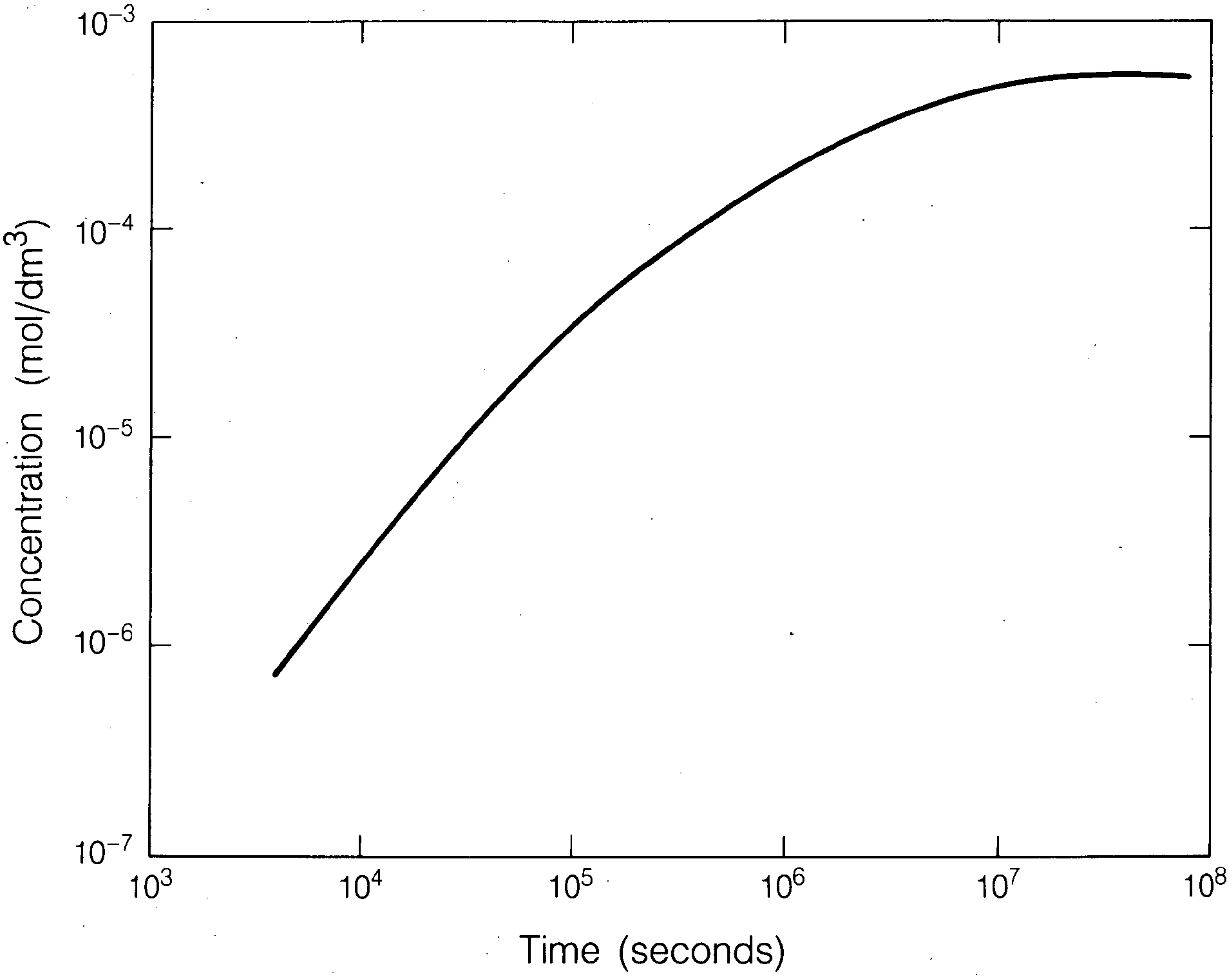
XBL 8610-12655

Fig. 21. Breakthrough of the aqueous carbonate species near the inflow boundary.



XBL 8610-12657

Fig. 22. Breakthrough of Mg^{2+}, Na^{+}, Ca^{2+} and K^{+} concentrations near the inflow boundary.



XBL 8610-12651

Fig. 23. Breakthrough of dissolved SIO_2 near the inflow boundary.

concentrations of the recharge water cations, as can be noticed in Fig. 22. Variation of the dissolved silica concentration in time is depicted in Fig. 23. Since the maximum solubility of amorphous silica in the *in situ* water at 30°C is about 2.10×10^{-3} *M*, the concentration of the amorphous silica in the water can only reach this level. However, because of convection, the breakthrough curve for a point near the inflow point shows the concentration stabilizing at around twice the incoming concentration (under saturated), as depicted in Fig. 23. Details of this application are provided in Appendix 10.

CONCLUSION

In this work, the geochemical equilibrium reactive transport code CHEMTRN was thoroughly revised, modified and supplemented. Specifically, the work included the following.

1. Thorough revision and optimization of the solution algorithms.
2. Options for reversible kinetic calcite and silica dissolution and irreversible glass dissolution added.
3. Option for oxidation-reduction added.
4. Option for carbon stable isotope fractionation during transport added.
5. Option for non-isothermal transport added.

Each stage of the development was verified against other methods if possible, otherwise assessed by examination of the code's performance in a multitude of hypothetical problems. Accounts of the applications are given in graphic presentations and also in the tabulations of the Appendices.

This new version of the code is called CHMTRNS.

ACKNOWLEDGEMENTS

This work was carried out under U.S. Department of Energy Contract No. DE-AC03-76SF00098 via the DOE-Nevada Office in cooperation with USGS-Denver. The authors are grateful to Dr.C.F. Tsang for his guidance and continuous support throughout the project.

REFERENCES

Benson, L.V., 1986. Unpublished., U.S.G.S. report.

Carnahan, C.L., 1987a. Simulation of chemically reactive solute transport under conditions of changing temperature In C.-F. Tsang, ed., Coupled Processes Associated with Nuclear Waste Repositories, Academic Press, Inc., Orlando, Florida, p. 249–257.

Carnahan, C.L., 1987b. Simulation of uranium transport with variable temperature and oxidation potential: The computer program THCC. In J.K. Bates and W.B. Seefeldt, eds., Scientific Basis for Nuclear Waste Management X, Materials Research Society Symposia Proceedings Vol. 84, Materials Research Society, Pittsburgh, Pennsylvania, p. 713–721.

Cederberg, G.A., Street, R.L., and Leckie, J.O., 1985. A groundwater mass transport and equilibrium chemistry model for multicomponent systems. Water Resour. Res., v. 21, p. 1095–1104.

Claassen, H.C., and White, A.F., 1979. Application of geochemical kinetic data to groundwater systems. A tuffaceous-rock system in southern Nevada. In E.A. Jenne, ed., Chemical Modeling in Aqueous Systems, ACS Symposium Series, No. 93, American Chemical Society, Washington, D.C., p. 771–793.

Clarke, E.C.W., and Glew, D.N., 1966. Evaluation of thermodynamic functions from equilibrium constants. Trans. Faraday Soc., v. 62, p. 539–547.

Crank, J., 1975, The Mathematics of Diffusion, Clarendon Press, Oxford.

Davis, J.A., James, R.O. and Leckie, J.O. 1978, Surface Ionization and the Oxide/Water Interface. I. Computation of Electrical Double Layer Properties in Simple Electrolytes, J. Colloid and Interface

Science, V. 63, p. 480–499.

Deines, P., Langmuir, D., and Harmon, R.S., 1974, Stable Carbon Isotope Ratios and Existence of a Gas Phase in the Evolution of Carbonate Ground Waters, Geochimica et Cosmochimica Acta, V. 38, p. 1147–1164.

James, R.V. and Rubin, Jacob, 1979. Applicability of Local Equilibrium Assumption to Transport Through Soils Affected by Ion Exchange: in Jenne, E.A., ed. Chemical Modeling in Aqueous Systems: American Chemical Society, Washington, D.C., ACS Symposium Series 93, p. 225–235.

Kirkner, D.J., Theis, T.L., and Jenning, A.A., 1984. Multicomponent solute transport with sorption and soluble complexation. Adv. Water Resour., v. 7, p. 120–125.

Kirkner, D.J., Jennings, A.A., and Theis, T.L., 1985. Multisolute mass transport with chemical interaction kinetics. J. Hydrology, v. 76, p. 107–117.

Miller, C.W., and Benson, L.V., 1983, Simulation of Solute Transport in a Chemically Reactive Heterogeneous System: Model Development and Application: Water Resonances Research, V. 19, p. 381–391.

Palciauskas, V.V., and Domenico, P.A., 1976, Solution Chemistry, Mass Transfer, and the Approach to Chemical Equilibrium in Porous Carbonate Rocks and Sediments: Geological Society of America Bulletin, V. 87, p. 207–214.

Pearson, J.F., and Hanshaw, B.B., 1970, Sources of Dissolved Carbonate Species in Groundwater and Their Effects on Carbon-14 Dating, Proceedings, Isotope Hydrology 1970, p. 271–286, International Atomic Energy, Vienna.

Plummer, L.N., Wigley, T.M.L. and Parkhurst, D.L., 1978. The Kinetics of Calcite Dissolution in CO_2 - Water System at 5° to 60°C and 0.0 to 1.0 ATM CO_2; American J. of Science, V. 278, p. 179–216.

Rimstidt, J.D., and Barnes, H.L., 1980. The kinetics of silica-water reactions. Geochim. Cosmochim. Acta, v. 44, p. 1683–1699.

Stumm, W., Morgan, J.J., 1970, Aquatic Chemistry, Wiley-Interscience, New York.

Valette-Silver, J.N., Thompson, J.M., and Ball, J.W., 1981: Relationship between water chemistry and sediment mineralogy in the Cerro Prieto Geothermal Field: A Prelimianry Report, in Proceedings of Third SYmposim on the Cerro Prieto Geothermal Field, Baja California, Mexico, March 24–26, 1981, San Francisco, California, U.S.A. Lawrence Berkeley Laboratory, LBL-11967, p. 263.

Wigley, T.M.L., 1975, Carbon 14 Dating from Closed and Open Systems, Water Resources Research, V. 11, No. 2, p. 324–328.

APPENDICES

INPUTS AND OUTPUTS OF SOLVED PROBLEMS

APPENDIX 1

SOIL WATER COMPOSITION

INPUT AND OUTPUT

INPUT

```
Static equilibrium CO2(gas) in water--Sinpart
UNITS      meter    sec.
type               0         0         0         0
iteration        100       100       100
Calc.crtia   1.0e-30   1.0e-03        20        10         5      1.75         5
grid              05      10.0       0.0
time               1       1.0       5.0
print time       1.0
physical        0.05       0.0       0.0
tran bound         0         0
temp bound         0         0
isothermal         0
Intl.           10.0
leach              0         0
WORD      blank
no.species         3         0         0         3         0         2         0
CO2(g)    0.0    0    23.1623e-023.1623e-02    23.1623e-023.1623e-02
H+        1.0    0    2   1.0e-05   1.0e-05    2   1.0e-05   1.0e-05
OH-       -80.3723-1.395e+03   32.5514-3.181e-02
            3.24e-10  3.24e-10
H2CO3       0  1.0  0.0  0.0
              87.598  -1055.95   -38.7233.4506e-02
                 0.0       0.0
HCO3-       0  1.0  0.0  1.0
             127.844   -532.58   -52.2593.1543e-02
                 0.0       0.0
CO3--       0 +1.0  0.0  2.0
             281.934  -3619.68  -112.5486.2045e-02
                 0.0       0.0
```

OUTPUT

```
1Static equilibrium CO2(gas) in water--Sinpart

    unit of length is   meter            unit of time is    sec.

    inner boundary is constant concentration          outer boundary is constant initial concentration


    dispersion coefficient= 0.1v+  0.           fluid velocity = 0.            number of grid points =    5


    number of print outs =    1
    given at the following times=  1.000

errsqr=   0.10e-29     climit=   0.10e-02     iterj=    20     iterm=    10     iterd=     5     dtmult=   0.17e+01     ndiv=     5

    number of sites(ion-exchange)    0.0000           number of sites(edl model)    0.0000
 this simulation is isothermal, temperature =  10.



     2       bare ions             0     sorbing ions (ionex)             0     sorbing ions (edl model)


component charge       kinetic     initial condition        boundary condition
                          code       type         value        type         value

CO2(g)      0.         0          2       3.1623e-02        2       3.1623e-02
H+          1.         0          2       1.0000e-05        2       1.0000e-05


     4      complexes             0       sorbing complexes(ionex)             0     sorbing complexes (edl model)


complex    charge       kcode                            stoichiometry

OH-         -1.         0         0.   0.   1.
H2CO3        0.         0         1.   0.   0.
HCO3-       -1.         0         1.   0.   1.
CO3--       -2.         0         1.   0.   2.
                                                               ** spatial grid **

    0.            2.500        5.000        7.500        10.00
```

OUTPUT

```
 1                                                          initial conditions
                                                              temp=   10.00


 component        total          bare ion        sorbed concentration        act. coef.     kd(ion ex)        kd(edl)
               aqueous conc.   concentration      (ionex)     (edl)

 CO2(g)         0.3162e-01     0.3009e-01     0.          0.                   1.000        0.                0.
 H+             0.1000e-04     0.1000e-04     0.          0.                  0.9915        0.                0.
 OH-            0.5637e-04     0.3296e-09     0.          0.                  0.9915        0.                0.

 complex          valence        aqueous conc.   sorption(io)      sorption(edl)
 H2CO3          0.             0.1473e-02     0.              0.
 HCO3-          -1.000         0.5637e-04     0.              0.
 CO3--          -2.000         0.1865e-09     0.              0.

    to have a net balance of zero charge, other 'species must be present (although not participating in the
    chemical reactions) with a total concentration times charge of  0.464e-04
```

```
 1                                 boundary or influx conditions
                                          temp=   10.00
 component        total          bare ion        sorbed concentration        act. coef.     kd(ion ex)        kd(edl)
               aqueous conc.   concentration      (ionex)     (edl)

 CO2(g)         0.3162e-01     0.3009e-01     0.          0.                   1.000        0.                0.
 H+             0.1000e-04     0.1000e-04     0.          0.                  0.9915        0.                0.
 OH-            0.5637e-04     0.3296e-09     0.          0.                  0.9915        0.                0.

 complex          valence        aqueous conc.   sorption(io)      sorption(edl)
 H2CO3          0.             0.1473e-02     0.              0.
 HCO3-          -1.000         0.5637e-04     0.              0.
 CO3--          -2.000         0.1865e-09     0.              0.

    to have a net balance of zero charge, other 'species must be present (although not participating in the
    chemical reactions) with a total concentration times charge of  0.464e-04
```

APPENDIX 2

KINETIC DISSOLUTION OF CALCITE DURING REACTIVE TRANSPORT

INPUT AND OUTPUT

Table of Chemical Reactions

1. Equilibrium chemical reaction

complexation

$H^+ + CO_3^{2-} = HCO_3^-$ log K = 10.30

$2H^+ + CO_3^{2-} = H_2CO_3$ log K = 16.7

dissociation of water

$H_2O = H^+ + OH^-$ $\log K_w = -13.99$

2. Kinetic chemical reactions (Plummer et al, 1978)

$$CaCO_3(c) + H^+ \rightleftarrows Ca^{2+} + HCO_3^-$$

$$CaCO_3(c) + H_3CO_3^0 \rightleftarrows Ca^{2+} + 2HCO_3^-$$

$$CaCO_3(c) + H_2O \rightleftarrows Ca^{2+} + HCO_3^- + OH^-$$

Net rate of dissolution:

$$R = k_1 a_{H}^{+} + k_2\ a_{H_2CO_3} + k_3\ a_{H_2O} - k_4\ a_{Ca}^{2+}\ a_{HCO_3^-}$$

INPUT

```
Partitioned Isothl.Kin. Calcite Diss.Tran.Int.shitsol=-24.5,shifts=1.0-sdypartd
UNITS      mo       year
 type              0         1         0         1
iteration        100       100       100
Calc.crtia   1.0e-30   1.0e-03        20        10         5      1.75         5
grid              20      10.0  836.6835
time              14 1.585e-10   1.0e-03
print time 1.142e-05 1.142e-04 1.141e-03 4.566e-03 1.141e-02 2.740e-02 5.479e-02
print time 1.370e-01 2.740e-01 4.110e-01 5.480e-01 6.849e-01 8.219e-01 9.589e-01
Physical         0.1   1.0e-10   0.1e+01
tran bound         1         0
temp bound         0
isothermal         0
Intl. temp      10.0
leach              0         0
WORD      tpkin
**Carbonate Diss. Rate Data next 7 lines
1 PARAMS           5     100.0         4     prepc
CDATS           19.8     6.383    10.377     8.446 3.154e+04
2 INXI(i)     2    0   09    0    0    0    5    8
Carbonate1     -1.32         0         0         0
         2     -4.59         0         0         0
         3     -6.94         0         0         0
         4       0.0         0         0         0
WORD      blank
In.Sh.Info     -24.5       1.0
Iso.Ratio1 -0.91e-03       6.3
         2 -4.54e-03    1099.0
         3 -3.40e-03     870.0
         4 -3.63e-03    1194.0
no.species         7         0         0         2         1         2         0
Na+       1.0    0    2   1.0e-03   1.0e-03    2   1.0e-03   1.0e-03
H+        1.0    0    2 8.653e-05 1.237e-10    1.  1.0e-05   1.0e-05
Co3--    -2.0   31    31.0122e-02 3.521e-05    3 1.529e-03 1.865e-10
neg.KI    0.0   32    1 1.000e-10 1.000e-10    1 3.899e-02 3.899e-02
Ca++      2.0    0    31.0122e-02 1.217e-04    3       0.0       0.0
Cl-      -1.0    0    4   1.0e-03   1.0e-03    0   1.0e-03   1.0e-03
OH-        -13.990       0.0       0.0       0.0
          8.653e-05 3.296e-10
HCO3-      33  0.0  1.0  1.0  0.0  0.0  0.0  0.0
              10.337       0.0       0.0       0.0
          8.649e-05 5.637e-05
H2CO3      34  0.0  2.0  1.0  0.0  0.0  0.0  0.0
              16.700       0.0       0.0       0.0
          2.359e-08 1.473e-03
CaCO3      10  0.0  0.0  1.0  0.0  1.0  0.0  0.0
              -8.446       0.0       0.0       0.0
                 .01       0.0
```

OUTPUT

```
1Partitioned Isothl.Kin. Calcite Diss.Tran.Int.shitsol=-24.5,shifts=1.0-sdypartd

    unit of length is   meter           unit of time is    year

kinetics of rev. precipitate dissolution considered


    inner boundary is constant flux          outer' boundry is constant initial concentration


    dispersion coefficient= 0.1v+  0.10e-09     fluid velocity =  1.00         number of grid points =   20


    number of print outs =   14
    given at the following times= 0.1142e-040.1142e-030.1141e-020.4566e-020.1141e-010.2740e-010.5479e-010.1370
                                  0.2740    0.4110    0.5480    0.6849    0.8219    0.9589

errsqr=   0.10e-29     climit=   0.10e-02     iterj=    20     iterm=    10     iterd=     5     dtmult=   0.17e+01     ndiv=     5

    number of sites(ion-exchange)    0.0000          number of sites(edl model)    0.0000
 this simulation is isothermal, temperature =  10.



     6      bare ions              0     sorbing ions (ionex)          0     sorbing ions (edl model)


component charge      kinetic     initial condition        boundary condition
                         code       type        value          type        value

Na+          1.         0          2       1.0000e-03       2       1.0000e-03
H+           1.         0          2       8.6530e-05       1       1.0000e-05
Co3--       -2.        31          3       1.0122e-02       3       1.5290e-03
neg.KI       0.        32          1       1.0000e-10       1       3.8990e-02
Ca++         2.         0          3       1.0122e-02       3       0.
Cl-         -1.         0          4       1.0000e-03       0       1.0000e-03


     3      complexes              0       sorbing complexes(ionex)            0     sorbing complexes (edl model)


complex    charge        kcode                        stoichiometry

OH-        -1.          0         0.    0.    0.    0.    0.    0.    1.
HCO3-      -1.         33         0.    1.    1.    0.    0.    0.    0.
H2CO3       0.         34         0.    2.    1.    0.    0.    0.    0.
```

OUTPUT

```
     1       solid precipitates
  precipitate  kinetic code         stoichiometry
 CaCO3            10           0.0  0.0  1.0  0.0  1.0  0.0  0.0
 Calcite dissolution information:

 no of terms in the rate formulla =     4
 no of other additional data needed =     5
 representative surface parameter =    0.100e+03
 xdiag = prepc

 additional data for rate calculation
   0.198e+02   0.638e+01   0.104e+02   0.845e+01   0.315e+05
binary element nos. of the terms of the rate equation
               2    0    9    0    0    0    5    8

coefficients in log function of thermodynamic constants
              -1.320     0.000     0.000     0.000
              -4.590     0.000     0.000     0.000
              -6.940     0.000     0.000     0.000
               0.000     0.000     0.000     0.000

          C-13 Fractionation parameters:
          shiftr=   -24.5000          shifts=     1.0000

coefficients in log function of thermodynamic constants
              -0.001     6.300
              -0.005  1099.000
              -0.003   870.000
              -0.004  1194.000

                                                  ** spatial grid **

   0.           0.5086e-02   0.1233e-01   0.2266e-01   0.3738e-01   0.5835e-01   0.8823e-01   0.1308       0.1915       0.2780
   0.4012       0.5768       0.8271        1.184        1.692        2.416        3.448        4.918        7.014        10.00
```

OUTPUT

```
1                                                       initial conditions
                                                          temp=   10.00

component       total          bare ion        sorbed concentration        act. coef.    kd(ion ex)       kd(edl)
             aqueous conc.   concentration      (ionex)     (edl)

Na+           0.1000e-02     0.1000e-02     0.         0.                  0.9596        0.              0.
H+            0.8653e-04     0.1188e-09     0.         0.                  0.9596        0.              0.
Co3--         0.1260e-03     0.3953e-04     0.         0.                  0.8478        0.              0.
neg.KI        0.1000e-09     0.1000e-09     0.         0.                   1.000        0.              0.
Ca++          0.1260e-03     0.1260e-03     0.         0.                  0.8478        0.              0.
Cl-           0.9930e-03     0.9930e-03     0.         0.                  0.9596        0.              0.
OH-           0.9357e-04     0.9357e-04     0.         0.                  0.9596        0.              0.

complex         valence        aqueous conc.   sorption(io)      sorption(edl)
HCO3-         -1.000         0.8649e-04     0.               0.
H2CO3         0.             0.2182e-07     0.               0.

 precipitate   equivalent moles/liter solution

CaCO3     0.100e-01

   to have a net balance of zero charge, other 'species must be present (although not participating in the
   chemical reactions) with a total concentration times charge of  0.
1                               boundary or influx conditions
                                        temp=   10.00
component       total          bare ion        sorbed concentration        act. coef.    kd(ion ex)       kd(edl)
             aqueous conc.   concentration      (ionex)     (edl)

Na+           0.1000e-02     0.1000e-02     0.         0.                  0.9647        0.              0.
H+            0.3000e-02     0.1000e-04     0.         0.                  0.9647        0.              0.
Co3--         0.1529e-02     0.3617e-09     0.         0.                  0.8661        0.              0.
neg.KI        0.3899e-01     0.3899e-01     0.         0.                   1.000        0.              0.
Ca++          0.1000e-34     0.1000e-34     0.         0.                  0.8661        0.              0.
Cl-           0.9930e-03     0.9930e-03     0.         0.                  0.9647        0.              0.
OH-           0.1100e-08     0.1100e-08     0.         0.                  0.9647        0.              0.

complex         valence        aqueous conc.   sorption(io)      sorption(edl)
HCO3-         -1.000         0.6806e-04     0.               0.
H2CO3         0.             0.1461e-02     0.               0.

 precipitate   equivalent moles/liter solution

CaCO3     0.

   to have a net balance of zero charge, other 'species must be present (although not participating in the
   chemical reactions) with a total concentration times charge of  0.510e-04
```

OUTPUT

```
1     time= 1.53364e-05           delt=  6.5728e-06          concentration of species in moles/liter solution


                                                              distance
   species        0.000       0.005       0.012       0.023       0.037       0.058       0.088       0.131       0.192

  Na+          1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03
  H+           1.3124e-10  1.1913e-10  1.1878e-10  1.1878e-10  1.1878e-10  1.1877e-10  1.1877e-10  1.1877e-10  1.1877e-10
  Co3--        3.9226e-05  3.9528e-05  3.9531e-05  3.9532e-05  3.9532e-05  3.9532e-05  3.9533e-05  3.9534e-05  3.9535e-05
  neg.KI       2.2358e-04  6.9224e-06  9.0205e-08  6.3719e-10  1.0154e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10
  Ca++         1.2532e-04  1.2602e-04  1.2604e-04  1.2604e-04  1.2604e-04  1.2604e-04  1.2604e-04  1.2605e-04  1.2605e-04
  Cl-          9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04
  OH-          8.4680e-05  9.3288e-05  9.3565e-05  9.3569e-05  9.3570e-05  9.3570e-05  9.3571e-05  9.3572e-05  9.3573e-05
shftsol       -6.6748e-01  9.4519e-01  9.9928e-01  9.9999e-01  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00
shift0       -3.0951e+00 -1.6555e+00 -1.6068e+00 -1.6062e+00 -1.6062e+00 -1.6062e+00 -1.6062e+00 -1.6062e+00 -1.6063e+00
shift1        6.9231e+00  8.3772e+00  8.4263e+00  8.4270e+00  8.4270e+00  8.4269e+00  8.4269e+00  8.4269e+00  8.4269e+00
shift2        5.1964e+00  6.6480e+00  6.6970e+00  6.6977e+00  6.6977e+00  6.6977e+00  6.6977e+00  6.6976e+00  6.6976e+00

   complexes

  HCO3-        9.4832e-05  8.6739e-05  8.6490e-05  8.6487e-05  8.6487e-05  8.6488e-05  8.6489e-05  8.6490e-05  8.6491e-05
  H2CO3        2.6435e-08  2.1948e-08  2.1820e-08  2.1818e-08  2.1818e-08  2.1818e-08  2.1818e-08  2.1818e-08  2.1818e-08

 precipitates

  CaCO3        9.9960e-03  9.9960e-03  9.9960e-03  9.9960e-03  9.9960e-03  9.9960e-03  9.9960e-03  9.9960e-03  9.9960e-03


                                                              distance
   species        0.278       0.401       0.577       0.827       1.184       1.692       2.416       3.448       4.918

  Na+          1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03
  H+           1.1877e-10  1.1876e-10  1.1876e-10  1.1875e-10  1.1874e-10  1.1873e-10  1.1871e-10  1.1868e-10  1.1864e-10
  Co3--        3.9537e-05  3.9540e-05  3.9543e-05  3.9549e-05  3.9556e-05  3.9567e-05  3.9582e-05  3.9602e-05  3.9631e-05
  neg.KI       1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10
  Ca++         1.2605e-04  1.2606e-04  1.2607e-04  1.2608e-04  1.2609e-04  1.2612e-04  1.2615e-04  1.2619e-04  1.2625e-04
  Cl-          9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04
  OH-          9.3575e-05  9.3578e-05  9.3583e-05  9.3588e-05  9.3597e-05  9.3609e-05  9.3626e-05  9.3649e-05  9.3681e-05
shftsol        1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00
shift0       -1.6063e+00 -1.6064e+00 -1.6064e+00 -1.6066e+00 -1.6067e+00 -1.6069e+00 -1.6073e+00 -1.6077e+00 -1.6083e+00
shift1        8.4268e+00  8.4268e+00  8.4267e+00  8.4266e+00  8.4264e+00  8.4262e+00  8.4259e+00  8.4254e+00  8.4248e+00
shift2        6.6976e+00  6.6975e+00  6.6974e+00  6.6973e+00  6.6971e+00  6.6969e+00  6.6966e+00  6.6961e+00  6.6955e+00

   complexes

  HCO3-        8.6493e-05  8.6496e-05  8.6500e-05  8.6506e-05  8.6515e-05  8.6527e-05  8.6543e-05  8.6567e-05  8.6599e-05
  H2CO3        2.1818e-08  2.1818e-08  2.1818e-08  2.1819e-08  2.1819e-08  2.1819e-08  2.1819e-08  2.1820e-08  2.1820e-08

 precipitates

  CaCO3        9.9959e-03  9.9959e-03  9.9959e-03  9.9959e-03  9.9959e-03  9.9959e-03  9.9959e-03  9.9958e-03  9.9957e-03
```

```
                                                   distance
  species       7.014      10.000
   Na+        1.0000e-03  1.0000e-03
   H+         1.1858e-10  1.1878e-10
   Co3--      3.9670e-05  3.9531e-05
   neg.KI     1.0000e-10  1.0000e-10
   Ca++       1.2634e-04  1.2604e-04
   Cl-        9.9296e-04  9.9296e-04
   OH-        9.3725e-05  9.3568e-05
shftsol       1.0000e+00  1.0000e+00
shift0       -1.6092e+00 -1.6062e+00
shift1        8.4239e+00  8.4270e+00
shift2        6.6946e+00  6.6977e+00

  complexes

   HCO3-      8.6643e-05  8.6486e-05
   H2CO3      2.1821e-08  2.1818e-08

 precipitates

   CaCO3      9.9957e-03  9.9960e-03

1    time= 9.59502e-01          delt=  1.0000e-03          concentration of species in moles/liter solution


                                                   distance
  species       0.000       0.005       0.012       0.023       0.037       0.058       0.088       0.131       0.192

   Na+        1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03
   H+         4.0317e-07  3.7337e-07  3.3418e-07  2.8477e-07  2.2620e-07  1.6272e-07  1.0240e-07  5.6526e-08  3.4790e-08
   Co3--      1.5940e-07  1.8116e-07  2.1735e-07  2.8109e-07  4.0213e-07  6.5450e-07  1.2370e-06  2.6046e-06  4.5485e-06
   neg.KI     3.8759e-02  3.8747e-02  3.8730e-02  3.8703e-02  3.8663e-02  3.8598e-02  3.8492e-02  3.8309e-02  3.7970e-02
   Ca++       5.3053e-04  5.5805e-04  5.9856e-04  6.5816e-04  7.4483e-04  8.6627e-04  1.0210e-03  1.1761e-03  1.2592e-03
   Cl-        9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04
   OH-        2.8354e-08  3.0666e-08  3.4341e-08  4.0433e-08  5.1138e-08  7.1523e-08  1.1449e-07  2.0882e-07  3.4048e-07
shftsol      -1.7903e+01 -1.7651e+01 -1.7292e+01 -1.6787e+01 -1.6100e+01 -1.5218e+01 -1.4212e+01 -1.3303e+01 -1.2816e+01
shift0      -2.2366e+01 -2.1926e+01 -2.1298e+01 -2.0416e+01 -1.9213e+01 -1.7669e+01 -1.5905e+01 -1.4316e+01 -1.3471e+01
shift1      -1.2542e+01 -1.2097e+01 -1.1463e+01 -1.0572e+01 -9.3573e+00 -7.7977e+00 -6.0157e+00 -4.4104e+00 -3.5574e+00
shift2      -1.4235e+01 -1.3791e+01 -1.3158e+01 -1.2269e+01 -1.1056e+01 -9.4992e+00 -7.7202e+00 -6.1176e+00 -5.2661e+00

  complexes

   HCO3-      1.1189e-03  1.1738e-03  1.2547e-03  1.3737e-03  1.5467e-03  1.7889e-03  2.0969e-03  2.4041e-03  2.5658e-03
   H2CO3      9.3146e-04  9.0353e-04  8.6244e-04  8.0195e-04  7.1393e-04  5.9038e-04  4.3234e-04  2.7175e-04  1.7788e-04

 precipitates

   CaCO3      8.8854e-03  7.3739e-03  6.4036e-03  5.1685e-03  3.7090e-03  2.2616e-03  1.6124e-03  3.8159e-03  8.1646e-03
```

OUTPUT

```
                                                   distance
  species       0.278       0.401       0.577       0.827       1.184       1.692       2.416       3.448       4.918

   Na+        1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03
   H+         2.9503e-08  2.7295e-08  2.2853e-08  1.2491e-08  2.5866e-09  2.8882e-10  1.2993e-10  1.1800e-10  1.1725e-10
   Co3--      5.3675e-06  5.5907e-06  6.0126e-06  7.7430e-06  1.5425e-05  3.6023e-05  4.0673e-05  4.0668e-05  4.0662e-05
   neg.KI     3.7289e-02  3.5783e-02  3.2031e-02  2.2355e-02  9.3401e-03  2.0714e-03  2.4407e-04  1.5860e-05  5.9138e-07
   Ca++       1.2613e-03  1.2201e-03  1.1102e-03  8.0746e-04  3.6297e-04  1.4593e-04  1.2842e-04  1.2842e-04  1.2844e-04
   Cl-        9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04
   OH-        4.0153e-07  4.3326e-07  5.1501e-07  9.2882e-07  4.3713e-06  3.8542e-05  8.5559e-05  9.4209e-05  9.4812e-05
shftsol      -1.2691e+01 -1.2640e+01 -1.2536e+01 -1.2274e+01 -1.1808e+01 -8.1188e+00 -7.6875e-01  8.7710e-01  9.9540e-01
shift0      -1.3257e+01 -1.3169e+01 -1.2991e+01 -1.2556e+01 -1.2034e+01 -9.4304e+00 -3.2143e+00 -1.7413e+00 -1.6347e+00
shift1      -3.3408e+00 -3.2520e+00 -3.0719e+00 -2.6334e+00 -2.1055e+00  5.2413e-01  6.8027e+00  8.2905e+00  8.3981e+00
shift2      -5.0499e+00 -4.9612e+00 -4.7815e+00 -4.3437e+00 -3.8167e+00 -1.1916e+00  5.0762e+00  6.5614e+00  6.6689e+00

  complexes

   HCO3-      2.5673e-03  2.4825e-03  2.2568e-03  1.6348e-03  7.0999e-04  1.9102e-04  9.7290e-05  8.8349e-05  8.7775e-05
   H2CO3      1.5092e-04  1.3525e-04  1.0344e-04  4.1547e-05  3.8339e-06  1.1699e-07  2.6841e-08  2.2137e-08  2.1853e-08

 precipitates

   CaCO3      9.8299e-03  1.0013e-02  1.0020e-02  1.0020e-02  1.0032e-02  1.0016e-02  9.9947e-03  9.9936e-03  9.9936e-03

                                                   distance
  species       7.014       10.000
   Na+        1.0000e-03  1.0000e-03
   H+         1.1722e-10  1.1878e-10
   Co3--      4.0662e-05  3.9531e-05
   neg.KI     1.3181e-08  1.0000e-10
   Ca++       1.2844e-04  1.2604e-04
   Cl-        9.9296e-04  9.9296e-04
   OH-        9.4835e-05  9.3568e-05
shftsol       9.9990e-01  1.0000e+00
shift0      -1.6307e+00 -1.6062e+00
shift1       8.4022e+00  8.4270e+00
shift2       6.6730e+00  6.6977e+00

  complexes

   HCO3-      8.7753e-05  8.6486e-05
   H2CO3      2.1842e-08  2.1818e-08

 precipitates

   CaCO3      9.9940e-03  9.9960e-03
```

APPENDIX 3

EQUILIBRIUM CALCITE DISSOLUTION DURING REACTIVE TRANSPORT

INPUT AND SAMPLE OUTPUT

INPUT

```
Partitioned Isothl.Equi.Calcite Diss.Tran.Int.shitsol=-24.5,shifts=1.0-sdyparte
UNITS      meter    year
 type              0         1         0         1
iteration        100       100       100
Calc.crtia   1.0e-30   1.0e-03        20        10         5      1.75         5
grid              20      10.0  836.6835
time              14 1.585e-10   1.0e-03
print time 1.142e-05 1.142e-04 1.141e-03 4.566e-03 1.141e-02 2.740e-02 5.479e-02
print time 1.370e-01 2.740e-01 4.110e-01 5.480e-01 6.849e-01 8.219e-01 9.589e-01
Physical         0.1   1.0e-10   0.1e+01
tran bound         1         0
temp bound         0
isothermal         0
Intl. temp      10.0
leach              0         0
WORD      blank
In.Sh.Info     -24.5       1.0
Iso.Ratio1 -0.91e-03       6.3
         2 -4.54e-03    1099.0
         3 -3.40e-03     870.0
         4 -3.63e-03    1194.0
no.species         7         0         0         2         1         2         0
Na+       1.0    0    2   1.0e-03   1.0e-03    2   1.0e-03   1.0e-03
H+        1.0    0    2 8.653e-05 1.237e-10    1   1.0e-05   1.0e-05
Co3--    -2.0   31    3 1.0001217 3.521e-05    3 1.529e-03 1.865e-10
neg.KI    0.0   32    1 1.000e-10 1.000e-10    1 3.899e-02 3.899e-02
Ca++      2.0    0    3 1.0001217 1.217e-04    3       0.0       0.0
Cl-      -1.0    0    4   1.0e-03   1.0e-03    0   1.0e-03   1.0e-03
OH-        -13.990       0.0       0.0       0.0
           8.653e-05 3.296e-10
HCO3-      33  0.0  1.0  1.0  0.0  0.0  0.0  0.0
              10.337       0.0       0.0       0.0
           8.649e-05 5.637e-05
H2CO3      34  0.0  2.0  1.0  0.0  0.0  0.0  0.0
              16.700       0.0       0.0       0.0
           2.359e-08 1.473e-03
CaCO3      10  0.0  0.0  1.0  0.0  1.0  0.0  0.0
              -8.446       0.0       0.0       0.0
                 1.0       0.0
```

OUTPUT

```
1Partitioned lsothl.Equi.Calcite Diss.Tran.Int.shitsol=-24.5,shifts=1.0-sdyparte

    unit of length is   meter              unit of time is    year

    inner boundary is constant flux          outer' boundry is constant initial concentration


    dispersion coefficient= 0.1v+  0.10e-09     fluid velocity =  1.00         number of grid points =   20


    number of print outs =   14
    given at the following times= 0.1142e-040.1142e-030.1141e-020.4566e-020.1141e-010.2740e-010.5479e-010.1370
                                  0.2740    0.4110    0.5480    0.6849    0.8219    0.9589

errsqr=   0.10e-29     climit=   0.10e-02     iterj=    20     iterm=    10     iterd=     5     dtmult=   0.17e+01     ndiv=     5

    number of sites(ion-exchange)    0.0000            number of sites(edl model)    0.0000
 this simulation is isothermal, temperature =  10.



      6      bare ions             0     sorbing ions (ionex)            0     sorbing ions (edl model)

component charge      kinetic     initial condition        boundary condition
                         code       type        value         type        value

Na+           1.         0         2       1.0000e-03         2       1.0000e-03
H+            1.         0         2       8.6530e-05         1       1.0000e-05
Co3--        -2.        31         3       1.0001e+00         3       1.5290e-03
neg.KI        0.        32         1       1.0000e-10         1       3.8990e-02
Ca++          2.         0         3       1.0001e+00         3       0.
Cl-          -1.         0         4       1.0000e-03         0       1.0000e-03
```

OUTPUT

```
     3      complexes              0       sorbing complexes(ionex)             0     sorbing complexes (edl model)

complex   charge       kcode                          stoichiometry

OH-         -1.          0         0.   0.   0.   0.   0.   0.   1.
HCO3-       -1.         33         0.   1.   1.   0.   0.   0.   0.
H2CO3        0.         34         0.   2.   1.   0.   0.   0.   0.


      1       solid precipitates

  precipitate  kinetic code          stoichiometry

CaCO3            10           0.0  0.0  1.0  0.0  1.0  0.0  0.0

          C-13 Fractionation parameters:

          shiftr=   -24.5000          shifts=     1.0000

coefficients in log function of thermodynamic constants
              -0.001     6.300
              -0.005  1039.000
              -0.003   870.000
              -0.004  1194.000

                                                   ** spatial grid **

    0.          0.5086e-02   0.1233e-01   0.2266e-01   0.3738e-01   0.5835e-01   0.8823e-01   0.1308       0.1915       0.2780
    0.4012      0.5768       0.8271        1.184        1.692        2.416        3.448        4.918        7.014        10.00
```

OUTPUT

```
1                                                          initial conditions
                                                             temp=   10.00

component       total           bare ion         sorbed concentration         act. coef.      kd(ion ex)          kd(edl)
              aqueous conc.    concentration      (ionex)     (edl)

Na+           0.1000e-02      0.1000e-02      0.         0.                    0.9596          0.                0.
H+            0.8653e-04      0.1188e-09      0.         0.                    0.9596          0.                0.
Co3--         0.1260e-03      0.3953e-04      0.         0.                    0.8478          0.                0.
neg.KI        0.1000e-09      0.1000e-09      0.         0.                     1.000          0.                0.
Ca++          0.1260e-03      0.1260e-03      0.         0.                    0.8478          0.                0.
Cl-           0.9930e-03      0.9930e-03      0.         0.                    0.9596          0.                0.
OH-           0.9357e-04      0.9357e-04      0.         0.                    0.9596          0.                0.

complex         valence         aqueous conc.   sorption(io)      sorption(edl)
HCO3-         -1.000          0.8649e-04      0.             0.
H2CO3         0.              0.2182e-07      0.             0.

 precipitate   equivalent moles/liter solution

CaCO3     1.00


   to have a net balance of zero charge, other 'species must be present (although not participating in the
   chemical reactions) with a total concentration times charge of  0.


1                                  boundary or influx conditions
                                          temp=   10.00
component       total           bare ion         sorbed concentration         act. coef.      kd(ion ex)          kd(edl)
              aqueous conc.    concentration      (ionex)     (edl)

Na+           0.1000e-02      0.1000e-02      0.         0.                    0.9647          0.                0.
H+            0.3000e-02      0.1000e-04      0.         0.                    0.9647          0.                0.
Co3--         0.1529e-02      0.3617e-09      0.         0.                    0.8661          0.                0.
neg.KI        0.3899e-01      0.3899e-01      0.         0.                     1.000          0.                0.
Ca++          0.1000e-34      0.1000e-34      0.         0.                    0.8661          0.                0.
Cl-           0.9930e-03      0.9930e-03      0.         0.                    0.9647          0.                0.
OH-           0.1100e-08      0.1100e-08      0.         0.                    0.9647          0.                0.

complex         valence         aqueous conc.   sorption(io)      sorption(edl)
HCO3-         -1.000          0.6806e-04      0.             0.
H2CO3         0.              0.1461e-02      0.             0.

 precipitate   equivalent moles/liter solution

CaCO3    0.


   to have a net balance of zero charge, other 'species must be present (although not participating in the
   chemical reactions) with a total concentration times charge of  0.510e-04
```

OUTPUT

```
1    time= 1.53364e-05          delt=  6.5728e-06         concentration of species in moles/liter solution


                                                          distance
   species         0.000       0.005       0.012       0.023       0.037       0.058       0.088       0.131       0.192

   Na+          1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03
   H+           1.3069e-10  1.1913e-10  1.1878e-10  1.1878e-10  1.1878e-10  1.1878e-10  1.1878e-10  1.1878e-10  1.1878e-10
   Co3--        3.9545e-05  3.9534e-05  3.9531e-05  3.9531e-05  3.9531e-05  3.9531e-05  3.9531e-05  3.9531e-05  3.9531e-05
   neg.KI       2.2358e-04  6.9224e-06  9.0205e-08  6.3719e-10  1.0154e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10
   Ca++         1.2600e-04  1.2603e-04  1.2604e-04  1.2604e-04  1.2604e-04  1.2604e-04  1.2604e-04  1.2604e-04  1.2604e-04
   Cl-          9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04
   OH-          8.5041e-05  9.3295e-05  9.3565e-05  9.3568e-05  9.3568e-05  9.3568e-05  9.3568e-05  9.3568e-05  9.3568e-05
shftsol        -6.5907e-01  9.4519e-01  9.9928e-01  9.9999e-01  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00
shift0        -3.0941e+00 -1.6556e+00 -1.6068e+00 -1.6062e+00 -1.6062e+00 -1.6062e+00 -1.6062e+00 -1.6062e+00 -1.6062e+00
shift1         6.9241e+00  8.3770e+00  8.4263e+00  8.4270e+00  8.4270e+00  8.4270e+00  8.4270e+00  8.4270e+00  8.4270e+00
shift2         5.1974e+00  6.6478e+00  6.6971e+00  6.6977e+00  6.6977e+00  6.6977e+00  6.6977e+00  6.6977e+00  6.6977e+00

  complexes

   HCO3-        9.5193e-05  8.6746e-05  8.6490e-05  8.6486e-05  8.6486e-05  8.6486e-05  8.6486e-05  8.6486e-05  8.6486e-05
   H2CO3        2.6423e-08  2.1948e-08  2.1820e-08  2.1818e-08  2.1818e-08  2.1818e-08  2.1818e-08  2.1818e-08  2.1818e-08

 precipitates

   CaCO3        9.9999e-01  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00

                                                          distance
   species         0.278       0.401       0.577       0.827       1.184       1.692       2.416       3.448       4.918

   Na+          1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03
   H+           1.1878e-10  1.1878e-10  1.1878e-10  1.1878e-10  1.1878e-10  1.1878e-10  1.1878e-10  1.1878e-10  1.1878e-10
   Co3--        3.9531e-05  3.9531e-05  3.9531e-05  3.9531e-05  3.9531e-05  3.9531e-05  3.9531e-05  3.9531e-05  3.9531e-05
   neg.KI       1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10
   Ca++         1.2604e-04  1.2604e-04  1.2604e-04  1.2604e-04  1.2604e-04  1.2604e-04  1.2604e-04  1.2604e-04  1.2604e-04
   Cl-          9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04
   OH-          9.3568e-05  9.3568e-05  9.3568e-05  9.3568e-05  9.3568e-05  9.3568e-05  9.3568e-05  9.3568e-05  9.3568e-05
shftsol         1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00
shift0        -1.6062e+00 -1.6062e+00 -1.6062e+00 -1.6062e+00 -1.6062e+00 -1.6062e+00 -1.6062e+00 -1.6062e+00 -1.6062e+00
shift1         8.4270e+00  8.4270e+00  8.4270e+00  8.4270e+00  8.4270e+00  8.4270e+00  8.4270e+00  8.4270e+00  8.4270e+00
shift2         6.6977e+00  6.6977e+00  6.6977e+00  6.6977e+00  6.6977e+00  6.6977e+00  6.6977e+00  6.6977e+00  6.6977e+00

  complexes

   HCO3-        8.6486e-05  8.6486e-05  8.6486e-05  8.6486e-05  8.6486e-05  8.6486e-05  8.6486e-05  8.6486e-05  8.6486e-05
   H2CO3        2.1818e-08  2.1818e-08  2.1818e-08  2.1818e-08  2.1818e-08  2.1818e-08  2.1818e-08  2.1818e-08  2.1818e-08

 precipitates

   CaCO3        1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00
```

```
                                                      distance
  species        7.014      10.000
   Na+        1.0000e-03  1.0000e-03
   H+         1.1878e-10  1.1878e-10
   Co3--      3.9531e-05  3.9531e-05
   neg.KI     1.0000e-10  1.0000e-10
   Ca++       1.2604e-04  1.2604e-04
   Cl-        9.9296e-04  9.9296e-04
   OH-        9.3568e-05  9.3568e-05
shftsol       1.0000e+00  1.0000e+00
shift0      -1.6062e+00 -1.6062e+00
shift1       8.4270e+00  8.4270e+00
shift2       6.6977e+00  6.6977e+00

  complexes

   HCO3-      8.6486e-05  8.6486e-05
   H2CO3      2.1818e-08  2.1818e-08

 precipitates

   CaCO3      1.0000e+00  1.0000e+00


1     time= 9.58995e-01          delt=  1.0000e-03          concentration of species in moles/liter solution


                                                      distance
  species        0.000       0.005       0.012       0.023       0.037       0.058       0.088       0.131       0.192

   Na+        1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03
   H+         3.2701e-08  3.2686e-08  3.2662e-08  3.2627e-08  3.2574e-08  3.2489e-08  3.2350e-08  3.2109e-08  3.1664e-08
   Co3--      4.9922e-06  4.9932e-06  4.9946e-06  4.9968e-06  5.0002e-06  5.0056e-06  5.0145e-06  5.0300e-06  5.0592e-06
   neg.KI     3.8759e-02  3.8747e-02  3.8729e-02  3.8703e-02  3.8662e-02  3.8597e-02  3.8491e-02  3.8308e-02  3.7968e-02
   Ca++       1.2958e-03  1.2955e-03  1.2950e-03  1.2942e-03  1.2931e-03  1.2914e-03  1.2884e-03  1.2834e-03  1.2740e-03
   Cl-        9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04
   OH-        3.6278e-07  3.6295e-07  3.6320e-07  3.6358e-07  3.6416e-07  3.6508e-07  3.6661e-07  3.6928e-07  3.7433e-07
shftsol      -1.2765e+01 -1.2765e+01 -1.2764e+01 -1.2763e+01 -1.2762e+01 -1.2760e+01 -1.2757e+01 -1.2752e+01 -1.2742e+01
shift0      -1.3384e+01 -1.3384e+01 -1.3383e+01 -1.3381e+01 -1.3379e+01 -1.3376e+01 -1.3371e+01 -1.3361e+01 -1.3344e+01
shift1      -3.4695e+00 -3.4689e+00 -3.4680e+00 -3.4667e+00 -3.4646e+00 -3.4612e+00 -3.4558e+00 -3.4463e+00 -3.4288e+00
shift2      -5.1784e+00 -5.1778e+00 -5.1769e+00 -5.1755e+00 -5.1734e+00 -5.1701e+00 -5.1647e+00 -5.1552e+00 -5.1378e+00

  complexes

   HCO3-      2.6390e-03  2.6383e-03  2.6374e-03  2.6359e-03  2.6336e-03  2.6299e-03  2.6239e-03  2.6135e-03  2.5943e-03
   H2CO3      1.7171e-04  1.7159e-04  1.7141e-04  1.7113e-04  1.7071e-04  1.7004e-04  1.6895e-04  1.6706e-04  1.6359e-04

 precipitates

   CaCO3      4.9410e-01  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00
```

OUTPUT

```
                                                           distance
    species        0.278       0.401       0.577       0.827       1.184       1.692       2.416       3.448       4.918

   Na+         1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03
   H+          3.0775e-08  2.8827e-08  2.4105e-08  1.3110e-08  2.6959e-09  2.9611e-10  1.3181e-10  1.1957e-10  1.1881e-10
   Co3--       5.1192e-06  5.2601e-06  5.6687e-06  7.3504e-06  1.4762e-05  3.4879e-05  3.9539e-05  3.9537e-05  3.9531e-05
   neg.KI      3.7286e-02  3.5777e-02  3.2019e-02  2.2335e-02  9.3257e-03  2.0668e-03  2.4336e-04  1.5803e-05  5.8893e-07
   Ca++        1.2551e-03  1.2127e-03  1.1042e-03  8.0436e-04  3.6143e-04  1.4365e-04  1.2602e-04  1.2602e-04  1.2604e-04
   Cl-         9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04  9.9296e-04
   OH-         3.8483e-07  4.1010e-07  4.8811e-07  8.8478e-07  4.1936e-06  3.7585e-05  8.4318e-05  9.2944e-05  9.3545e-05
shftsol       -1.2722e+01 -1.2678e+01 -1.2568e+01 -1.2293e+01 -1.1825e+01 -8.1980e+00 -7.9521e-01  8.7521e-01  9.9533e-01
shift0       -1.3309e+01 -1.3233e+01 -1.3044e+01 -1.2585e+01 -1.2046e+01 -9.4820e+00 -3.2153e+00 -1.7187e+00 -1.6104e+00
shift1       -3.3938e+00 -3.3165e+00 -3.1261e+00 -2.6627e+00 -2.1182e+00  4.7197e-01  6.8017e+00  8.3133e+00  8.4227e+00
shift2       -5.1028e+00 -5.0257e+00 -4.8356e+00 -4.3729e+00 -3.8294e+00 -1.2437e+00  5.0752e+00  6.5842e+00  6.6935e+00

    complexes

   HCO3-       2.5554e-03  2.4684e-03  2.2456e-03  1.6294e-03  7.0839e-04  1.8970e-04  9.5994e-05  8.7080e-05  8.6508e-05
   H2CO3       1.5674e-04  1.4208e-04  1.0859e-04  4.3471e-05  3.9874e-06  1.1914e-07  2.6873e-08  2.2116e-08  2.1829e-08

  precipitates

   CaCO3       1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00  1.0000e+00

                                                           distance
    species        7.014      10.000
   Na+         1.0000e-03  1.0000e-03
   H+          1.1878e-10  1.1878e-10
   Co3--       3.9531e-05  3.9531e-05
   neg.KI      1.3119e-08  1.0000e-10
   Ca++        1.2604e-04  1.2604e-04
   Cl-         9.9296e-04  9.9296e-04
   OH-         9.3568e-05  9.3568e-05
shftsol        9.9990e-01  1.0000e+00
shift0       -1.6063e+00 -1.6062e+00
shift1        8.4269e+00  8.4270e+00
shift2        6.6976e+00  6.6977e+00

    complexes

   HCO3-       8.6487e-05  8.6486e-05
   H2CO3       2.1818e-08  2.1818e-08

  precipitates

   CaCO3       1.0000e+00  1.0000e+00
```

APPENDIX 4

KINETIC CALCITE DISSOLUTION IN BATCH REACTIONS

INPUT AND SAMPLE OUTPUT

INPUT

```
Partitioned Isothl.Kin. Calcite Diss.NoTr.Int.shitsol=-24.5,shifts=1.0-sdypartk
 UNITS      meter    year
 type              0         1         0         1
iteration        100       100       100
Calc.crtia   1.0e-30   1.0e-03        20        10         5      1.75         5
grid               5      6.56       0.0
time              14 1.585e-10   1.0e-03
print time 1.142e-05 1.142e-04 1.141e-03 4.566e-03 1.141e-02 2.740e-02 5.479e-02
print time 1.370e-01 2.740e-01 4.110e-01 5.480e-01 6.849e-01 8.219e-01 9.589e-01
Physical     1.0e-20   1.0e-20   1.0e-20
tran bound         1          0
temp bound         0
isothermal         0
Intl. temp      10.0
leach              0          0
WORD      tpkin
**Carbonate Diss. Rate Data next 7 lines
1 PARAMS           5     100.0         4     prepc
2 CDATS         19.8     6.383    10.377     8.446 3.154e+04
3 INXI(i)     2    0   09    0    0    0    5    8
Carbonate1     -1.32         0         0         0
         2     -4.59         0         0         0
         3     -6.94         0         0         0
         4       0.0         0         0         0
WORD      blank
In.Sh.Info     -24.5       1.0
Iso.Ratio1 -0.91e-03       6.3
         2 -4.54e-03    1099.0
         3 -3.40e-03     870.0
         4 -3.63e-03    1194.0
no.species         7         0         0         2         1         2         0
Na+       1.0    0    2   1.0e-03   1.0e-03     2   1.0e-03   1.0e-03
H+        1.0    0    1 1.000e-05 1.000e-05     1   1.0e-05   1.0e-05
Co3--    -2.0   31    3 1.529e-03 1.865e-10     3 1.529e-03 1.865e-10
neg.KI    0.0   32    1 3.899e-02 3.899e-02     1 3.899e-02 3.899e-02
Ca++      2.0    0    3 1.000e-19 1.000e-19     3       0.0       0.0
Cl-      -1.0    0    1   1.0e-03   1.0e-03     0   1.0e-03   1.0e-03
OH-        -13.990         0.0       0.0       0.0
           3.296e-10 3.296e-10
HCO3-      33  0.0  1.0  1.0  0.0  0.0  0.0  0.0
              10.337       0.0       0.0       0.0
           5.637e-05 5.637e-05
H2CO3      34  0.0  2.0  1.0  0.0  0.0  0.0  0.0
              16.700       0.0       0.0       0.0
           1.473e-03 1.473e-03
CaCO3      10  0.0  0.0  1.0  0.0  1.0  0.0  0.0
              -8.446       0.0       0.0       0.0
                 0.0       0.0
```

OUTPUT

```
1Partitioned Isothl.Kin. Calcite Diss.NoTr.Int.shitsol=-24.5,shifts=1.0-sdypartk

    unit of length is    meter           unit of time is     year

kinetics of rev. precipitate dissolution considered


    inner boundary is constant flux          outer' boundry is constant initial concentration


    dispersion coefficient= 0.0v+  0.10e-19     fluid velocity = 0.100e-19     number of grid points =    5


    number of print outs =   14
    given at the following times= 0.1142e-040.1142e-030.1141e-020.4566e-020.1141e-010.2740e-010.5479e-010.1370
                                  0.2740    0.4110    0.5480    0.6849    0.8219    0.9589

errsqr=   0.10e-29     climit=   0.10e-02     iterj=    20     iterm=    10     iterd=    5     dtmult=   0.17e+01     ndiv=

    number of sites(ion-exchange)    0.0000          number of sites(edl model)    0.0000
 this simulation is isothermal, temperature =  10.



     6      bare ions             0     sorbing ions (ionex)            0     sorbing ions (edl model)


component charge      kinetic     initial condition       boundary condition
                        code        type        value        type        value

Na+          1.         0          2       1.0000e-03        2       1.0000e-03
H+           1.         0          1       1.0000e-05        1       1.0000e-05
Co3--       -2.        31          3       1.5290e-03        3       1.5290e-03
neg.KI       0.        32          1       3.8990e-02        1       3.8990e-02
Ca++         2.         0          3       1.0000e-19        3       0.
Cl-         -1.         0          1       1.0000e-03        0       1.0000e-03


     3      complexes             0       sorbing complexes(ionex)            0     sorbing complexes (edl model)


complex   charge       kcode                           stoichiometry

OH-        -1.          0        0.    0.    0.    0.    0.    0.    1.
HCO3-      -1.         33        0.    1.    1.    0.    0.    0.    0.
H2CO3       0.         34        0.    2.    1.    0.    0.    0.    0.


      1       solid precipitates
```

OUTPUT

```
  precipitate  kinetic code          stoichiometry
CaCO3            10           0.0  0.0  1.0  0.0  1.0  0.0  0.0
Calcite dissolution information:


 no of terms in the rate formulla =     4
 no of other additional data needed =     5
 representative surface parameter =    0.100e+03
 xdiag = prepc

 additional data for rate calculation
   0.198e+02   0.638e+01   0.104e+02   0.845e+01   0.315e+05
binary element nos. of the terms of the rate equation
              2    0    9    0    0    0    5    8

coefficients in log function of thermodynamic constants
              -1.320     0.000     0.000     0.000
              -4.590     0.000     0.000     0.000
              -6.940     0.000     0.000     0.000
               0.000     0.000     0.000     0.000
          C-13 Fractionation parameters:
          shiftr=   -24.5000          shifts=     1.0000
coefficients in log function of thermodynamic constants
              -0.001     6.300
              -0.005  1099.000
              -0.003   870.000
              -0.004  1194.000
                                                      ** spatial grid **
    0.           1.640        3.280        4.920        6.560
```

OUTPUT

```
1                                                     initial conditions
                                                        temp=   10.00

component        total           bare ion        sorbed concentration        act. coef.     kd(ion ex)        kd(edl)
               aqueous conc.    concentration     (ionex)     (edl)

Na+            0.1000e-02     0.1000e-02     0.          0.                   0.9646         0.               0.
H+             0.3000e-02     0.1000e-04     0.          0.                   0.9646         0.               0.
Co3--          0.1529e-02     0.3619e-09     0.          0.                   0.8657         0.               0.
neg.KI         0.3899e-01     0.3899e-01     0.          0.                    1.000         0.               0.
Ca++           0.1000e-18     0.1000e-18     0.          0.                   0.8657         0.               0.
Cl-            0.1000e-02     0.1000e-02     0.          0.                   0.9646         0.               0.
OH-            0.1100e-08     0.1100e-08     0.          0.                   0.9646         0.               0.

complex          valence        aqueous conc.   sorption(io)      sorption(edl)
HCO3-          -1.000         0.6807e-04     0.              0.
H2CO3          0.             0.1461e-02     0.              0.

 precipitate   equivalent moles/liter solution

CaCO3     1.00

   to have a net balance of zero charge, other 'species must be present (although not participating in the
   chemical reactions) with a total concentration times charge of  0.581e-04
1                              boundary or influx conditions
                                         temp=   10.00
component        total           bare ion        sorbed concentration        act. coef.     kd(ion ex)        kd(edl)
               aqueous conc.    concentration     (ionex)     (edl)

Na+            0.1000e-02     0.1000e-02     0.          0.                   0.9646         0.               0.
H+             0.3000e-02     0.1000e-04     0.          0.                   0.9646         0.               0.
Co3--          0.1529e-02     0.3619e-09     0.          0.                   0.8657         0.               0.
neg.KI         0.3899e-01     0.3899e-01     0.          0.                    1.000         0.               0.
Ca++           0.1000e-34     0.1000e-34     0.          0.                   0.8657         0.               0.
Cl-            0.1000e-02     0.1000e-02     0.          0.                   0.9646         0.               0.
OH-            0.1100e-08     0.1100e-08     0.          0.                   0.9646         0.               0.

complex          valence        aqueous conc.   sorption(io)      sorption(edl)
HCO3-          -1.000         0.6807e-04     0.              0.
H2CO3          0.             0.1461e-02     0.              0.

 precipitate   equivalent moles/liter solution

CaCO3     0.

   to have a net balance of zero charge, other 'species must be present (although not participating in the
   chemical reactions) with a total concentration times charge of  0.581e-04
```

OUTPUT

```
1    time= 1.53364e-05          delt=  6.5728e-06          concentration of species in moles/liter solution


                                                      distance
   species        0.000       1.640       3.280       4.920       6.560        0.000        0.000        0.000        0.000

    Na+         1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  0.           0.           0.           0.
    H+          6.4738e-06  5.0116e-06  5.0116e-06  5.0116e-06  1.0000e-05  0.           0.           0.           0.
    Co3--       8.5850e-10  1.4243e-09  1.4243e-09  1.4243e-09  3.6192e-10  0.           0.           0.           0.
    neg.KI      3.8990e-02  3.8990e-02  3.8990e-02  3.8990e-02  3.8990e-02  0.           0.           0.           0.
    Ca++        1.9809e-05  3.5175e-05  3.5175e-05  3.5175e-05  1.0000e-19  0.           0.           0.           0.
    Cl-         1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  0.           0.           0.           0.
    OH-         1.7019e-09  2.2013e-09  2.2013e-09  2.2013e-09  1.0998e-09  0.           0.           0.           0.
shftsol        -2.4174e+01 -2.3927e+01 -2.3927e+01 -2.3927e+01 -2.4500e+01  0.           0.           0.           0.
shift0        -3.3236e+01 -3.2817e+01 -3.2817e+01 -3.2817e+01 -3.3778e+01  0.           0.           0.           0.
shift1        -2.3521e+01 -2.3098e+01 -2.3098e+01 -2.3098e+01 -2.4068e+01  0.           0.           0.           0.
shift2        -2.5195e+01 -2.4773e+01 -2.4773e+01 -2.4773e+01 -2.5742e+01  0.           0.           0.           0.

   complexes

    HCO3-       1.0416e-04  1.3343e-04  1.3343e-04  1.3343e-04  6.8070e-05  0.           0.           0.           0.
    H2CO3       1.4446e-03  1.4307e-03  1.4307e-03  1.4307e-03  1.4609e-03  0.           0.           0.           0.

  precipitates

    CaCO3       9.9998e-01  9.9996e-01  9.9996e-01  9.9996e-01  1.0000e+00  0.           0.           0.           0.
```

OUTPUT

```
 1    time= 8.21953e-01          delt=  1.0000e-03         concentration of species in moles/liter solution


                                                            distance
    species         0.000        1.640        3.280        4.920        6.560        0.000        0.000        0.000        0.000

   Na+          1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  0.          0.          0.          0.
   H+           3.1184e-08  3.1184e-08  3.1184e-08  3.1184e-08  1.0000e-05  0.          0.          0.          0.
   Co3--        5.3028e-06  5.3028e-06  5.3028e-06  5.3028e-06  3.6192e-10  0.          0.          0.          0.
   neg.KI       3.8990e-02  3.8990e-02  3.8990e-02  3.8990e-02  3.8990e-02  0.          0.          0.          0.
   Ca++         1.3111e-03  1.3111e-03  1.3111e-03  1.3111e-03  1.0000e-19  0.          0.          0.          0.
   Cl-          1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  1.0000e-03  0.          0.          0.          0.
   OH-          3.8071e-07  3.8071e-07  3.8071e-07  3.8071e-07  1.0998e-09  0.          0.          0.          0.
shftsol        -1.2728e+01 -1.2728e+01 -1.2728e+01 -1.2728e+01 -2.4500e+01  0.          0.          0.          0.
shift0        -1.3322e+01 -1.3322e+01 -1.3322e+01 -1.3322e+01 -3.3778e+01  0.          0.          0.          0.
shift1        -3.4062e+00 -3.4062e+00 -3.4062e+00 -3.4062e+00 -2.4068e+01  0.          0.          0.          0.
shift2        -5.1152e+00 -5.1152e+00 -5.1152e+00 -5.1152e+00 -2.5742e+01  0.          0.          0.          0.

    complexes

   HC03-        2.6693e-03  2.6693e-03  2.6693e-03  2.6693e-03  6.8070e-05  0.          0.          0.          0.
   H2C03        1.6550e-04  1.6550e-04  1.6550e-04  1.6550e-04  1.4609e-03  0.          0.          0.          0.

  precipitates

   CaC03        9.9869e-01  9.9869e-01  9.9869e-01  9.9869e-01  1.0000e+00  0.          0.          0.          0.
```

APPENDIX 5

KINETIC CALCITE PRECIPITATION DURING REACTIVE TRANSPORT

INPUT AND SAMPLE OUTPUT

Table of Chemical Reactions

1. Equilibrum chemical reaction

ion exchange

$Na^+ + \overline{H} \quad = \quad H^+ + \overline{Na}$ log K = −0.176

$\frac{1}{2} Ca^{2+} \quad + \quad \overline{H} = H^+ + \frac{1}{2} \overline{C}_a$ log K = −0.181

complexation

same as in Appendix 2

dissociation of water

same as in Appendix 2

2. Kinetic chemical reactions

Same rate rate formula as in problem 1 (in this case k_4 term will dominate and reverse reaction starts).

INPUT

```
Isothermal kinetic precip.,aqueous complex,w.dissoc.transport-chemd15t
UNITS      meter     year
 type              0         1         0
iteration        100       100       100
Calc.crtia   1.0e-30   1.0e-03        20        10         5      1.75         5
Grid              20      10.0 836.68355
time              10   1.0e-10   1.0e-03
prin time   0.000001  0.000003  0.000005  0.000008   0.00001   0.00005  0.0001
print time    0.0005     0.001      0.01
physical         1.0       0.0   1.0e-01
tran bound         0         0
temp bound         0         0
isothermal         0
intl. temp      20.0
leach              0         0
WORD      tpkin
   carbonate precip. rate equation data next three lines
1 PARAMS           5     100.0         4     prepc
2 CDATS         19.8     6.383    10.377     8.446 3.154e+04
3 INXi(i)     2    0   10    0    0    0    4    9
4 CONK(1)      -1.32         0         0         0
5 CONK(2)      -4.59         0         0         0
6 CONK(3)      -6.94         0         0         0
7 CONK(4)        0.0         0         0         0
WORD      blank
no.species         6         2         0         2         1         2
cecns          0.328
      na  1.0    0    2 2.764e-04 2.764e-04    2   1.0e-03   1.0e-03    1   0
      na    4.22e-03  1.52e-02
      na     -0.1760         0         0         0
       h  1.0         1   1.2e-10   1.2e-10    1 1.232e-10 1.232e-10    0   0
     co3 -2.0    0    31.3203e-04 3.200e-05    311.800e-05 3.209e-05    0   0
      ca  2.0    0    3 1.168e-04 1.168e-04    3 1.169e-04 1.169e-04    1   0
      ca       0.160     0.156
      ca     -0.3570         0         0         0
      cl -1.0         4 2.764e-04 2.764e-04    4   1.0e-03   1.0e-03    0   0
      oh   -13.990
      oh     1.0e-04 8.307e-05    0    0
    hco3    0  0.0  1.0  1.0  0.0  0.0  0.0
    hco3      10.337         0         0
    hco3     1.0e-04 8.589e-05    0    0
   h2co3    0  0.0  2.0  1.0  0.0  0.0  0.0
   h2co3      16.700         0         0         0
   h2co3     3.0e-08 2.441e-08    0    0
   caco3   11  0.0  0.0  1.0  1.0  0.0  0.0
   caco3      -8.446         0         0
   caco3         0.0      0.00    0    0
```

OUTPUT

```
1Isothermal kinetic precip.,aqueous complex,w.dissoc.transport-chemd15t

    unit of length is   meter            unit of time is    year

kinetics of rev. precipitate dissolution considered


    inner boundary is constant concentration          outer boundary is constant initial concentration


    dispersion coefficient= 1.0v+  0.           fluid velocity = 0.100         number of grid points =   20


    number of print outs =   10
    given at the following times= 0.1000e-050.3000e-050.5000e-050.8000e-050.1000e-040.5000e-040.1000e-030.5000e-03
                                  0.1000e-020.1000e-01

errsqr=   0.10e-29     climit=   0.10e-02      iterj=    20      iterm=    10     iterd=     5     dtmult=   0.17e+01     ndiv=

    number of sites(ion-exchange)    0.3280          number of sites(edl model)    0.0000
 this simulation is isothermal, temperature =  20.



      5      bare ions             2     sorbing ions (ionex)           0     sorbing ions (edl model)


component charge      kinetic     initial condition        boundary condition
                         code       type         value         type         value

      na     1.          0          2       2.7640e-04         2       1.0000e-03
       h     1.          0          1       1.2000e-10         1       1.2320e-10
     co3    -2.          0          3       1.3203e-04         3       1.1800e-04
      ca     2.          0          3       1.1680e-04         3       1.1690e-04
      cl    -1.          0          4       2.7640e-04         4       1.0000e-03


      3      complexes              0       sorbing complexes(ionex)            0     sorbing complexes (edl model)


complex    charge        kcode                         stoichiometry

      oh    -1.         0          0.   0.   0.   0.   0.   1.
    hco3    -1.         0          0.   1.   1.   0.   0.   0.
   h2co3     0.         0          0.   2.   1.   0.   0.   0.
```

OUTPUT

```
     1       solid precipitates

  precipitate  kinetic code          stoichiometry

   caco3          11           0.0  0.0  1.0  1.0  0.0  0.0

 Calcite dissolution information:

  no of terms in the rate formulla =     4

  no of other additional data needed =     5

  representative surface parameter =    0.100e+03

  xdiag = prepc

  additional data for rate calculation

    0.198e+02   0.638e+01   0.104e+02   0.845e+01   0.315e+05

 binary element nos. of the terms of the rate equation
               2    0   10    0    0    0    4    9

 coefficients in log function of thermodynamic constants
               -1.320      0.000      0.000      0.000
               -4.590      0.000      0.000      0.000
               -6.940      0.000      0.000      0.000
                0.000      0.000      0.000      0.000

                                                      ** spatial grid **

    0.            0.5086e-02   0.1233e-01   0.2266e-01   0.3738e-01   0.5835e-01   0.8823e-01   0.1308       0.1915       0.2780
    0.4012        0.5768       0.8271        1.184        1.692        2.416        3.448        4.918        7.014       10.00
```

OUTPUT

```
1                                                              initial conditions
                                                                 temp=    20.00


 component       total          bare ion        sorbed concentration        act. coef.    kd(ion ex)        kd(edl)
              aqueous conc.    concentration     (ionex)     (edl)

      na      0.2764e-03      0.2764e-03      0.4375e-02 0.                  0.9718          15.83          0.
       h      0.9109e-04      0.1200e-09      0.          0.                  0.9718        0.             0.
     co3      0.1302e-03      0.3915e-04      0.          0.                  0.8918        0.             0.
      ca      0.1150e-03      0.1150e-03      0.1618      0.                  0.8918          1407.          0.
      cl      0.2467e-03      0.2467e-03      0.          0.                  0.9718        0.             0.
      oh      0.9030e-04      0.9030e-04      0.          0.                  0.9718        0.             0.

 complex         valence        aqueous conc.   sorption(io)      sorption(edl)
     hco3     -1.000          0.9104e-04      0.              0.
    h2co3     0.              0.2380e-07      0.              0.

  precipitate   equivalent moles/liter solution

    caco3 0.181e-05

    to have a net balance of zero charge, other 'species must be present (although not participating in the
    chemical reactions) with a total concentration times charge of  0.

1                                   boundary or influx conditions
                                                temp=    20.00
 component       total          bare ion        sorbed concentration        act. coef.    kd(ion ex)        kd(edl)
              aqueous conc.    concentration     (ionex)     (edl)

      na      0.1000e-02      0.1000e-02      0.1587e-01 0.                  0.9600          15.87          0.
       h      0.8198e-04      0.1232e-09      0.          0.                  0.9600        0.             0.
     co3      0.1180e-03      0.3604e-04      0.          0.                  0.8492        0.             0.
      ca      0.1169e-03      0.1169e-03      0.1561      0.                  0.8492          1335.          0.
      cl      0.9896e-03      0.9896e-03      0.          0.                  0.9600        0.             0.
      oh      0.9013e-04      0.9013e-04      0.          0.                  0.9600        0.             0.

 complex         valence        aqueous conc.   sorption(io)      sorption(edl)
     hco3     -1.000          0.8193e-04      0.              0.
    h2co3     0.              0.2146e-07      0.              0.

  precipitate   equivalent moles/liter solution

    caco3 0.

    to have a net balance of zero charge, other 'species must be present (although not participating in the
    chemical reactions) with a total concentration times charge of -0.121e-16
```

OUTPUT

```
1     time= 3.60156e-06          delt=  1.3541e-06          concentration of species in moles/liter solution


                                                             distance
     species        0.000       0.005       0.012       0.023       0.037       0.058       0.088       0.131       0.192

          na     1.0000e-03  2.7966e-04  2.7641e-04  2.7640e-04  2.7640e-04  2.7640e-04  2.7640e-04  2.7640e-04  2.7640e-04
           h     1.2320e-10  1.2006e-10  1.2000e-10  1.2000e-10  1.2000e-10  1.2000e-10  1.2000e-10  1.2000e-10  1.2000e-10
         co3     3.6044e-05  3.9123e-05  3.9154e-05  3.9154e-05  3.9154e-05  3.9154e-05  3.9154e-05  3.9154e-05  3.9154e-05
          ca     1.1690e-04  1.1750e-04  1.1500e-04  1.1499e-04  1.1499e-04  1.1499e-04  1.1499e-04  1.1499e-04  1.1499e-04
          cl     9.8965e-04  2.5518e-04  2.4676e-04  2.4673e-04  2.4673e-04  2.4673e-04  2.4673e-04  2.4673e-04  2.4673e-04
          oh     9.0131e-05  9.0292e-05  9.0298e-05  9.0298e-05  9.0298e-05  9.0298e-05  9.0298e-05  9.0298e-05  9.0298e-05

  sorbed(ionex)

          na     1.5873e-02  4.3797e-03  4.3747e-03  4.3747e-03  4.3747e-03  4.3747e-03  4.3747e-03  4.3747e-03  4.3747e-03
          ca     1.5606e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01

     complexes

        hco3     8.1935e-05  9.0934e-05  9.1041e-05  9.1042e-05  9.1042e-05  9.1042e-05  9.1042e-05  9.1041e-05  9.1041e-05
       h2co3     2.1458e-08  2.3773e-08  2.3799e-08  2.3799e-08  2.3799e-08  2.3799e-08  2.3799e-08  2.3799e-08  2.3799e-08

   precipitates

       caco3     0.          1.8106e-06  1.8106e-06  1.8106e-06  1.8107e-06  1.8107e-06  1.8108e-06  1.8109e-06  1.8111e-06

                                                             distance
     species        0.278       0.401       0.577       0.827       1.184       1.692       2.416       3.448       4.918

          na     2.7640e-04  2.7640e-04  2.7640e-04  2.7640e-04  2.7640e-04  2.7640e-04  2.7640e-04  2.7639e-04  2.7639e-04
           h     1.2000e-10  1.2000e-10  1.2000e-10  1.2000e-10  1.2000e-10  1.2000e-10  1.2000e-10  1.2001e-10  1.2001e-10
         co3     3.9154e-05  3.9153e-05  3.9153e-05  3.9153e-05  3.9153e-05  3.9152e-05  3.9151e-05  3.9150e-05  3.9148e-05
          ca     1.1499e-04  1.1499e-04  1.1499e-04  1.1499e-04  1.1499e-04  1.1499e-04  1.1499e-04  1.1498e-04  1.1498e-04
          cl     2.4673e-04  2.4673e-04  2.4673e-04  2.4673e-04  2.4673e-04  2.4673e-04  2.4673e-04  2.4673e-04  2.4673e-04
          oh     9.0297e-05  9.0297e-05  9.0297e-05  9.0297e-05  9.0296e-05  9.0295e-05  9.0294e-05  9.0293e-05  9.0291e-05

  sorbed(ionex)

          na     4.3747e-03  4.3747e-03  4.3747e-03  4.3747e-03  4.3747e-03  4.3747e-03  4.3747e-03  4.3747e-03  4.3747e-03
          ca     1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01

     complexes

        hco3     9.1041e-05  9.1041e-05  9.1041e-05  9.1040e-05  9.1040e-05  9.1039e-05  9.1038e-05  9.1037e-05  9.1035e-05
       h2co3     2.3799e-08  2.3799e-08  2.3799e-08  2.3799e-08  2.3799e-08  2.3799e-08  2.3799e-08  2.3799e-08  2.3799e-08

   precipitates

       caco3     1.8113e-06  1.8116e-06  1.8121e-06  1.8127e-06  1.8137e-06  1.8150e-06  1.8169e-06  1.8195e-06  1.8233e-06
```

OUTPUT

```
                                                         distance
     species       7.014      10.000
          na   2.7639e-04  2.7640e-04
           h   1.2001e-10  1.2000e-10
         co3   3.9146e-05  3.9154e-05
          ca   1.1498e-04  1.1499e-04
          cl   2.4673e-04  2.4673e-04
          oh   9.0288e-05  9.0298e-05

   sorbed(ionex)

          na   4.3747e-03  4.3747e-03
          ca   1.6181e-01  1.6181e-01

     complexes

        hco3   9.1032e-05  9.1042e-05
       h2co3   2.3799e-08  2.3799e-08

    precipitates

       caco3   1.8287e-06  1.8105e-06

                    1

                    2

     convergence is obtained in less than 6 iterations so time step is increased by 1.75
     delt=  2.3697e-06          at time=  4.9556e-06
          2

                    1

                    2
  calculation time=     0.52040e+01seconds

     convergence is obtained in less than 6 iterations so time step is increased by 1.75
     delt=  2.3697e-06          at time=  7.3253e-06
```

OUTPUT

```
1    time= 1.01327e-02          delt=  1.0000e-03         concentration of species in moles/liter solution


                                                            distance
    species        0.000       0.005       0.012       0.023       0.037       0.058       0.088       0.131       0.192

          na   1.0000e-03  7.7995e-04  5.6810e-04  4.3485e-04  3.6853e-04  3.1981e-04  2.8809e-04  2.7763e-04  2.7609e-04
           h   1.2320e-10  1.2443e-10  1.2591e-10  1.2714e-10  1.2712e-10  1.2506e-10  1.2215e-10  1.2068e-10  1.2042e-10
         co3   3.6044e-05  3.5670e-05  3.5168e-05  3.4721e-05  3.4854e-05  3.6062e-05  3.7789e-05  3.8707e-05  3.8874e-05
          ca   1.1690e-04  1.9211e-04  2.4914e-04  2.5037e-04  2.0517e-04  1.5489e-04  1.2508e-04  1.1601e-04  1.1471e-04
          cl   9.8965e-04  9.2171e-04  8.2607e-04  6.9699e-04  5.3919e-04  3.8398e-04  2.8467e-04  2.5195e-04  2.4706e-04
          oh   9.0131e-05  8.9252e-05  8.8099e-05  8.6917e-05  8.6378e-05  8.7197e-05  8.8857e-05  8.9803e-05  8.9978e-05

 sorbed(ionex)

          na   1.5873e-02  9.6654e-03  6.1801e-03  4.7103e-03  4.3957e-03  4.3750e-03  4.3754e-03  4.3751e-03  4.3751e-03
          ca   1.5606e-01  1.5917e-01  1.6091e-01  1.6164e-01  1.6180e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01

    complexes

        hco3   8.1935e-05  8.1873e-05  8.1871e-05  8.2243e-05  8.3602e-05  8.6289e-05  8.9144e-05  9.0484e-05  9.0717e-05
       h2co3   2.1458e-08  2.1653e-08  2.1936e-08  2.2335e-08  2.2846e-08  2.3359e-08  2.3681e-08  2.3784e-08  2.3799e-08

  precipitates

       caco3   0.          8.0088e-06  1.3337e-05  1.4760e-05  1.1934e-05  7.1041e-06  3.5926e-06  2.4909e-06  2.3920e-06

                                                            distance
    species        0.278       0.401       0.577       0.827       1.184       1.692       2.416       3.448       4.918

          na   2.7597e-04  2.7595e-04  2.7594e-04  2.7594e-04  2.7594e-04  2.7594e-04  2.7594e-04  2.7594e-04  2.7594e-04
           h   1.2043e-10  1.2044e-10  1.2045e-10  1.2045e-10  1.2045e-10  1.2045e-10  1.2045e-10  1.2045e-10  1.2045e-10
         co3   3.8869e-05  3.8859e-05  3.8855e-05  3.8853e-05  3.8853e-05  3.8853e-05  3.8853e-05  3.8853e-05  3.8853e-05
          ca   1.1460e-04  1.1458e-04  1.1458e-04  1.1457e-04  1.1457e-04  1.1457e-04  1.1457e-04  1.1457e-04  1.1457e-04
          cl   2.4674e-04  2.4673e-04  2.4673e-04  2.4673e-04  2.4673e-04  2.4673e-04  2.4673e-04  2.4673e-04  2.4673e-04
          oh   8.9971e-05  8.9959e-05  8.9954e-05  8.9953e-05  8.9953e-05  8.9953e-05  8.9953e-05  8.9953e-05  8.9953e-05

 sorbed(ionex)

          na   4.3751e-03  4.3751e-03  4.3751e-03  4.3751e-03  4.3751e-03  4.3751e-03  4.3751e-03  4.3751e-03  4.3751e-03
          ca   1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01  1.6181e-01

    complexes

        hco3   9.0715e-05  9.0703e-05  9.0698e-05  9.0697e-05  9.0696e-05  9.0696e-05  9.0696e-05  9.0696e-05  9.0696e-05
       h2co3   2.3800e-08  2.3800e-08  2.3800e-08  2.3800e-08  2.3800e-08  2.3800e-08  2.3800e-08  2.3800e-08  2.3800e-08

  precipitates

       caco3   2.4245e-06  2.4460e-06  2.4543e-06  2.4565e-06  2.4568e-06  2.4568e-06  2.4568e-06  2.4568e-06  2.4568e-06
```

OUTPUT

```
                                                            distance
   species        7.014       10.000
        na    2.7594e-04   2.7640e-04
         h    1.2045e-10   1.2000e-10
       co3    3.8853e-05   3.9154e-05
        ca    1.1457e-04   1.1499e-04
        cl    2.4673e-04   2.4673e-04
        oh    8.9953e-05   9.0298e-05

sorbed(ionex)

        na    4.3751e-03   4.3747e-03
        ca    1.6181e-01   1.6181e-01

   complexes

      hco3    9.0696e-05   9.1042e-05
     h2co3    2.3800e-08   2.3799e-08

 precipitates

     caco3    2.4568e-06   1.8105e-06
```

APPENDIX 6

ISOTHERMAL SILICA WATER REACTION IN BATCH SIMULATION

INPUTS AND SAMPLE OUTPUT

1 – Kinetic Dissolution

2 – Kinetic Precipitation

INPUT

```
isothermal kinetic dissolution of SiO2--S1N
UNITS      meter     second
 type              0         1         0
iteration        100       100       100
Calc.crtia   1.0e-30   1.0e-03        20        10         5      1.75         5
grid              20      6.56       0.0
Time              10     360.0   7.2e+05
print time     360.0    3600.0   18000.0   36000.0  108000.0  144000.0  216000.0
print time  360000.0  504000.0  648000.0
physical         0.0       0.0       0.0
tran bound         0         0
Temp bound         0
isothermal         0
Intl. temp     105.0
leach              0         0
WORD      tpkin
 reversible rate constants next four lines
1 PARAM            0     265.0         2     preps
2 INXI(i)     0    0   -1    0
3 CONK(1)    -11.309
4 CONK(2)    -8.0888
WORD      blank
no.species         1         0         0         0         1         0         0
H4SiO4      0    1    3   0.1e+01   1.0e-20    3   0.1e+01   1.0e-20
Sio2        2  1.0
Sio2        -20.0000
SiO2         0.1e+01   0.1e+01
```

OUTPUT

```
1isothermal kinetic dissolution of Si02--S1N

    unit of length is   meter            unit of time is     secon

kinetics of rev. precipitate dissolution considered


    inner boundary is constant concentration          outer boundary is constant initial concentration


    dispersion coefficient= 0.0v+  0.           fluid velocity = 0.           number of grid points =   20


    number of print outs =   10
    given at the following times=  360.0     3600.    0.1800e+050.3600e+050.1080e+060.1440e+060.2160e+060.3600e+06
                                  0.5040e+060.6480e+06

errsqr=   0.10e-29    climit=   0.10e-02    iterj=    20    iterm=    10    iterd=     5    dtmult=   0.17e+01    ndiv=

    number of sites(ion-exchange)    0.0000          number of sites(edl model)    0.0000
 this simulation is isothermal, temperature = 105.



     1      bare ions             0     sorbing ions (ionex)            0     sorbing ions (edl model)


component charge      kinetic     initial condition       boundary condition
                         code       type        value         type        value

H4Si04      0.           1          3      1.0000e+00        3      1.0000e+00


      1       solid precipitates

  precipitate  kinetic code          stoichiometry

Sio2                2           1.0
```

OUTPUT

```
Silicate-water reaction information:

 no of terms in the rate formulla =      2

 no of other additional data needed =     0

 representative surface parameter =    0.265e+03

 xdiag = preps


binary element nos. of the terms of the rate equation
              0    0   -1    0

coefficients in log function of thermodynamic constants
             -11.309     0.000     0.000     0.000
              -8.089     0.000     0.000     0.000



                                                                   ** spatial grid **

0.            0.3453        0.6905        1.036         1.381         1.726         2.072         2.417         2.762         3.107
 3.453         3.798         4.143        4.488         4.834         5.179         5.524         5.869         6.215         6.560
```

OUTPUT

```
1                                                                 initial conditions
                                                                    temp=  105.00

component        total          bare ion          sorbed concentration         act. coef.     kd(ion ex)          kd(edl)
             aqueous conc.    concentration        (ionex)     (edl)

H4SiO4        0.1000e-19     0.1000e-19     0.           0.                     1.000          0.              0.

 precipitate   equivalent moles/liter solution

Sio2      1.00

   to have a net balance of zero charge, other 'species must be present (although not participating in the
   chemical reactions) with a total concentration times charge of  0.


1                                 boundary or influx conditions
                                               temp=  105.00
component        total          bare ion          sorbed concentration         act. coef.     kd(ion ex)          kd(edl)
             aqueous conc.    concentration        (ionex)     (edl)

H4SiO4        0.1000e-19     0.1000e-19     0.           0.                     1.000          0.              0.

 precipitate   equivalent moles/liter solution

Sio2      1.00

   to have a net balance of zero charge, other 'species must be present (although not participating in the
   chemical reactions) with a total concentration times charge of  0.

                      1

                      2
calculation time=      0.92900e-02seconds

   convergence is obtained in less than 6 iterations so time step is increased by 1.75
   delt=  3.6000e+02          at time=  3.6000e+02
```

OUTPUT

```
1     time= 4.38187e+03          delt=  1.9294e+03          concentration of species in moles/liter solution


                                                            distance
   species        0.000       0.345       0.691       1.036       1.381       1.726       2.072       2.417       2.762

  H4Si04       1.0000e-20  5.6657e-06  5.6657e-06  5.6657e-06  5.6657e-06  5.6657e-06  5.6657e-06  5.6657e-06  5.6657e-06

 precipitates

  Sio2         1.0000e+00  9.9999e-01  9.9999e-01  9.9999e-01  9.9999e-01  9.9999e-01  9.9999e-01  9.9999e-01  9.9999e-01


                                                            distance
   species        3.107       3.453       3.798       4.143       4.488       4.834       5.179       5.524       5.869

  H4Si04       5.6657e-06  5.6657e-06  5.6657e-06  5.6657e-06  5.6657e-06  5.6657e-06  5.6657e-06  5.6657e-06  5.6657e-06

 precipitates

  Sio2         9.9999e-01  9.9999e-01  9.9999e-01  9.9999e-01  9.9999e-01  9.9999e-01  9.9999e-01  9.9999e-01  9.9999e-01


                                                            distance
   species        6.215       6.560
  H4Si04       5.6657e-06  1.0000e-20

 precipitates

  Sio2         9.9999e-01  1.0000e+00

                  1

                  2

   convergence is obtained in less than 6 iterations so time step is increased by 1.75
   delt=  3.3764e+03          at time=  6.3113e+03
        2

                  1

                  2

   convergence is obtained in less than 6 iterations so time step is increased by 1.75
   delt=  5.9087e+03          at time=  9.6877e+03
        2
```

OUTPUT

```
1    time= 7.78704e+05          delt=  1.6972e+05          concentration of species in moles/liter solution

                                                              distance
  species        0.000       0.345       0.691       1.036       1.381       1.726       2.072       2.417       2.762

 H4SiO4       1.0000e-20  4.7309e-04  4.7309e-04  4.7309e-04  4.7309e-04  4.7309e-04  4.7309e-04  4.7309e-04  4.7309e-04

precipitates

 Sio2         1.0000e+00  9.9953e-01  9.9953e-01  9.9953e-01  9.9953e-01  9.9953e-01  9.9953e-01  9.9953e-01  9.9953e-01

                                                              distance
  species        3.107       3.453       3.798       4.143       4.488       4.834       5.179       5.524       5.869

 H4SiO4       4.7309e-04  4.7309e-04  4.7309e-04  4.7309e-04  4.7309e-04  4.7309e-04  4.7309e-04  4.7309e-04  4.7309e-04

precipitates

 Sio2         9.9953e-01  9.9953e-01  9.9953e-01  9.9953e-01  9.9953e-01  9.9953e-01  9.9953e-01  9.9953e-01  9.9953e-01

                                                              distance
  species        6.215       6.560
 H4SiO4       4.7309e-04  1.0000e-20

precipitates

 Sio2         9.9953e-01  1.0000e+00
```

INPUT

```
isothermal kinetic precipitation of SiO2--S3N
UNITS      meter     second
 type              0         1         0
iteration        100       100       100
Calc.crtia   1.0e-30   1.0e-03        20        10         5      1.75         5
grid              20      6.56       0.0
Time              10     360.0   7.2e+05
print time     360.0    3600.0   18000.0   36000.0  108000.0  144000.0  216000.0
print time  360000.0  504000.0  648000.0
physical         0.0       0.0       0.0
tran bound         0         0
Temp bound         0
isothermal         0
Intl. temp     105.0
leach              0         0
WORD      tpkin
 reversible rate constants next four lines
1 PARAM            0     261.0         2     preps
2 INXI(i)     0    0   -1    0
3 CONK(1)    -11.309
4 CONK(2)    -8.0888
WORD      blank
no.species         1         0         0         0         1         0         0
H4sio4      0    1    3   1.0e-03   1.0e-03    3   1.0e-03   1.0e-03
Sio2        2  1.0
Sio2         -3.0000
Sio2         1.0e-20   1.0e-20
```

OUTPUT

```
1isothermal kinetic precipitation of SiO2--S3N

    unit of length is   meter            unit of time is     secon

kinetics of rev. precipitate dissolution considered


    inner boundary is constant concentration          outer boundary is constant initial concentration


    dispersion coefficient= 0.0v+  0.           fluid velocity = 0.            number of grid points =   20


    number of print outs =   10
    given at the following times=  360.0     3600.    0.1800e+050.3600e+050.1080e+060.1440e+060.2160e+060.3600e+06
                                  0.5040e+060.6480e+06

errsqr=   0.10e-29     climit=   0.10e-02     iterj=    20     iterm=    10     iterd=     5     dtmult=   0.17e+01     ndiv=

    number of sites(ion-exchange)    0.0000            number of sites(edl model)    0.0000
. this simulation is isothermal, temperature = 105.



     1      bare ions             0     sorbing ions (ionex)            0     sorbing ions (edl model)


component charge      kinetic     initial condition       boundary condition
                        code        type        value         type        value

H4sio4      0.          1           3      1.0000e-03        3      1.0000e-03



      1       solid precipitates

  precipitate  kinetic code          stoichiometry

Sio2               2            1.0
```

OUTPUT

```
Silicate-water reaction information:

 no of terms in the rate formulla =     2

 no of other additional data needed =     0

 representative surface parameter =    0.261e+03

 xdiag = preps


binary element nos. of the terms of the rate equation
             0    0   -1    0

coefficients in log function of thermodynamic constants
            -11.309     0.000     0.000     0.000
             -8.089     0.000     0.000     0.000



                                                   ** spatial grid **

0.          0.3453      0.6905      1.036       1.381       1.726       2.072       2.417       2.762       3.107
 3.453       3.798       4.143       4.488       4.834       5.179       5.524       5.869       6.215       6.560
```

OUTPUT

```
1                                                        initial conditions
                                                          temp=  105.00

 component       total        bare ion      sorbed concentration       act. coef.    kd(ion ex)       kd(edl)
              aqueous conc.   concentration    (ionex)    (edl)

 H4sio4      0.1000e-02    0.1000e-02    0.         0.                 1.000        0.             0.

  precipitate   equivalent moles/liter solution

 Sio2     0.100e-19

    to have a net balance of zero charge, other 'species must be present (although not participating in the
    chemical reactions) with a total concentration times charge of  0.


1                                 boundary or influx conditions
                                           temp=  105.00
 component       total        bare ion      sorbed concentration       act. coef.    kd(ion ex)       kd(edl)
              aqueous conc.   concentration    (ionex)    (edl)

 H4sio4      0.1000e-02    0.1000e-02    0.         0.                 1.000        0.             0.

  precipitate   equivalent moles/liter solution

 Sio2     0.100e-19

    to have a net balance of zero charge, other 'species must be present (although not participating in the
    chemical reactions) with a total concentration times charge of  0.

                    1

                    2
calculation time=     0.91630e-02seconds

    convergence is obtained in less than 6 iterations so time step is increased by 1.75
    delt=  3.6000e+02           at time=  3.6000e+02
```

OUTPUT

```
1    time= 7.78704e+05        delt=  1.6972e+05         concentration of species in moles/liter solution

                                                     distance
  species       0.000       0.345       0.691       1.036       1.381       1.726       2.072       2.417       2.762

 H4sio4      1.0000e-03  6.8944e-04  6.8944e-04  6.8944e-04  6.8944e-04  6.8944e-04  6.8944e-04  6.8944e-04  6.8944e-04
precipitates
 Sio2        1.0000e-20  3.1056e-04  3.1056e-04  3.1056e-04  3.1056e-04  3.1056e-04  3.1056e-04  3.1056e-04  3.1056e-04

                                                     distance
  species       3.107       3.453       3.798       4.143       4.488       4.834       5.179       5.524       5.869

 H4sio4      6.8944e-04  6.8944e-04  6.8944e-04  6.8944e-04  6.8944e-04  6.8944e-04  6.8944e-04  6.8944e-04  6.8944e-04
precipitates
 Sio2        3.1056e-04  3.1056e-04  3.1056e-04  3.1056e-04  3.1056e-04  3.1056e-04  3.1056e-04  3.1056e-04  3.1056e-04

                                                     distance
  species       6.215       6.560
 H4sio4      6.8944e-04  1.0000e-03
precipitates
 Sio2        3.1056e-04  1.0000e-20
```

OUTPUT

```
1    time= 4.38187e+03          delt=  1.9294e+03          concentration of species in moles/liter solution


                                                           distance
  species        0.000       0.345       0.691       1.036       1.381       1.726       2.072       2.417       2.762

 H4sio4      1.0000e-03  9.9631e-04  9.9631e-04  9.9631e-04  9.9631e-04  9.9631e-04  9.9631e-04  9.9631e-04  9.9631e-04

precipitates

 Sio2        1.0000e-20  3.6852e-06  3.6852e-06  3.6852e-06  3.6852e-06  3.6852e-06  3.6852e-06  3.6852e-06  3.6852e-06

                                                           distance
  species        3.107       3.453       3.798       4.143       4.488       4.834       5.179       5.524       5.869

 H4sio4      9.9631e-04  9.9631e-04  9.9631e-04  9.9631e-04  9.9631e-04  9.9631e-04  9.9631e-04  9.9631e-04  9.9631e-04

precipitates

 Sio2        3.6852e-06  3.6852e-06  3.6852e-06  3.6852e-06  3.6852e-06  3.6852e-06  3.6852e-06  3.6852e-06  3.6852e-06

                                                           distance
  species        6.215       6.560
 H4sio4      9.9631e-04  1.0000e-03

precipitates

 Sio2        3.6852e-06  1.0000e-20
```

APPENDIX 7

KINETIC DISSOLUTION OF GLASS IN BATCH SIMULATION INPUT AND SAMPLE OUTPUT

INPUT

```
Kinetic imperical glass dissolution-batch case       setglss2
UNITS      meter    second
 type              0         1         0
iteration        100       100       100
Calc.crtia   1.0e-30   1.0e-03        20        10         5      1.75         5
 grid              5      6.56       0.0
 time              7  1.50e+00  1.00e+06
print time  1.58e+04  4.93e+05  7.48e+06  5.36e+07  1.66e+08  2.82e+08  3.52e+08
Physical     1.0e-30   1.0e-30   1.0e-30
boundary           1         0
temp bound         0
Isothermal         0
Intl. temp      20.0
leach              0         0
WORD      exkin
  glass rate equation data next line
1 PARAMS           0   3.0e+06         0     dissg
WORD      blank
no.species         6         0         0         1         0         0
      na  1.0    1    1   1.0e-10   1.0e-10    1   1.0e-10   1.0e-10
       h  1.0    2    1  3.16e-05  3.16e-05    1  3.16e-05  3.16e-05    0    0
       k  1.0    3    1   1.0e-10   1.0e-10    1   1.0e-10   1.0e-10
      ca +2.0    4    1   1.0e-10   1.0e-10    1   1.0e-10   1.0e-10
      mg +2.0    5    1   1.0e-10   1.0e-10    1   1.0e-10   1.0e-10
    hco3 -1.0    0    2  2.16e-03  2.16e-03    2  2.16e-03  2.16e-03    0    0
h2co3       0  0.0  1.0  0.0  0.0  0.0  1.0
h2co3        +6.3655       0.0       0.0       0.0
h2co3       2.13e-03  2.13e-03
```

OUTPUT

```
1Kinetic imperical glass dissolution-batch case       setglss2

    unit of length is   meter            unit of time is    second

kinetics of non-rev. silica dissolution considered


    inner boundary is constant flux          outer' boundry is constant initial concentration


    dispersion coefficient= 0.0v+  0.10e-29     fluid velocity = 0.100e-29     number of grid points =    5


    number of print outs =    7
    given at the following times= 0.1580e+050.4930e+060.7480e+070.5360e+080.1660e+090.2820e+090.3520e+09

errsqr=  0.10e-29     climit=   0.10e-02     iterj=    20     iterm=    10     iterd=     5     dtmult=   0.17e+01     ndiv=

    number of sites(ion-exchange)    0.0000           number of sites(edl model)    0.0000
  this simulation is isothermal, temperature =  20.



     6      bare ions             0     sorbing ions (ionex)            0     sorbing ions (edl model)


component charge      kinetic     initial condition        boundary condition
                         code       type        value          type        value

      na     1.         1           1      1.0000e-10        1       1.0000e-10
       h     1.         2           1      3.1600e-05        1       3.1600e-05
       k     1.         3           1      1.0000e-10        1       1.0000e-10
      ca     2.         4           1      1.0000e-10        1       1.0000e-10
      mg     2.         5           1      1.0000e-10        1       1.0000e-10
    hco3    -1.         0           2      2.1600e-03        2       2.1600e-03


     1     complexes             0       sorbing complexes(ionex)            0     sorbing complexes (edl model)


complex   charge       kcode                         stoichiometry

h2co3      0.          0         0.   1.   0.   0.   0.   1.
```

OUTPUT

```
Glass dissolution information:

 no of terms in the rate formulla =     0

 no of other additional data needed =     0

 representative surface parameter =    0.300e+07

 xdiag = dissg

                                                              ** spatial grid **


0.            1.640        3.280        4.920        6.560
```

OUTPUT

```
1                                                                   initial conditions
                                                                     temp=   20.00


 component        total          bare ion         sorbed concentration         act. coef.     kd(ion ex)        kd(edl)
               aqueous conc.    concentration      (ionex)     (edl)

       na      0.1000e-09      0.1000e-09     0.          0.                   0.9936         0.               0.
        h      0.2162e-02      0.3160e-04     0.          0.                   0.9936         0.               0.
        k      0.1000e-09      0.1000e-09     0.          0.                   0.9936         0.               0.
       ca      0.1000e-09      0.1000e-09     0.          0.                   0.9746         0.               0.
       mg      0.1000e-09      0.1000e-09     0.          0.                   0.9746         0.               0.
     hco3      0.2160e-02      0.2944e-04     0.          0.                   0.9936         0.               0.

 complex          valence          aqueous conc.   sorption(io)      sorption(edl)
 h2co3         0.                  0.2131e-02     0.              0.

    to have a net balance of zero charge, other 'species must be present (although not participating in the
    chemical reactions) with a total concentration times charge of -0.216e-05
```

```
1                                boundary or influx conditions
                                               temp=   20.00
 component        total          bare ion         sorbed concentration         act. coef.     kd(ion ex)        kd(edl)
               aqueous conc.    concentration      (ionex)     (edl)

       na      0.1000e-09      0.1000e-09     0.          0.                   0.9936         0.               0.
        h      0.2162e-02      0.3160e-04     0.          0.                   0.9936         0.               0.
        k      0.1000e-09      0.1000e-09     0.          0.                   0.9936         0.               0.
       ca      0.1000e-09      0.1000e-09     0.          0.                   0.9746         0.               0.
       mg      0.1000e-09      0.1000e-09     0.          0.                   0.9746         0.               0.
     hco3      0.2160e-02      0.2944e-04     0.          0.                   0.9936         0.               0.

 complex          valence          aqueous conc.   sorption(io)      sorption(edl)
 h2co3         0.                  0.2131e-02     0.              0.

    to have a net balance of zero charge, other 'species must be present (although not participating in the
    chemical reactions) with a total concentration times charge of -0.216e-05
```

OUTPUT

```
1    time= 2.70798e+04          delt=  1.1606e+04          concentration of species in moles/liter solution


                                                     distance
  species       0.000       1.640       3.280       4.920       6.560       0.000       0.000       0.000       0.000

       na    1.1081e-05  1.1081e-05  1.1081e-05  1.1081e-05  1.0000e-10  0.          0.          0.          0.
        h    1.3839e-05  1.3839e-05  1.3839e-05  1.3839e-05  3.1600e-05  0.          0.          0.          0.
        k    1.2447e-06  1.2447e-06  1.2447e-06  1.2447e-06  1.0000e-10  0.          0.          0.          0.
       ca    2.7805e-05  2.7805e-05  2.7805e-05  2.7805e-05  1.0000e-10  0.          0.          0.          0.
       mg    1.5245e-05  1.5245e-05  1.5245e-05  1.5245e-05  1.0000e-10  0.          0.          0.          0.
     hco3    6.7051e-05  6.7051e-05  6.7051e-05  6.7051e-05  2.9437e-05  0.          0.          0.          0.

  complexes

 h2co3       2.0929e-03  2.0929e-03  2.0929e-03  2.0929e-03  2.1306e-03  0.          0.          0.          0.


                                                     distance
  species       0.000       0.000       0.000       0.000       0.000
       na    0.          0.          0.          0.          0.
        h    0.          0.          0.          0.          0.
        k    0.          0.          0.          0.          0.
       ca    0.          0.          0.          0.          0.
       mg    0.          0.          0.          0.          0.
     hco3    0.          0.          0.          0.          0.

  complexes

 h2co3       0.          0.          0.          0.          0.
```

OUTPUT

```
1    time= 3.52706e+08          delt=  1.0000e+06          concentration of species in moles/liter solution


                                                           distance
    species        0.000       1.640       3.280       4.920       6.560       0.000       0.000       0.000       0.000

         na    6.8549e-04  6.8549e-04  6.8549e-04  6.8549e-04  1.0000e-10  0.          0.          0.          0.
          h    1.7803e-07  1.7803e-07  1.7803e-07  1.7803e-07  3.1600e-05  0.          0.          0.          0.
          k    1.5982e-04  1.5982e-04  1.5982e-04  1.5982e-04  1.0000e-10  0.          0.          0.          0.
         ca    3.9455e-04  3.9455e-04  3.9455e-04  3.9455e-04  1.0000e-10  0.          0.          0.          0.
         mg    3.4236e-04  3.4236e-04  3.4236e-04  3.4236e-04  1.0000e-10  0.          0.          0.          0.
       hco3    1.5802e-03  1.5802e-03  1.5802e-03  1.5802e-03  2.9437e-05  0.          0.          0.          0.

    complexes

   h2co3       5.7977e-04  5.7977e-04  5.7977e-04  5.7977e-04  2.1306e-03  0.          0.          0.          0.

                                                           distance
    species        0.000       0.000       0.000       0.000       0.000
         na    0.          0.          0.          0.          0.
          h    0.          0.          0.          0.          0.
          k    0.          0.          0.          0.          0.
         ca    0.          0.          0.          0.          0.
         mg    0.          0.          0.          0.          0.
       hco3    0.          0.          0.          0.          0.

    complexes

   h2co3       0.          0.          0.          0.          0.
```

APPENDIX 8

EQUILIBRIUM TRANSPORT CONSIDERING REDOX

INPUT AND SAMPLE OUTPUT

INPUT

```
Equilibrium redox pH=4/EH=.3/Pco2=e-02/150C,Into water/pH=10/EH=-.3/50C
UNITS      meter    second
 type              0         1         0
iteration        100       100       100
Calc.crtia   1.0e-30   1.0e-03        20        10         5      1.75         5
grid              41      10.0    57.665
time              11       1.0       5.0
print time   1.0e+03   1.2e+03   1.4e+03   1.6e+03   1.8e+03   2.0e+03   2.2e+03
print time   2.4e+03   2.6e+03   2.8e+03
physical        0.05   1.0e-09   1.0e-03
tran bound         1         0
temp bound         1         0
nonisothml         1
Intl&Bound      50.0     150.0   8.4e+05    8.8+05       1.0
leach              0         0
WORD      redox
R-F consts    8.3143   96487.0
WORD      blank
no.species         5         0         0         7         2         3         4
UO2++     2.0    0    3   00.0000   00.0000    3   2.0e-06   5.0e-07
CO3--    -2.0    0    3   00.0000   00.0000    1  5.75e-14  5.75e-14
H+        1.0    0    1   1.0e-10   1.0e-10    1   1.0e-04   1.0e-04
E-       -1.0    0    1  4.78e+04  4.78e+04    1  2.67e-04  2.67e-04
OH-       -80.3723-1.395e+03   32.5514-3.181e-02
OH-       1.4956e-065.4758e-04
HCO3        0  0.0  1.0  1.0  0.0  0.0
HCO3-        -77.105   4593.31    29.0425.9438e-04
HCO3-        00.0000   1.1e-07
H2CO3       0  0.0  1.0  2.0  0.0  0.0
H2CO3        -40.366   5576.57    11.3313.4655e-02
H2CO3        00.0000   1.0e-04
UO2OH+      0  1.0  0.0  0.0  0.0  1.0
UO2OH+        83.003  -1107.24   -32.4563.0666e-02
UO2OH+       00.0000   1.5e-06
UO2CO3      0  1.0  1.0  0.0  0.0  0.0
UO2CO3      -275.356   13160.7    97.666-1.221e-03
UO2CO3       00.0000   1.6e-08
UO2CO3)2    0  1.0  2.0  0.0  0.0  0.0
             -59.815   2717.83    27.543-1.091e-03
             00.0000   5.0e-15
UO2+        0  1.0  0.0  0.0  1.0  0.0
            -356.197   13420.1    133.24-5.501e-02
             00.0000   1.0e-10
U(OH)5-     0  1.0  0.0  0.0  2.0  1.0
              -406.5   18088.5    152.77-7.661e-02
             00.0000   00.0000
UO2OH)2C    0  1.0  0.0  0.0  0.0  2.0
            -150.747   -72.008    58.326-5.563e-02
             00.0000   00.0000
UO2(C)      0  1.0  0.0  0.0  2.0  0.0
             644.664  -26101.3   -243.741.0999e-01
             00.0000   00.0000
```

OUTPUT

```
1Equilibrium redox pH=4/EH=.3/Pco2=e-02/150C,Into water/pH=10/EH=-.3/50C

   unit of length is   meter           unit of time is    second

redox is considered


   inner boundary is constant flux          outer' boundry is constant initial concentration


   dispersion coefficient= 0.1v+  0.10e-08     fluid velocity = 0.100e-02     number of grid points =   41


   number of print outs =   11
   given at the following times=  1000.     1200.     1400.     1600.     1800.     2000.     2200.     2400.
                                  2600.     2800.     0.

errsqr=   0.10e-29     climit=   0.10e-02     iterj=    20     iterm=    10     iterd=     5     dtmult=   0.17e+01     ndiv=

   number of sites(ion-exchange)    0.0000          number of sites(edl model)    0.0000
 simulation of mixing with heat transport, with initial temperature =  50.
   heat capacities of fluid and solid are  0.84e+06  and  0.88e+06
   thermal conductivity of solid   1.0
 inner boundary condition is constant flux of fluid with temperature = 150.
 temperature is held at initial value at outer boundary



    4      bare ions              0     sorbing ions (ionex)             0     sorbing ions (edl model)


component charge      kinetic     initial condition        boundary condition
                        code        type         value         type         value

U02++        2.          0          3      0.                   3       2.0000e-06
C03--       -2.          0          3      0.                   1       5.7500e-14
H+           1.          0          1      1.0000e-10           1       1.0000e-04
E-          -1.          0          1      4.7800e+04           1       2.6700e-04
```

OUTPUT

```
      8      complexes            0       sorbing complexes(ionex)            0      sorbing complexes (edl model)

 complex   charge       kcode                             stoichiometry

 OH-         -1.         0        0.    0.    0.    0.    1.
 HCO3        -1.         0        0.    1.    1.    0.    0.
 H2CO3        0.         0        0.    1.    2.    0.    0.
 UO2OH+       1.         0        1.    0.    0.    0.    1.
 UO2CO3       0.         0        1.    1.    0.    0.    0.
 UO2CO3)2    -2.         0        1.    2.    0.    0.    0.
 UO2+         1.         0        1.    0.    0.    1.    0.
 U(OH)5-     -1.         0        1.    0.    0.    2.    1.


      2      solid precipitates


 precipitate  kinetic code         stoichiometry

UO2OH)2C          0          1.0   0.0   0.0   0.0   2.0
UO2(C)            0          1.0   0.0   0.0   2.0   0.0

redox constants
gas constant R=    0.831e+01          Faraday constant F=    0.965e+05
                                                       ** spatial grid **

    0.          0.1883e-01   0.3966e-01   0.6272e-01   0.8824e-01   0.1165      0.1477      0.1823      0.2206      0.2630
    0.3098      0.3617       0.4191       0.4827       0.5530       0.6308      0.7169      0.8122      0.9177       1.034
     1.164       1.307        1.465        1.640        1.834        2.048       2.285       2.548       2.839       3.160
     3.516       3.910        4.346        4.829        5.363        5.954       6.608       7.332       8.133       9.019
    10.00
```

OUTPUT

```
1                                                              initial conditions
                                                                 temp=   50.00

component        total           bare ion         sorbed concentration         act. coef.     kd(ion ex)          kd(edl)
              aqueous conc.    concentration      (ionex)      (edl)

U02++         0.1183e-34      0.1000e-37     0.          0.                    0.9007        0.              0.
C03--         0.1565e-43      0.1565e-43     0.          0.                    0.9007        0.              0.
H+            0.1000e-09      0.1000e-09     0.          0.                    0.9742        0.              0.
E-            0.4780e+05      0.4780e+05     0.          0.                     1.000        0.              0.
OH-           0.5480e-03      0.5480e-03     0.          0.                    0.9742        0.              0.

complex         valence         aqueous conc.   sorption(io)      sorption(edl)
HC03          -1.000          0.             0.             0.
H2C03         0.              0.             0.             0.
U02OH+         1.000          0.5693e-44     0.             0.
U02C03        0.              0.             0.             0.
U02C03)2      -2.000          0.             0.             0.
U02+           1.000          0.             0.             0.
U(OH)5-       -1.000          0.1182e-34     0.             0.


 precipitate   equivalent moles/liter solution

U02OH)2C 0.
U02(C)   0.

   to have a net balance of zero charge, other 'species must be present (although not participating in the
   chemical reactions) with a total concentration times charge of  0.478e+05
```

OUTPUT

```
1                                boundary or influx conditions
                                            temp=  150.00
 component          total          bare ion          sorbed concentration           act. coef.     kd(ion ex)          kd(edl)
                 aqueous conc.    concentration       (ionex)     (edl)

 U02++           0.2000e-05     0.4942e-06      0.          0.                       0.9549         0.               0.
 C03--           0.9289e-04     0.5750e-13      0.          0.                       0.9549         0.               0.
 H+              0.2856e-03     0.1000e-03      0.          0.                       0.9885         0.               0.
 E-              0.2670e-03     0.2670e-03      0.          0.                        1.000         0.               0.
 0H-             0.1495e-05     0.2381e-07      0.          0.                       0.9885         0.               0.

 complex           valence         aqueous conc.   sorption(io)       sorption(edl)
 HC03            -1.000         0.1047e-06      0.              0.
 H2C03           0.             0.9277e-04      0.              0.
 U020H+           1.000         0.1471e-05      0.              0.
 U02C03          0.             0.1451e-07      0.              0.
 U02C03)2        -2.000         0.4581e-14      0.              0.
 U02+             1.000         0.1994e-07      0.              0.
 U(0H)5-         -1.000         0.9613e-16      0.              0.


  precipitate   equivalent moles/liter solution

 U020H)2C 0.
 U02(C)   0.


    to have a net balance of zero charge, other 'species must be present (although not participating in the
    chemical reactions) with a total concentration times charge of  0.165e-03
```

OUTPUT

```
1    time= 1.00256e+03          delt=  5.0000e+00         concentration of species in moles/liter solution


                                                            distance
  species        0.000       0.019       0.040       0.063       0.088       0.116       0.148       0.182       0.221

 UO2++         2.2515e-13  1.1054e-13  5.3930e-14  2.5946e-14  1.2230e-14  5.6152e-15  2.4973e-15  1.0697e-15  4.3836e-16
 CO3--         6.0703e-14  6.1341e-14  6.2243e-14  6.3536e-14  6.5414e-14  6.8178e-14  7.2315e-14  7.8621e-14  8.8451e-14
 H+            9.8367e-05  9.8310e-05  9.8228e-05  9.8109e-05  9.7932e-05  9.7667e-05  9.7263e-05  9.6638e-05  9.5657e-05
 E-            1.0053e+01  1.4322e+01  2.0453e+01  2.9383e+01  4.2581e+01  6.2381e+01  9.2542e+01  1.3919e+02  2.1241e+02
 OH-           2.3845e-08  2.3717e-08  2.3540e-08  2.3293e-08  2.2946e-08  2.2459e-08  2.1778e-08  2.0832e-08  1.9543e-08
 E-           -8.3927e-02 -9.6735e-02 -1.0959e-01 -1.2259e-01 -1.3579e-01 -1.4923e-01 -1.6288e-01 -1.7664e-01 -1.9035e-01
 tempx         1.4934e+02  1.4907e+02  1.4869e+02  1.4816e+02  1.4740e+02  1.4633e+02  1.4480e+02  1.4262e+02  1.3954e+02

  complexes

 HCO3          1.0803e-07  1.0879e-07  1.0985e-07  1.1135e-07  1.1352e-07  1.1668e-07  1.2131e-07  1.2819e-07  1.3853e-07
 H2CO3         9.2760e-05  9.2751e-05  9.2738e-05  9.2719e-05  9.2692e-05  9.2650e-05  9.2587e-05  9.2489e-05  9.2337e-05
 UO2OH+        6.6850e-13  3.2575e-13  1.5727e-13  7.4552e-14  3.4413e-14  1.5337e-14  6.5377e-15  2.6362e-15  9.9172e-16
 UO2CO3        6.7295e-15  3.2861e-15  1.5912e-15  7.5752e-16  3.5183e-16  1.5822e-16  6.8355e-17  2.8121e-17  1.0907e-17
 UO2CO3)2      2.2885e-21  1.1387e-21  5.6610e-22  2.7965e-22  1.3685e-22  6.6269e-23  3.1794e-23  1.5165e-23  7.2310e-24
 UO2+          3.3884e-10  2.3600e-10  1.6347e-10  1.1205e-10  7.5653e-11  5.0054e-11  3.2263e-11  2.0114e-11  1.2017e-11
 U(OH)5-       6.4257e-14  6.4525e-14  6.4902e-14  6.5435e-14  6.6194e-14  6.7282e-14  6.8847e-14  7.1099e-14  7.4330e-14

precipitates

 UO2OH)2C      0.          0.          0.          0.          0.          0.          0.          0.          0.
 UO2(C)        2.5305e-04  3.1980e-09  1.8956e-09  1.2167e-09  8.4168e-10  5.8621e-10  4.3575e-10  3.2533e-10  2.5851e-10


                                                            distance
  species        0.263       0.310       0.362       0.419       0.483       0.553       0.631       0.717       0.812

 UO2++         1.7048e-16  6.2240e-17  2.1020e-17  6.4442e-18  1.7600e-18  4.2654e-19  9.5181e-20  2.1968e-20  6.1494e-22
 CO3--         1.0419e-13  1.3022e-13  1.7510e-13  2.5728e-13  4.2622e-13  8.9254e-13  4.2583e-12  1.5022e-09  9.7450e-07
 H+            9.4095e-05  9.1579e-05  8.7484e-05  8.0786e-05  6.9837e-05  5.2119e-05  2.4388e-05  1.0844e-06  4.9208e-09
 E-            3.2895e+02  5.1671e+02  8.2212e+02  1.3217e+03  2.1388e+03  3.4638e+03  5.5695e+03  8.7950e+03  1.3452e+04
 OH-           1.7834e-08  1.5674e-08  1.3128e-08  1.0435e-08  8.0215e-09  6.5197e-09  8.0654e-09  1.0942e-07  1.6514e-05
 E-           -2.0374e-01 -2.1646e-01 -2.2806e-01 -2.3812e-01 -2.4638e-01 -2.5300e-01 -2.5863e-01 -2.6426e-01 -2.7061e-01
 tempx         1.3525e+02  1.2938e+02  1.2161e+02  1.1179e+02  1.0014e+02  8.7496e+01  7.5232e+01  6.4903e+01  5.7547e+01

  complexes

 HCO3          1.5425e-07  1.7844e-07  2.1617e-07  2.7625e-07  3.7703e-07  5.7779e-07  1.3175e-06  2.1523e-05  6.5063e-05
 H2CO3         9.2094e-05  9.1705e-05  9.1074e-05  9.0043e-05  8.8355e-05  8.5579e-05  8.0747e-05  5.4273e-05  7.0898e-07
 UO2OH+        3.4214e-16  1.0600e-16  2.8817e-17  6.7538e-18  1.3774e-18  2.6895e-19  7.5623e-20  2.4283e-19  1.0325e-18
 UO2CO3        3.9436e-18  1.3143e-18  4.0084e-19  1.1271e-19  3.0748e-20  9.7171e-21  7.1749e-21  4.5497e-19  6.9359e-18
 UO2CO3)2      3.4706e-24  1.6878e-24  8.3592e-25  4.2593e-25  2.3558e-25  1.8190e-25  6.9102e-25  1.5426e-20  1.4642e-16
 UO2+          6.8016e-12  3.5949e-12  1.7458e-12  7.6740e-13  3.0335e-13  1.0950e-13  3.8040e-14  1.3994e-14  6.1025e-16
 U(OH)5-       7.8916e-14  8.5305e-14  9.3958e-14  1.0542e-13  1.2140e-13  1.5171e-13  2.8105e-13  5.2815e-12  1.0068e-10
```

OUTPUT

```
precipitates

 U02OH)2C     0.          0.          0.          0.          0.          0.          0.          0.          0.
 U02(C)       2.0104e-10  1.6543e-10  1.3386e-10  1.1372e-10  9.5677e-11  8.1888e-11  6.9937e-11  5.5445e-11  0.


                                                      distance
  species        0.918       1.034       1.164       1.307       1.465       1.640       1.834       2.048       2.285


 U02++        6.7171e-23  2.0801e-23  8.4004e-24  3.7960e-24  1.7375e-24  7.4170e-25  2.7604e-25  8.5634e-26  2.1677e-26
 C03--        5.8344e-06  8.6313e-06  8.0121e-06  5.4209e-06  2.8232e-06  1.1618e-06  3.8399e-07  1.0308e-07  2.2643e-08
 H+           6.0656e-10  2.6830e-10  1.6922e-10  1.2963e-10  1.1208e-10  1.0442e-10  1.0139e-10  1.0037e-10  1.0008e-10
 E-           1.9609e+04  2.6816e+04  3.4033e+04  4.0043e+04  4.4121e+04  4.6351e+04  4.7329e+04  4.7674e+04  4.7772e+04
 OH-          1.0622e-04  2.1483e-04  3.2705e-04  4.2257e-04  4.8823e-04  5.2438e-04  5.4029e-04  5.4591e-04  5.4751e-04
 E-          -2.7765e-01 -2.8461e-01 -2.9054e-01 -2.9484e-01 -2.9748e-01 -2.9884e-01 -2.9942e-01 -2.9962e-01 -2.9968e-01
 tempx        5.3212e+01  5.1133e+01  5.0328e+01  5.0077e+01  5.0015e+01  5.0002e+01  5.0000e+01  5.0000e+01  5.0000e+01

  complexes

 HC03         4.8883e-05  3.2128e-05  1.8734e-05  9.6513e-06  4.3260e-06  1.6543e-06  5.3029e-07  1.4085e-07  3.0849e-08
 H2C03        6.4102e-08  1.8397e-08  6.7140e-09  2.6370e-09  1.0192e-09  3.6260e-10  1.1280e-10  2.9651e-11  6.4753e-12
 U02OH+       7.2691e-19  4.5324e-19  2.7652e-19  1.6030e-19  8.4360e-20  3.8574e-20  1.4775e-20  4.6293e-21  1.1751e-21
 U02C03       4.1255e-18  1.7925e-18  6.5094e-19  1.9520e-19  4.6021e-20  8.0395e-21  9.8657e-22  8.2089e-23  4.5636e-24
 U02C03)2     4.9996e-16  3.1348e-16  1.0457e-16  2.1145e-17  2.5941e-18  1.8646e-19  7.5623e-21  1.6891e-22  2.0627e-24
 U02+         9.9019e-17  4.2255e-17  2.1635e-17  1.1459e-17  5.7602e-18  2.5782e-18  9.7894e-19  3.0581e-19  7.7563e-20
 U(OH)5-      2.2513e-10  3.2004e-10  3.3990e-10  2.7946e-10  1.7964e-10  9.0765e-11  3.6256e-11  1.1526e-11  2.9379e-12

precipitates

 U02OH)2C     0.          0.          0.          0.          0.          0.          0.          0.          0.
 U02(C)       0.          0.          0.          0.          0.          0.          0.          0.          0.


                                                      distance
  species        2.548       2.839       3.160       3.516       3.910       4.346       4.829       5.363       5.954

 U02++        4.4562e-27  7.4653e-28  1.0253e-28  1.1621e-29  1.0937e-30  8.5957e-32  5.6720e-33  3.1577e-34  1.4898e-35
 C03--        4.0924e-09  6.1126e-10  7.5788e-11  7.8350e-12  6.7841e-13  4.9415e-14  3.0407e-15  1.5870e-16  7.0519e-18
 H+           1.0001e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10
 E-           4.7795e+04  4.7799e+04  4.7800e+04  4.7800e+04  4.7800e+04  4.7800e+04  4.7800e+04  4.7800e+04  4.7800e+04
 OH-          5.4787e-04  5.4794e-04  5.4795e-04  5.4795e-04  5.4796e-04  5.4796e-04  5.4796e-04  5.4796e-04  5.4796e-04
 E-          -2.9969e-01 -2.9970e-01 -2.9970e-01 -2.9970e-01 -2.9970e-01 -2.9970e-01 -2.9970e-01 -2.9970e-01 -2.9970e-01
 tempx        5.0000e+01  5.0000e+01  5.0000e+01  5.0000e+01  5.0000e+01  5.0000e+01  5.0000e+01  5.0000e+01  5.0000e+01

  complexes

 HC03         5.5717e-09  8.3211e-10  1.0317e-10  1.0666e-11  9.2350e-13  6.7267e-14  4.1392e-15  2.1604e-16  9.5995e-18
 H2C03        1.1687e-12  1.7452e-13  2.1638e-14  2.2369e-15  1.9368e-16  1.4108e-17  8.6812e-19  4.5309e-20  2.0133e-21
 U02OH+       2.4173e-22  4.0501e-23  5.5628e-24  6.3049e-25  5.9336e-26  4.6635e-27  3.0772e-28  1.7131e-29  8.0826e-31
 U02C03       1.6955e-25  4.2426e-27  7.2248e-29  8.4654e-31  6.8981e-33  3.9490e-35  1.6035e-37  4.6591e-40  9.7675e-43
 U02C03)2     1.3851e-26  5.1768e-29  1.0930e-31  1.3240e-34  9.3417e-38  3.8954e-41  9.7329e-45  1.4760e-48  1.3750e-52
 U02+         1.5953e-20  2.6727e-21  3.6709e-22  4.1606e-23  3.9156e-24  3.0774e-25  2.0307e-26  1.1305e-27  5.3337e-29
 U(OH)5-      6.0493e-13  1.0137e-13  1.3924e-14  1.5781e-15  1.4852e-16  1.1673e-17  7.7023e-19  4.2880e-20  2.0231e-21
```

OUTPUT

```
precipitates

 UO2OH)2C     0.          0.          0.          0.          0.          0.          0.          0.          0.
 UO2(C)       0.          0.          0.          0.          0.          0.          0.          0.          0.


                                                            distance
  species       6.608       7.332       8.133       9.019      10.000
 UO2++        5.9811e-37  2.0511e-38  6.0284e-40  1.5261e-41 1.0000-1363
 CO3--        2.6772e-19  8.7119e-21  2.4375e-22  5.8824e-24 1.5653-1369
 H+           1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10
 E-           4.7800e+04  4.7800e+04  4.7800e+04  4.7800e+04  4.7800e+04
 OH-          5.4796e-04  5.4796e-04  5.4796e-04  5.4796e-04  5.4796e-04
 E-          -2.9970e-01 -2.9970e-01 -2.9970e-01 -2.9970e-01 -2.9970e-01


tempx         5.0000e+01  5.0000e+01  5.0000e+01  5.0000e+01  5.0000e+01

 complexes

HCO3          3.6443e-19  1.1859e-20  3.3181e-22  8.0075e-24  0.
H2CO3         7.6433e-23  2.4872e-24  6.9590e-26  1.6794e-27  0.
UO2OH+        3.2450e-32  1.1128e-33  3.2706e-35  8.2794e-37 5.6928-1370
UO2CO3        1.4887e-45  1.6613e-48  1.3662e-51  8.3460e-55  0.
UO2CO3)2      7.9559e-57  2.8891e-61  6.6473e-66  9.7458e-71  0.
UO2+          2.1414e-30  7.3432e-32  2.1583e-33  5.4636e-35  0.
U(OH)5-       8.1222e-23  2.7853e-24  8.1864e-26  2.0723e-27 1.1821-1360

precipitates

UO2OH)2C      0.          0.          0.          0.          0.
UO2(C)        0.          0.          0.          0.          0.


1    time= 2.40381e+03          delt=  5.0000e+00          concentration of species in moles/liter solution


                                                                 distance
   species         0.000       0.019       0.040       0.063       0.088       0.116       0.148       0.182       0.221

  UO2++         1.7018e-06  2.1257e-06  1.0999e-06  5.4821e-07  2.6316e-07  1.2140e-07  5.3658e-08  2.2656e-08  9.1079e-09
  CO3--         3.5306e-14  2.8454e-14  3.1207e-14  3.3390e-14  3.5860e-14  3.9227e-14  4.3520e-14  4.8235e-14  5.2604e-14
  H+            1.2833e-04  1.4330e-04  1.3663e-04  1.3196e-04  1.2725e-04  1.2158e-04  1.1536e-04  1.0954e-04  1.0492e-04
  E-            2.3951e-03  3.3042e-03  4.5884e-03  6.4941e-03  9.3666e-03  1.3781e-02  2.0710e-02  3.1843e-02  5.0177e-02
  OH-           1.8613e-08  1.6694e-08  1.7488e-08  1.8090e-08  1.8744e-08  1.9599e-08  2.0630e-08  2.1693e-08  2.2606e-08
  E-            2.1978e-01  2.0806e-01  1.9610e-01  1.8344e-01  1.7010e-01  1.5603e-01  1.4119e-01  1.2551e-01  1.0894e-01
  tempx         1.4999e+02  1.4999e+02  1.4998e+02  1.4997e+02  1.4996e+02  1.4995e+02  1.4992e+02  1.4988e+02  1.4982e+02
```

OUTPUT

```
 complexes

 HCO3        8.1883e-08  7.3456e-08  7.6976e-08  7.9658e-08  8.2584e-08  8.6411e-08  9.1061e-08  9.5920e-08  1.0023e-07
 H2CO3       9.2776e-05  9.2784e-05  9.2794e-05  9.2799e-05  9.2801e-05  9.2799e-05  9.2796e-05  9.2791e-05  9.2787e-05
 UO2OH+      3.9331e-06  4.3920e-06  2.3857e-06  1.2318e-06  6.1340e-07  2.9619e-07  1.3797e-07  6.1317e-08  2.5701e-08
 UO2CO3      3.0237e-08  3.0240e-08  1.7231e-08  9.2120e-09  4.7579e-09  2.4047e-09  1.1807e-09  5.5269e-10  2.4193e-10
 UO2CO3)2    5.8609e-15  4.7246e-15  2.9529e-15  1.6895e-15  9.3750e-16  5.1859e-16  2.8272e-16  1.4685e-16  7.0241e-17
 UO2+        6.1265e-07  1.0532e-06  7.5788e-07  5.3517e-07  3.7082e-07  2.5184e-07  1.6740e-07  1.0871e-07  6.8853e-08
 U(OH)5-     2.0687e-14  4.3977e-14  4.6076e-14  4.7676e-14  4.9420e-14  5.1702e-14  5.4472e-14  5.7357e-14  5.9901e-14

precipitates

 UO2OH)2C    2.9091e-04  2.2686e-05  0.          0.          0.          0.          0.          0.          0.
 UO2(C)      0.          1.0618e-05  3.0100e-05  1.4199e-05  6.6223e-06  3.0653e-06  1.4129e-06  6.5135e-07  3.0199e-07


                                                       distance
 species        0.263       0.310       0.362       0.419       0.483       0.553       0.631       0.717       0.812

 UO2++       3.4734e-09  1.2516e-09  4.2424e-10  1.3463e-10  3.9785e-11  1.0879e-11  2.7298e-12  6.2133e-13  1.2604e-13
 CO3--       5.6028e-14  5.8378e-14  5.9979e-14  6.1369e-14  6.3144e-14  6.5999e-14  7.0951e-14  7.9744e-14  9.5467e-14
 H+          1.0179e-04  9.9963e-05  9.9060e-05  9.8681e-05  9.8546e-05  9.8501e-05  9.8477e-05  9.8444e-05  9.8372e-05
 E-          8.1172e-02  1.3506e-01  2.3155e-01  4.0982e-01  7.5013e-01  1.4223e+00  2.7974e+00  5.7136e+00  1.2126e+01
 OH-         2.3245e-08  2.3581e-08  2.3656e-08  2.3516e-08  2.3162e-08  2.2525e-08  2.1460e-08  1.9750e-08  1.7164e-08
 E-          9.1402e-02  7.2844e-02  5.3196e-02  3.2402e-02  1.0425e-02 -1.2734e-02 -3.6997e-02 -6.2151e-02 -8.7766e-02
 tempx       1.4972e+02  1.4956e+02  1.4930e+02  1.4885e+02  1.4810e+02  1.4683e+02  1.4469e+02  1.4115e+02  1.3547e+02

 complexes

 HCO3        1.0353e-07  1.0580e-07  1.0743e-07  1.0898e-07  1.1108e-07  1.1449e-07  1.2035e-07  1.3048e-07  1.4791e-07
 H2CO3       9.2784e-05  9.2782e-05  9.2780e-05  9.2778e-05  9.2775e-05  9.2771e-05  9.2762e-05  9.2746e-05  9.2716e-05
 UO2OH+      1.0077e-08  3.6804e-09  1.2491e-09  3.9265e-10  1.1361e-10  2.9911e-11  7.0347e-12  1.4358e-12  2.4355e-13
 UO2CO3      9.7838e-11  3.6418e-11  1.2492e-11  3.9524e-12  1.1503e-12  3.0539e-13  7.2832e-14  1.5234e-14  2.6985e-15
 UO2CO3)2    3.0347e-17  1.1829e-17  4.2032e-18  1.3796e-18  4.2278e-19  1.2195e-19  3.3341e-20  8.6959e-21  2.1631e-21
 UO2+        4.2435e-08  2.5385e-08  1.4693e-08  8.1963e-09  4.3822e-09  2.2278e-09  1.0641e-09  4.6900e-10  1.8582e-10
 U(OH)5-     6.1807e-14  6.3057e-14  6.3851e-14  6.4472e-14  6.5196e-14  6.6297e-14  6.8093e-14  7.0986e-14  7.5364e-14

precipitates

 UO2OH)2C    0.          0.          0.          0.          0.          0.          0.          0.          0.
 UO2(C)      1.4159e-07  6.7333e-08  3.2407e-08  1.5685e-08  7.5632e-09  3.6010e-09  1.6797e-09  7.6356e-10  3.3879e-10


                                                       distance
 species        0.918       1.034       1.164       1.307       1.465       1.640       1.834       2.048       2.285

 UO2++       2.2172e-14  3.2514e-15  3.8094e-16  3.5696e-17  2.9278e-18  2.4967e-19  2.7864e-20  4.5683e-21  2.2586e-23
 CO3--       1.2323e-13  1.6943e-13  2.3590e-13  3.1247e-13  3.9949e-13  6.2458e-13  4.0784e-12  6.4197e-08  7.3326e-06
 H+          9.8200e-05  9.7776e-05  9.6701e-05  9.3946e-05  8.6842e-05  6.8448e-05  2.5103e-05  6.5828e-08  4.2940e-10
 E-          2.6741e+01  6.1215e+01  1.4515e+02  3.5414e+02  8.8520e+02  2.2555e+03  5.5265e+03  1.1858e+04  2.1100e+04
 OH-         1.3630e-08  9.5250e-09  5.7463e-09  3.1279e-09  1.7736e-09  1.3409e-09  2.6400e-09  8.6725e-07  1.2776e-04
 E-         -1.1314e-01 -1.3739e-01 -1.5981e-01 -1.8041e-01 -2.0037e-01 -2.2130e-01 -2.4268e-01 -2.6202e-01 -2.7727e-01
 tempx       1.2684e+02  1.1479e+02  9.9835e+01  8.3939e+01  6.9906e+01  5.9837e+01  5.4035e+01  5.1362e+01  5.0377e+01
```

OUTPUT

```
  complexes

 HC03         1.7713e-07  2.2307e-07  2.8675e-07  3.6035e-07  4.3872e-07  5.7291e-07  1.4563e-06  6.1036e-05  4.4511e-05
 H2C03        9.2659e-05  9.2546e-05  9.2319e-05  9.1839e-05  9.0729e-05  8.7932e-05  8.0692e-05  8.7446e-06  4.0958e-08
 U02OH+       3.2327e-14  3.1324e-15  2.1202e-16  1.0680e-17  5.0671e-19  3.3861e-20  7.7423e-21  4.1858e-19  2.9722e-19
 U02C03       3.8672e-16  4.2777e-17  3.5888e-18  2.4667e-19  1.7316e-20  1.8888e-21  1.3001e-21  3.1822e-18  1.6791e-18
 U02C03)2     5.0089e-22  1.0021e-22  1.5288e-23  1.6655e-24  1.5755e-25  2.5983e-26  1.1113e-25  4.1522e-18  2.4702e-16
 U02+         6.3974e-11  1.8490e-11  4.4222e-12  9.1381e-13  1.8305e-13  4.1124e-14  1.1817e-14  4.2148e-15  3.6684e-17
 U(OH)5-      8.1131e-14  8.6623e-14  8.7894e-14  8.1956e-14  7.3247e-14  7.5491e-14  1.7626e-13  5.6534e-11  1.3976e-10

precipitates

 U02OH)2C     0.          0.          0.          0.          0.          0.          0.          0.          0.
 U02(C)       1.5107e-10  7.2994e-11  4.3192e-11  3.2092e-11  2.7151e-11  2.3952e-11  2.3605e-11  0.          0.


                                                          distance
  species        2.548       2.839       3.160       3.516       3.910       4.346       4.829       5.363       5.954

 U02++        5.6795e-24  1.9733e-24  7.5474e-25  2.7617e-25  8.7905e-26  2.3154e-26  4.9658e-27  8.6828e-28  1.2464e-28
 C03--        8.7748e-06  5.8869e-06  2.7659e-06  9.8617e-07  2.7802e-07  6.3428e-08  1.1867e-08  1.8364e-09  2.3654e-10
 H+           1.9372e-10  1.3328e-10  1.1170e-10  1.0370e-10  1.0100e-10  1.0023e-10  1.0004e-10  1.0001e-10  1.0000e-10
 E-           3.1108e+04  3.9214e+04  4.4203e+04  4.6575e+04  4.7460e+04  4.7723e+04  4.7786e+04  4.7798e+04  4.7800e+04
 OH-          2.8105e-04  4.0929e-04  4.8956e-04  5.2802e-04  5.4242e-04  5.4670e-04  5.4772e-04  5.4792e-04  5.4795e-04
 E-          -2.8782e-01 -2.9420e-01 -2.9752e-01 -2.9897e-01 -2.9950e-01 -2.9965e-01 -2.9969e-01 -2.9970e-01 -2.9970e-01
 tempx        5.0086e+01  5.0016e+01  5.0002e+01  5.0000e+01  5.0000e+01  5.0000e+01  5.0000e+01  5.0000e+01  5.0000e+01

  complexes

 HC03         2.3652e-05  1.0795e-05  4.2239e-06  1.3942e-06  3.8240e-07  8.6545e-08  1.6162e-08  2.5001e-09  3.2200e-10
 H2C03        9.7213e-09  3.0341e-09  9.9169e-10  3.0344e-10  8.1019e-11  1.8193e-11  3.3911e-12  5.2437e-13  6.7534e-14
 U02OH+       1.6161e-19  8.0826e-20  3.6742e-20  1.4459e-20  4.7229e-21  1.2534e-21  2.6930e-22  4.7104e-23  6.7624e-24
 U02C03       4.8529e-19  1.1038e-19  1.9578e-20  2.5394e-21  2.2740e-22  1.3656e-23  5.4792e-25  1.4825e-26  2.7412e-28
 U02C03)2     8.5104e-17  1.2974e-17  1.0810e-18  4.9992e-20  1.2620e-21  1.7291e-23  1.2980e-25  5.4347e-28  1.2944e-30
 U02+         1.3449e-17  5.8419e-18  2.5068e-18  9.6443e-19  3.1257e-19  8.2767e-20  1.7773e-20  3.1085e-21  4.4625e-22
 U(OH)5-      1.6990e-10  1.3595e-10  7.8625e-11  3.4358e-11  1.1654e-11  3.1272e-12  6.7366e-13  1.1789e-13  1.6926e-14

precipitates

 U02OH)2C     0.          0.          0.          0.          0.          0.          0.          0.          0.
 U02(C)       0.          0.          0.          0.          0.          0.          0.          0.          0.
```

OUTPUT

```
                                                    distance
 species       6.608      7.332      8.133      9.019      10.000
 U02++      1.4805e-29 1.4653e-30 1.2162e-31 8.5367e-33 1.0000-2419
 C03--      2.5501e-11 2.3127e-12 1.7725e-13 1.1561e-14 1.5653-2425
 H+         1.0000e-10 1.0000e-10 1.0000e-10 1.0000e-10  1.0000e-10
 E-         4.7800e+04 4.7800e+04 4.7800e+04 4.7800e+04  4.7800e+04
 OH-        5.4795e-04 5.4796e-04 5.4796e-04 5.4796e-04  5.4796e-04
 E-        -2.9970e-01 -2.9970e-01 -2.9970e-01 -2.9970e-01 -2.9970e-01


 tempx      5.0000e+01 5.0000e+01 5.0000e+01 5.0000e+01  5.0000e+01

 complexes

 HC03       3.4714e-11 3.1482e-12 2.4129e-13 1.5737e-14  0.
 H2C03      7.2806e-15 6.6027e-16 5.0605e-17 3.3005e-18  0.
 U02OH+     8.0322e-25 7.9500e-26 6.5982e-27 4.6314e-28 5.6928-2426
 U02C03     3.5101e-30 3.1507e-32 2.0042e-34 9.1753e-37  0.
 U02C03)2   1.7868e-33 1.4546e-36 7.0914e-40 2.1174e-43  0.
 U02+       5.3004e-23 5.2462e-24 4.3541e-25 3.0563e-26  0.
 U(OH)5-    2.0104e-15 1.9899e-16 1.6515e-17 1.1592e-18 1.1821-2416

precipitates

 U02OH)2C   0.         0.         0.         0.          0.
 U02(C)     0.         0.         0.         0.          0.
```

APPENDIX 9

RECHARGE WATER COMPOSITION

INPUT AND SAMPLE OUTPUT

INPUT

```
Static equilibrium co2(g) disolving in water-partitioning of C considered
UNITS      meter    sec.
type               0         0         0         1
iteration        100       100       100
Calc.crtia   1.0e-30   1.0e-03        20        10         5      1.75         5
grid              05      10.0       0.0
time               1       1.0       5.0
print time       1.0
physical        0.05       0.0       0.0
tran bound         0         0
temp bound         0         0
isothermal         0
Intl.           10.0
leach              0         0
WORD      blank
In.sh.Info    -24.50     +0.01
Iso.Ratio1 -0.91e-03       6.3
         2 -4.54e-03    1099.0
         3 -3.40e-03     870.0
         4 -3.63e-03    1194.0
no.species         3         0         0         3         0         2         0
CO2(g)    0.0    0    23.1623e-023.1623e-02    23.1623e-023.1623e-02
H+        1.0    0    2   1.0e-05   1.0e-05    2   1.0e-05   1.0e-05
OH-       -80.3723-1.395e+03   32.5514-3.181e-02
            3.24e-10  3.24e-10
H2CO3       0  1.0  0.0  0.0
              87.598  -1055.95   -38.7233.4506e-02
                 0.0       0.0
HCO3-       0  1.0  0.0  1.0
             127.844   -532.58   -52.2593.1543e-02
                 0.0       0.0
CO3--       0 +1.0  0.0  2.0
             281.934  -3619.68  -112.5486.2045e-02
                 0.0       0.0
```

OUTPUT

```
1Static equilibrium co2(g) disolving in water-partitioning of C considered

    unit of length is   meter            unit of time is    sec.

    inner boundary is constant concentration          outer boundary is constant initial concentration


    dispersion coefficient= 0.1v+  0.           fluid velocity = 0.           number of grid points =    5


    number of print outs =    1
    given at the following times=  1.000

errsqr=   0.10e-29     climit=   0.10e-02     iterj=    20     iterm=    10     iterd=     5     dtmult=   0.17e+01     ndiv=

    number of sites(ion-exchange)    0.0000           number of sites(edl model)    0.0000
 this simulation is isothermal, temperature =  10.



     2      bare ions             0     sorbing ions (ionex)            0     sorbing ions (edl model)


component charge      kinetic     initial condition       boundary condition
                        code        type        value         type        value

CO2(g)       0.         0          2      3.1623e-02        2      3.1623e-02
H+           1.         0          2      1.0000e-05        2      1.0000e-05


     4     complexes             0       sorbing complexes(ionex)            0     sorbing complexes (edl model)


complex   charge       kcode                             stoichiometry

OH-         -1.        0         0.   0.   1.
H2CO3        0.        0         1.   0.   0.
HCO3-       -1.        0         1.   0.   1.
CO3--       -2.        0         1.   0.   2.

          C-13 Fractionation parameters:

          shiftr=   -24.5000          shifts=     0.0100

coefficients in log function of thermodynamic constants
              -0.001     6.300
              -0.005  1099.000
              -0.003   870.000
              -0.004  1194.000
```

OUTPUT

```
                                                      ** spatial grid **

     0.            2.500        5.000        7.500        10.00

 1                                                      initial conditions
                                                          temp=   10.00


 component       total           bare ion         sorbed concentration         act. coef.     kd(ion ex)        kd(edl)
              aqueous conc.    concentration      (ionex)     (edl)

 CO2(g)        0.3162e-01      0.3009e-01      0.          0.                    1.000          0.               0.
 H+            0.1000e-04      0.1000e-04      0.          0.                   0.9915          0.               0.
 OH-           0.5637e-04      0.3296e-09      0.          0.                   0.9915          0.               0.

 complex          valence        aqueous conc.   sorption(io)       sorption(edl)
 H2CO3         0.              0.1473e-02      0.              0.
 HCO3-         -1.000          0.5637e-04      0.              0.
 CO3--         -2.000          0.1865e-09      0.              0.


    to have a net balance of zero charge, other 'species must be present (although not participating in the
    chemical reactions) with a total concentration times charge of  0.464e-04


** Recharge water stable isotop characteristics **

Alfa(1) =   0.9992e+00     Alfa(2) =   0.1009e+01     Alfa(3) =   0.1007e+01     Alfa(4) =   0.1011e+01

shiftr =   -0.245e+02     H2CO3 =     0.147e-02     HCO3- =     0.564e-04     CO3-- =     0.187e-09

xmc =    0.153e-02     shftsol =    -0.249e+02     shift1 =    -0.253e+02     shift2 =    -0.155e+02     shift3 =    -0.172e+02
shifts =   -0.122e+02

 1                                boundary or influx conditions
                                         temp=   10.00
 component       total           bare ion         sorbed concentration         act. coef.     kd(ion ex)        kd(edl)
              aqueous conc.    concentration      (ionex)     (edl)

 CO2(g)        0.3162e-01      0.3009e-01      0.          0.                    1.000          0.               0.
 H+            0.1000e-04      0.1000e-04      0.          0.                   0.9915          0.               0.
 OH-           0.5637e-04      0.3296e-09      0.          0.                   0.9915          0.               0.

 complex          valence        aqueous conc.   sorption(io)       sorption(edl)
 H2CO3         0.              0.1473e-02      0.              0.
 HCO3-         -1.000          0.5637e-04      0.              0.
 CO3--         -2.000          0.1865e-09      0.              0.

    to have a net balance of zero charge, other 'species must be present (although not participating in the
    chemical reactions) with a total concentration times charge of  0.464e-04
```

APPENDIX 10

SCOPING SIMULATION OF YUCCA MOUNTAIN (NEVADA) GROUNDWATER

INPUT AND SAMPLE OUTPUT

INPUT

```
Groundwater evolution. Kinetic diss. of Calcit silica and glass-C-13 partition
UNITS     meter   second
Type                0        01         0         1
Iteration         100       100       100
Calc.crtia   1.0e-30   1.0e-03        20        10         5      1.75         5
Grid               20      10.0 836.68355
Time               17   5.0e-03  3.15e+04
Print time      360.0    3600.0   36000.0  144000.0  360000.0  864000.0 17.28e+05
print time 43.20e+05 86.40e+05 12.96e+06 17.28e+06 21.60e+06 25.92e+06 30.24e+06
print time 34.56e+06 38.88e+06 43.20e+06
Physical         0.10   1.0e-10  3.17e-08
Tran bound          1         0
Temp bound          0
Isothermal          0
Intl. temp       30.0
Leach               0         0
WORD      tpkin
*****Carbonate rock dissolution data; next 7 lines*****
1 PARAMS            5     100.0         4     prepc
2 CDATS          19.8     6.383    10.377     8.446 1.000e-03
3 INXi(i)     2    0   12    0    0    0    5   11
4 CONK(1)       -1.32         0         0         0
5 CONK(2)       -4.59         0         0         0
6 CONK(3)       -6.94         0         0         0
7 CONK(4)         0.0         0         0         0
WORD      tpkin
*****Silica dissolution data; next 4 lines*****
1 PARAMS            0     100.0         2     preps
2 INXi(i)     0    0   -9    0
3 CONK(1)      -0.369   -3438.0       0.0-7.890e-04
4 CONK(2)      -0.707   -2598.0       0.0       0.0
WORD      exkin
*****Glass dissolution data; next 1 line*****
1 PARAMS            0   3.0e+06         0     dissg
WORD      blank
 C-13 data      -17.4       1.4
1  CONP(1) -0.91e-03       6.3
2  CONP(2) -4.54e-03    1099.0
3  CONP(3) -3.40e-03     870.0
4  CONP(4) -3.63e-03    1194.0
No.species         10         0         0         2         2         2         0
Na+      +1.0    1    2   1.0e-10   1.0e-10    2  3.00e-04   3.0e-04
H+       +1.0    2    2 8.653e-05 1.237e-10    2  8.59e-04   5.1e-09
Co3--    -2.0   31    3 1.012e-02 3.521e-05    3  8.59e-04   1.0e-06
neg.KI    0.0   32    1 1.000e-10 1.000e-10    1 1.615e-02 1.615e-02
Ca++     +2.0    4    3 1.012e-02 1.217e-04    1  2.45e-04  2.45e-04
Cl-      -1.0    0    1   1.0e-10   1.0e-10    1  3.39e-05  3.39e-05
K+       +1.0    3    1   1.0e-10   1.0e-10    1  1.49e-04  1.49e-04
Mg++     +2.0    5    1   1.0e-10   1.0e-10    1  6.17e-05  6.17e-05
H4SiO4    0.0    0    3   1.0e-02   1.0e-10    3  2.67e-04  2.67e-04
OH-        -13.990       0.0       0.0       0.0
           8.653e-05   2.0e-06
```

INPUT

```
HCO3-       33  0.0  1.0  1.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0
  0.0
               10.337       0.0        0.0        0.0        0.0
            8.649e-05  8.59e-04
H2CO3       34  0.0  2.0  1.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0
  0.0
               16.700       0.0        0.0        0.0        0.0
            2.359e-08       0.0

CaCO3       13  0.0  0.0  1.0  0.0  1.0  0.0  0.0  0.0  0.0
  0.0
               -8.446       0.0        0.0        0.0        0.0
                 0.01       0.0
SiO2        14  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  1.0  0.0
  0.0
                -20.0
                 0.01       0.0
```

OUTPUT

```
1Groundwater evolution. Kinetic diss. of Calcit silica and glass-C-13 partition

    unit of length is  meter             unit of time is  second

kinetics of rev. precipitate dissolution considered


kinetics of rev. precipitate dissolution considered


kinetics of non-rev. silica dissolution considered


    inner boundary is constant flux          outer' boundry is constant initial concentration


    dispersion coefficient= 0.1v+  0.10e-09     fluid velocity = 0.317e-07     number of grid points =   20


    number of print outs =   17
    given at the following times=  360.0     3600.    0.3600e+050.1440e+060.3600e+060.8640e+060.1728e+070.4320e+07
                                  0.8640e+070.1296e+080.1728e+080.2160e+080.2592e+080.3024e+080.3456e+080.3888e+080.4320e+08

errsqr=   0.10e-29     climit=   0.10e-02     iterj=    20     iterm=    10     iterd=     5     dtmult=   0.17e+01     ndiv=

    number of sites(ion-exchange)    0.0000          number of sites(edl model)    0.0000
 this simulation is isothermal, temperature =  30.



     9      bare ions            0     sorbing ions (ionex)            0     sorbing ions (edl model)


component charge      kinetic     initial condition       boundary condition
                         code       type       value          type       value

Na+         1.          1          2      1.0000e-10        2      3.0000e-04
H+          1.          2          2      8.6530e-05        2      8.5900e-04
Co3--      -2.         31          3      1.0120e-02        3      8.5900e-04
neg.KI      0.         32          1      1.0000e-10        1      1.6150e-02
Ca++        2.          4          3      1.0120e-02        1      2.4500e-04
Cl-        -1.          0          1      1.0000e-10        1      3.3900e-05
K+          1.          3          1      1.0000e-10        1      1.4900e-04
Mg++        2.          5          1      1.0000e-10        1      6.1700e-05
H4SiO4      0.          0          3      1.0000e-02        3      2.6700e-04


     3     complexes            0       sorbing complexes(ionex)            0     sorbing complexes (edl model)
```

OUTPUT

```
complex   charge      kcode                         stoichiometry

OH-        -1.          0        0.    0.    0.    0.    0.    0.    0.    0.    0.    1.
HCO3-      -1.         33        0.    1.    1.    0.    0.    0.    0.    0.    0.    0.
H2CO3       0.         34        0.    2.    1.    0.    0.    0.    0.    0.    0.    0.



       2       solid precipitates

   precipitate  kinetic code          stoichiometry

  CaCO3            13            0.0  0.0  1.0  0.0  1.0  0.0  0.0  0.0  0.0  0.0
  SiO2             14            0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  1.0  0.0

  Calcite dissolution information:


   no of terms in the rate formulla =     4

   no of other additional data needed =     5

   representative surface parameter =    0.100e+03

   xdiag = prepc


   additional data for rate calculation

     0.198e+02   0.638e+01   0.104e+02   0.845e+01   0.100e-02

  binary element nos. of the terms of the rate equation
                 2    0   12    0    0    0    5   11

  coefficients in log function of thermodynamic constants
                -1.320     0.000      0.000      0.000
                -4.590     0.000      0.000      0.000
                -6.940     0.000      0.000      0.000
                 0.000     0.000      0.000      0.000

  Silicate-water reaction information:


   no of terms in the rate formulla =     2

   no of other additional data needed =     0

   representative surface parameter =    0.100e+03

   xdiag = preps
```

OUTPUT

```
binary element nos. of the terms of the rate equation
              0    0   -9    0

coefficients in log function of thermodynamic constants
              -0.369 -3438.000     0.000    -0.001
              -0.707 -2598.000     0.000     0.000

Glass dissolution information:


 no of terms in the rate formulla =     0

 no of other additional data needed =     0

 representative surface parameter =    0.300e+07

 xdiag = dissg


         C-13 Fractionation parameters:

         shiftr=   -17.4000          shifts=     1.4000

coefficients in log function of thermodynamic constants
              -0.001     6.300
              -0.005  1099.000
              -0.003   870.000
              -0.004  1194.000
                                                   ** spatial grid **

   0.           0.5086e-02   0.1233e-01   0.2266e-01   0.3738e-01   0.5835e-01   0.8823e-01   0.1308       0.1915       0.2780
   0.4012       0.5768       0.8271        1.184        1.692        2.416        3.448        4.918        7.014        10.00
```

OUTPUT

```
1                                                  initial conditions
                                                     temp=   30.00

component       total          bare ion         sorbed concentration        act. coef.     kd(ion ex)        kd(edl)
           aqueous conc.    concentration        (ionex)     (edl)

Na+          0.1000e-09      0.1000e-09     0.         0.                     0.9778         0.             0.
H+           0.8653e-04      0.1237e-09     0.         0.                     0.9778         0.             0.
Co3--        0.1217e-03      0.3521e-04     0.         0.                     0.9141         0.             0.
neg.KI       0.1000e-09      0.1000e-09     0.         0.                      1.000         0.             0.
Ca++         0.1217e-03      0.1217e-03     0.         0.                     0.9141         0.             0.
Cl-          0.1000e-09      0.1000e-09     0.         0.                     0.9778         0.             0.
K+           0.1000e-09      0.1000e-09     0.         0.                     0.9778         0.             0.
Mg++         0.1000e-09      0.1000e-09     0.         0.                     0.9141         0.             0.
H4SiO4       0.1000e-19      0.1000e-19     0.         0.                      1.000         0.             0.
OH-          0.8654e-04      0.8654e-04     0.         0.                     0.9778         0.             0.

complex         valence         aqueous conc.   sorption(io)      sorption(edl)
HCO3-        -1.000          0.8648e-04     0.              0.
H2CO3        0.              0.2359e-07     0.              0.

 precipitate   equivalent moles/liter solution

CaCO3    0.100e-01
SiO2     0.100e-01

   to have a net balance of zero charge, other 'species must be present (although not participating in the
   chemical reactions) with a total concentration times charge of  0.990e-08
```

OUTPUT

```
1                              boundary or influx conditions
                                         temp=   30.00
 component      total          bare ion          sorbed concentration         act. coef.     kd(ion ex)         kd(edl)
             aqueous conc.    concentration      (ionex)     (edl)

 Na+           0.3000e-03     0.3000e-03     0.          0.                    0.9601         0.               0.
 H+            0.8590e-03     0.5046e-08     0.          0.                    0.9601         0.               0.
 Co3--         0.8590e-03     0.9028e-05     0.          0.                    0.8496         0.               0.
 neg.KI        0.1615e-01     0.1615e-01     0.          0.                     1.000         0.               0.
 Ca++          0.2450e-03     0.2450e-03     0.          0.                    0.8496         0.               0.
 Cl-           0.3390e-04     0.3390e-04     0.          0.                    0.9601         0.               0.
 K+            0.1490e-03     0.1490e-03     0.          0.                    0.9601         0.               0.
 Mg++          0.6170e-04     0.6170e-04     0.          0.                    0.8496         0.               0.
 H4Si04        0.2670e-03     0.2670e-03     0.          0.                     1.000         0.               0.
 OH-           0.2200e-05     0.2200e-05     0.          0.                    0.9601         0.               0.

 complex         valence       aqueous conc.   sorption(io)      sorption(edl)
 HC03-         -1.000         0.8409e-03     0.              0.
 H2C03         0.             0.9023e-05     0.              0.


  precipitate   equivalent moles/liter solution

 CaC03    0.
 Si02     0.

    to have a net balance of zero charge, other 'species must be present (although not participating in the
    chemical reactions) with a total concentration times charge of -0.167e-03
```

```
1    time= 4.74489e+03          delt=  1.9447e+03         concentration of species in moles/liter solution


                                                             distance
   species       0.000       0.005       0.012       0.023       0.037       0.058       0.088       0.131       0.192

    Na+        1.1344e-05  3.2266e-06  9.6824e-07  6.8630e-07  6.6993e-07  6.6946e-07  6.6945e-07  6.6943e-07  6.6942e-07
    H+         1.3624e-10  1.2547e-10  1.2262e-10  1.2227e-10  1.2225e-10  1.2224e-10  1.2223e-10  1.2222e-10  1.2220e-10
    Co3--      4.0049e-05  3.6681e-05  3.5646e-05  3.5515e-05  3.5510e-05  3.5514e-05  3.5519e-05  3.5526e-05  3.5537e-05
    neg.KI     5.7509e-04  1.3760e-04  1.6056e-05  9.0329e-07  2.4882e-08  4.3726e-10  1.0232e-10  1.0001e-10  1.0000e-10
    Ca++       1.2614e-04  1.2281e-04  1.2188e-04  1.2177e-04  1.2177e-04  1.2178e-04  1.2179e-04  1.2181e-04  1.2183e-04
    Cl-        1.2073e-06  2.8893e-07  3.3802e-08  1.9959e-09  1.5202e-10  1.0071e-10  1.0000e-10  1.0000e-10  1.0000e-10
    K+         5.8201e-06  1.7955e-06  6.7695e-07  5.3747e-07  5.2938e-07  5.2916e-07  5.2916e-07  5.2916e-07  5.2916e-07
    Mg++       2.2637e-06  5.9250e-07  1.2754e-07  6.9489e-08  6.6117e-08  6.6022e-08  6.6018e-08  6.6015e-08  6.6010e-08
    H4SiO4     1.0027e-05  2.8053e-06  7.9851e-07  5.4831e-07  5.3381e-07  5.3341e-07  5.3340e-07  5.3340e-07  5.3340e-07
    OH-        7.8732e-05  8.5356e-05  8.7295e-05  8.7542e-05  8.7559e-05  8.7564e-05  8.7570e-05  8.7579e-05  8.7591e-05
shftsol       -2.4866e+00  3.2499e-01  1.2689e+00  1.3926e+00  1.3998e+00  1.4000e+00  1.4000e+00  1.4000e+00  1.4000e+00
shift0       -4.3565e+00 -1.6599e+00 -7.4909e-01 -6.2959e-01 -6.2267e-01 -6.2255e-01 -6.2265e-01 -6.2279e-01 -6.2300e-01
shift1        3.9017e+00  6.6206e+00  7.5390e+00  7.6595e+00  7.6665e+00  7.6666e+00  7.6665e+00  7.6664e+00  7.6661e+00
shift2        2.5455e+00  5.2608e+00  6.1779e+00  6.2982e+00  6.3052e+00  6.3053e+00  6.3052e+00  6.3051e+00  6.3049e+00

   complexes

    HCO3-      1.0789e-04  9.1293e-05  8.6787e-05  8.6230e-05  8.6201e-05  8.6205e-05  8.6211e-05  8.6220e-05  8.6232e-05
    H2CO3      3.2346e-08  2.5247e-08  2.3467e-08  2.3251e-08  2.3239e-08  2.3238e-08  2.3238e-08  2.3238e-08  2.3238e-08

  precipitates

    CaCO3      9.9983e-03  9.9983e-03  9.9983e-03  9.9983e-03  9.9983e-03  9.9983e-03  9.9982e-03  9.9982e-03  9.9982e-03
    SiO2       9.9995e-03  9.9995e-03  9.9995e-03  9.9995e-03  9.9995e-03  9.9995e-03  9.9995e-03  9.9995e-03  9.9995e-03


                                                             distance
   species       0.278       0.401       0.577       0.827       1.184       1.692       2.416       3.448       4.918

    Na+        6.6939e-07  6.6935e-07  6.6930e-07  6.6923e-07  6.6914e-07  6.6902e-07  6.6887e-07  6.6870e-07  6.6852e-07
    H+         1.2218e-10  1.2215e-10  1.2210e-10  1.2204e-10  1.2196e-10  1.2186e-10  1.2173e-10  1.2158e-10  1.2142e-10
    Co3--      3.5551e-05  3.5571e-05  3.5598e-05  3.5636e-05  3.5685e-05  3.5749e-05  3.5827e-05  3.5920e-05  3.6022e-05
    neg.KI     1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10
    Ca++       1.2186e-04  1.2191e-04  1.2197e-04  1.2205e-04  1.2216e-04  1.2230e-04  1.2247e-04  1.2268e-04  1.2290e-04
    Cl-        1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10  1.0000e-10
    K+         5.2916e-07  5.2916e-07  5.2916e-07  5.2916e-07  5.2916e-07  5.2916e-07  5.2916e-07  5.2916e-07  5.2916e-07
    Mg++       6.6004e-08  6.5995e-08  6.5983e-08  6.5967e-08  6.5945e-08  6.5918e-08  6.5884e-08  6.5845e-08  6.5804e-08
    H4SiO4     5.3340e-07  5.3340e-07  5.3340e-07  5.3340e-07  5.3340e-07  5.3340e-07  5.3340e-07  5.3340e-07  5.3340e-07
    OH-        8.7609e-05  8.7633e-05  8.7666e-05  8.7711e-05  8.7770e-05  8.7846e-05  8.7941e-05  8.8052e-05  8.8174e-05
shftsol        1.4000e+00  1.4000e+00  1.4000e+00  1.4000e+00  1.4000e+00  1.4000e+00  1.4000e+00  1.4000e+00  1.4000e+00
shift0       -6.2329e-01 -6.2370e-01 -6.2425e-01 -6.2500e-01 -6.2600e-01 -6.2728e-01 -6.2886e-01 -6.3072e-01 -6.3276e-01
shift1        7.6659e+00  7.6654e+00  7.6649e+00  7.6641e+00  7.6631e+00  7.6618e+00  7.6602e+00  7.6584e+00  7.6563e+00
shift2        6.3046e+00  6.3042e+00  6.3036e+00  6.3029e+00  6.3018e+00  6.3006e+00  6.2990e+00  6.2971e+00  6.2950e+00
```

OUTPUT

```
  complexes

 HCO3-       8.6249e-05  8.6274e-05  8.6307e-05  8.6351e-05  8.6411e-05  8.6487e-05  8.6582e-05  8.6694e-05  8.6816e-05
 H2CO3      -2.3239e-08  2.3239e-08  2.3239e-08  2.3239e-08  2.3239e-08  2.3240e-08  2.3240e-08  2.3241e-08  2.3241e-08

precipitates

 CaCO3       9.9982e-03  9.9981e-03  9.9981e-03  9.9980e-03  9.9979e-03  9.9977e-03  9.9976e-03  9.9974e-03  9.9971e-03
 SiO2        9.9995e-03  9.9995e-03  9.9995e-03  9.9995e-03  9.9995e-03  9.9995e-03  9.9995e-03  9.9995e-03  9.9995e-03

                                                          distance
   species        7.014       10.000
   Na+         6.6836e-07  1.0000e-10
   H+          1.2126e-10  1.2367e-10
   Co3--       3.6124e-05  3.5208e-05
   neg.KI      1.0000e-10  1.0000e-10
   Ca++        1.2312e-04  1.2171e-04
   Cl-         1.0000e-10  1.0000e-10
   K+          5.2917e-07  1.0000e-10
   Mg++        6.5764e-08  1.0000e-10
   H4SiO4      5.3341e-07  1.0000e-20
   OH-         8.8296e-05  8.6540e-05
shftsol        1.4000e+00  1.4000e+00
shift0        -6.3480e-01 -6.0566e-01
shift1         7.6543e+00  7.6836e+00
shift2         6.2930e+00  6.3223e+00

   complexes

  HCO3-        8.6938e-05  8.6483e-05
  H2CO3        2.3242e-08  2.3589e-08

 precipitates

  CaCO3        9.9969e-03  9.9983e-03
  SiO2         9.9995e-03  1.0000e-02
```

OUTPUT

```
1     time= 4.32179e+07          delt=  3.1500e+04          concentration of species in moles/liter solution


                                                       distance
    species          0.000       0.005       0.012       0.023       0.037       0.058       0.088       0.131       0.192

    Na+         3.0470e-04  3.0493e-04  3.0526e-04  3.0573e-04  3.0640e-04  3.0735e-04  3.0871e-04  3.1064e-04  3.1339e-04
    H+          2.6808e-09  2.6106e-09  2.5212e-09  2.4159e-09  2.3063e-09  2.2117e-09  2.1456e-09  2.0965e-09  2.0360e-09
    Co3--       1.7020e-05  1.7476e-05  1.8092e-05  1.8871e-05  1.9748e-05  2.0551e-05  2.1106e-05  2.1469e-05  2.1893e-05
    neg.KI      1.6126e-02  1.6125e-02  1.6123e-02  1.6120e-02  1.6116e-02  1.6110e-02  1.6099e-02  1.6081e-02  1.6046e-02
    Ca++        2.5025e-04  2.5051e-04  2.5080e-04  2.5106e-04  2.5106e-04  2.5036e-04  2.4846e-04  2.4499e-04  2.3960e-04
    Cl-         3.3849e-05  3.3847e-05  3.3843e-05  3.3838e-05  3.3829e-05  3.3815e-05  3.3793e-05  3.3754e-05  3.3682e-05
    K+          1.5115e-04  1.5126e-04  1.5141e-04  1.5162e-04  1.5193e-04  1.5236e-04  1.5298e-04  1.5384e-04  1.5504e-04
    Mg++        6.2651e-05  6.2697e-05  6.2764e-05  6.2859e-05  6.2994e-05  6.3186e-05  6.3462e-05  6.3858e-05  6.4427e-05
    H4SiO4      5.1032e-04  5.2228e-04  5.3917e-04  5.6294e-04  5.9623e-04  6.4250e-04  7.0612e-04  7.9227e-04  9.0637e-04
    OH-         4.1450e-06  4.2567e-06  4.4079e-06  4.6003e-06  4.8192e-06  5.0255e-06  5.1800e-06  5.3005e-06  5.4564e-06
shftsol        -1.7296e+01 -1.7291e+01 -1.7286e+01 -1.7281e+01 -1.7282e+01 -1.7299e+01 -1.7342e+01 -1.7420e+01 -1.7541e+01
shift0         -1.7476e+01 -1.7473e+01 -1.7471e+01 -1.7471e+01 -1.7477e+01 -1.7499e+01 -1.7546e+01 -1.7627e+01 -1.7751e+01
shift1         -9.3267e+00 -9.3241e+00 -9.3216e+00 -9.3212e+00 -9.3276e+00 -9.3494e+00 -9.3969e+00 -9.4784e+00 -9.6039e+00
shift2         -1.0665e+01 -1.0662e+01 -1.0660e+01 -1.0660e+01 -1.0666e+01 -1.0688e+01 -1.0735e+01 -1.0817e+01 -1.0942e+01

    complexes

    HCO3-       8.4071e-04  8.4054e-04  8.4026e-04  8.3975e-04  8.3879e-04  8.3705e-04  8.3406e-04  8.2927e-04  8.2173e-04
    H2CO3       4.7876e-06  4.6611e-06  4.4997e-06  4.3088e-06  4.1084e-06  3.9316e-06  3.8008e-06  3.6930e-06  3.5548e-06

  precipitates

    CaCO3       9.9107e-03  9.8216e-03  9.8115e-03  9.8410e-03  9.9299e-03  1.0065e-02  1.0178e-02  1.0213e-02  1.0201e-02
    SiO2        6.2162e-03  6.2391e-03  6.2712e-03  6.3162e-03  6.3787e-03  6.4644e-03  6.5804e-03  6.7335e-03  6.9288e-03


                                                       distance
    species          0.278       0.401       0.577       0.827       1.184       1.692       2.416       3.448       4.918

    Na+         3.1723e-04  3.2240e-04  3.2880e-04  3.2923e-04  2.6114e-04  1.4402e-04  8.0063e-05  6.5135e-05  6.2020e-05
    H+          1.9409e-09  1.7840e-09  1.5119e-09  9.6662e-10  3.2675e-10  1.0654e-10  6.6051e-11  6.0041e-11  6.0243e-11
    Co3--       2.2601e-05  2.3890e-05  2.6577e-05  3.4904e-05  5.8280e-05  6.3547e-05  4.9508e-05  4.5675e-05  4.5084e-05
    neg.KI      1.5977e-02  1.5821e-02  1.5435e-02  1.4031e-02  9.2288e-03  3.4089e-03  6.6229e-04  6.9372e-05  4.0841e-06
    Ca++        2.3128e-04  2.1784e-04  1.9441e-04  1.4542e-04  8.3433e-05  7.3212e-05  9.2452e-05  9.9991e-05  1.0123e-04
    Cl-         3.3537e-05  3.3210e-05  3.2399e-05  2.9452e-05  1.9372e-05  7.1555e-06  1.3903e-06  1.4572e-07  8.6725e-09
    K+          1.5666e-04  1.5863e-04  1.6040e-04  1.5648e-04  1.2574e-04  8.2922e-05  6.1652e-05  5.7124e-05  5.5211e-05
    Mg++        6.5243e-05  6.6399e-05  6.8017e-05  6.8514e-05  5.1693e-05  2.3752e-05  9.0588e-06  5.6852e-06  5.2057e-06
    H4SiO4      1.0527e-03  1.2319e-03  1.4366e-03  1.6458e-03  1.8097e-03  1.8875e-03  1.9074e-03  1.9116e-03  1.8969e-03
    OH-         5.7211e-06  6.2192e-06  7.3272e-06  1.1412e-05  3.3409e-05  1.0135e-04  1.6284e-04  1.7904e-04  1.7841e-04
shftsol        -1.7726e+01 -1.8023e+01 -1.8543e+01 -1.9636e+01 -2.0464e+01 -1.6021e+01 -4.4182e+00  7.0401e-01  1.3586e+00
shift0         -1.7943e+01 -1.8252e+01 -1.8802e+01 -2.0007e+01 -2.1406e+01 -1.8233e+01 -7.4116e+00 -2.4654e+00 -1.8066e+00
shift1         -9.7971e+00 -1.0109e+01 -1.0664e+01 -1.1879e+01 -1.3289e+01 -1.0090e+01  8.2124e-01  5.8085e+00  6.4727e+00
shift2         -1.1135e+01 -1.1446e+01 -1.2000e+01 -1.3214e+01 -1.4622e+01 -1.1427e+01 -5.3079e-01  4.4497e+00  5.1131e+00
```

OUTPUT

```
 complexes

HCO3-       8.0942e-04  7.8770e-04  7.4495e-04  6.3079e-04  3.6356e-04  1.3209e-04  6.4313e-05  5.3992e-05  5.3490e-05
H2CO3       3.3396e-06  2.9897e-06  2.3999e-06  1.3047e-06  2.5687e-07  3.0764e-08  9.3228e-09  7.1183e-09  7.0770e-09

precipitates

CaCO3       1.0182e-02  1.0160e-02  1.0135e-02  1.0103e-02  1.0065e-02  1.0037e-02  1.0024e-02  1.0022e-02  1.0022e-02
SiO2        7.1658e-03  7.4320e-03  7.6973e-03  7.9132e-03  8.0370e-03  8.0801e-03  8.0885e-03  8.0903e-03  8.0785e-03


                                                              distance
   species        7.014      10.000
  Na+         7.2687e-05  1.0000e-10
  H+          5.4590e-11  1.2367e-10
  Co3--       4.6232e-05  3.5208e-05
  neg.KI      1.4048e-07  1.0000e-10
  Ca++        9.9067e-05  1.2171e-04
  Cl-         3.9466e-10  1.0000e-10
  K+          6.5540e-05  1.0000e-10
  Mg++        5.9913e-06  1.0000e-10
  H4SiO4      2.0633e-03  1.0000e-20
  OH-         1.9706e-04  8.6540e-05
shftsol       1.3985e+00  1.4000e+00
shift0      -1.9390e+00 -6.0566e-01
shift1       6.3392e+00  7.6836e+00
shift2       4.9797e+00  6.3223e+00

   complexes

  HCO3-       4.9617e-05  8.6483e-05
  H2CO3       5.9433e-09  2.3589e-08

 precipitates

  CaCO3       1.0020e-02  9.9983e-03
  SiO2        8.2560e-03  1.0000e-02
```

APPENDIX 11

INPUT PREPARATION GUIDE

CHMTRNS

USER'S GUIDE

Card 1a: The title of the problem can be up to 80 characters long.

variable	**Title**
columns	1–80
format	20A4

TITLE	title information for the problem

Card 1b: List of units of length and time.

variable	**UIENGHT**	**UTIME**
columns	11–18	19–26
format	A8	A8

ULENGHT	Unit of length
UTIME	Unit of time

Card 2: geometry (1-D or radial), mode (static or dynamic), initial guess (given or calculated) and C-13 isotop partition option

variable	**ITYPE**	**IDYNAM**	**IG**	**IPART**
columns	11–20	21–30	31–40	41–50
format	I10	I10	I10	I10

ITYPE	geometry: 0-one-dimensional; 1-radial
IDYNAM	code to indicate if the problem should stop after computing equilibrium distribution of species for initial and boundary conditions 0-stop; 1-continue execution for time and transport effects
IG	code to indicate whether the program or the user will provide initial guesses of the concentrations: 0-user will; 1-program will

IPART	code to indicate if C-13 partition should be considered 0-no partition; 1-partition

Card 3a: These parameters control the number of iterations performed, when calculating the equilibrium distribution of species, before intermediate calculations are printed. If the calculations converge slowly because of a poor initial guess, for example, then examining the old and new guesses will provide an idea of how the guess can be improved. To suppress printing intermediate calculations, values of 100 are recommended for all of these parameters. If the calculations do not converge after 100 iterations, the program will stop executing and print a message.

variable	IN1	IN2	IN3
columns	11–20	21–30	31–40
format	I10	I10	I10

IN1	number of iterations before printing when calculating the equilibrium distribution of the initial conditions
IN2	number of iterations before printing when calculating the equilibrium distribution at the grid point where the boundary condition is specified
IN3	number of iterations before printing when calculating the equilibrium distribution at a node where a new precipitate is forming

Card 3b: These parameters specify the convergence criterion (ERRSQR) in dynamic mode, the concentration threshold (CLIMIT), the interval of iteration (ITERJ) when a new jacolian matrix is calculated, the number of iterations at which the current time step may be increased, upon convergence (ITERM), or decrease, when problem does not converge (ITERD), the multiplier **DTMULT** and the number (NDIV) of times that the time step can be decreased before the program stops due to nonconvergence

variable	CLIMIT	ERRSQR	ITERJ	ITERM	ITERD	DTMULT	NDIV
column	11–20	21–30	31–40	41–50	51–60	61–70	71–80
format	E10.3	E10.3	I10	I10	I10	E10.3	I10

CLIMIT	minimum concentration threshold of any species
ERRSQR	maximum least square error limit
ITERJ	interval of iterations when a new jacolian is calculated
ITERM	if solution converges within this number of iterations the time step is multiplied by DTMULT

ITERD	if the solution does not converge within this number of iterations, the time step is divided by DTDIV and the solution procedure is restarted at the previous time level plus the new time step
ITMULT	the time step multiplier
NDIV	number of times DELT can be decreased

Card 4: The user must supply several parameters which control the grid spacing. For an evenly spaced grid (GC = 0), the grid spacing DX is equal to **XMAX/(NMAX-1)**. For a variable grid, DX varies and is the difference between adjacent grid points. X(i) calculated from

$$X(i) = \left[\frac{GC^{\left(\frac{i-1}{N-1}\right)} - 1}{GC - 1} \right] XMAX + X(1) \qquad i = 2,NMAX$$

where X(1) = 0 for one-dimensional geometry and X(1) = R_w for radial geometry

variable	NMAX	XMAX	GC	R_w
columns	11-20	21-30	31-40	41-50
format	I10	F10.3	F10.3	F10.3

NMAX	number of grid points
XMAX	distance between the first and last grid points (L)
GC	parameter used to calculate the grid spacing (see equation above)
R_w	radious of inner boundary in axial symmetry case. To be left blank for one-dimensional geometry

Card 5: The user must specify the number of times at which the results are to be printed, the size of the first time step and the size of the maximum time step allowed. The initial time step is usually between 10^{-4} and 10^{-5} (in units of time used for the flow velocity). To minimize numerical dispersion, the maximum time step should be approximately DX/VO, where DX is the grid spacing calculated from the grid parameters of Card 4 and VO is the velocity of CARD 7.

variable	KMAX	DTINI	DTMAX
columns	11-20	21-30	31-40
format	I10	F10.3	F10.3

KMAX	number of times at which the results are to be printed

DTINI	size of the initial time step [t]
DTMAX	size of the maximum time step permitted [t]

Card 6: The program will print the results at times nearest those specified, **KMAX** times, where **KMAX** was given on **Card 5**, are read, seven to a card.

variable	TPRINT(K), K = 1, KMAX
columns	11-80/1-80
format	7F10.3/8F10.3

TPRINT	array which holds the times at which the results are to be printed [t]

Card 7: The hydrodynamic dispersion coefficient is calculated from the dispersivity, diffusion coefficient and fluid velocity given by the user (DOA * VO + DOB).

variable	DOA	DOB	VO
columns	11-20	21-30	31-40
format	F10.3	F10.3	F10.3

DOA	dispersivity [L]
DOB	diffusion coefficient $[L^2/t]$
VO	for one-dimensional flow (ITYPE = 0 on **Card 2**), this is the mean fluid velocity (i.e., the specific discharge divided by the porosity) [L/t]; for radial flow (ITYPE = 1 on **Card 2**), it is a constant equal to the radial distance times the mean fluid velocity [1/t]

Card 8: the user must provide integer codes for the type of flow boundary conditions at both the inner and outer boundaries. At the inner boundary, either a concentration or a flux boundary condition is possible. At the outer boundary, either a concentration boundary where the fluid remains at the initial conditions or a no-flux boundary is possible.

variable	IBND(1)	IBND(2)
columns	11-20	21-30
format	I10	I10

IBND(1)	type of boundary condition to be imposed at the inner boundary: 0-constant concentration; 1-constant flux

IBND(2)	type of boundary condition to be imposed at the outer boundary: 0-constant concentration equal to initial conditions; 1-no flux

Card 9: Integer codes for the type of thermal boundary conditions

variable	IBNDT(1)	IBNDT(2)
column	11–20	21–30
format	I10	I10

IBNDT(1)	type of boundary conditions imposed at the inner boundary: 0-constant temperature; 1-constant flux
IBNDT(2)	type of boundary conditions imposed at the outer boundary: 0-constant temperature equal to initial conditions; 1-no flux

Card 10: The thermal mode is specified here. The choices are constant temperature, mixing of waters with different temperatures, or a fixed temperature gradient.

variable	ITEMP
columns	11–20
format	I10

ITEMP	code for type of thermal mode: 0-constant temperature; 1-mixing; 2-fixed gradient

Card 10a: read only if ITEMP = 0 on Card 10

variable	TEMP
columns	11–20
format	F10.3

TEMP	constant temperature for this simulation

Card 10b: read only if ITEMP = 1 on Card 10.

variable	TEMP	TEMPI	HCW	HCM	TCM
columns	11–20	21–30	31–40	41–50	51–60
format	F10.3	F10.3	E10.2	E10.2	E10.2

TEMP	initial temperature
TEMPI	if IBND(2) = 0, constant temperature at inner boundary; if IBND(2) = 1, temperature of incoming fluid
HCW	heat capacity of fluid per unit volume of porous medium
HCM	heat capacity of fluid per unit volume of porous medium
TCM	thermal conductivity of saturated porous medium

Card 10c: read only if **TEMP = 2** on **Card 10.**

variable	TEMPI	TEMPO
columns	11–20	21–30
format	F10.3	F10.3

TEMPI	temperature at inner boundary
TEMPO	temperature at outer boundary

Card 11: There is an option in the program to allow a solid to leach into the groundwater; however, at this time, the option is limited to one solid located at the first grid point. The user must include the solid in the list of precipitates and give its position in that list on this card.

variable	LEACH	NB
columns	11–20	21–30
format	I10	I10

LEACH	position in the list of precipitates of the solid which is leaching: 0-no leaching
NB	location of the solid which is leaching: 0-no leaching; 1-solid is leaching at the first grid point

Card 12a: redox and kinetic option indicator–omit **cards 12** if **WORD** is BLANK

variable	WORD
column	11–15
format	A5

WORD option activator:

exkin–Claassen and White's (1979) imperical glass dissulation rate activated

tpkin–plummer et al. (1978)'s rate equation and/or Rimstidt and Barnes (1980) silica-water rate equation activated

redox–redox option activated

Card 12b: redox constants-needed only if WORD is REDOX

variable	GASCON	FARCON
column	11–20	21–30
format	F10.3	F10.3

GASCON	Universal gas constant
FARCON	Faraday constant

Card 12c: Information regrading the kinetic option considered

variable	non
column	1–80
format	80X

Card 12d: kinetic dissolution constants–glass dissolution information to be read last

variable	NPARAM	SAREA	NSRT	XDIAG
column	11–20	21–30	31–40	41–45
format	I10	F10.3	I10	A5

NPARAM	number of additional data for a kinetic equation to be read in the next card
SAREA	a constant representing surface influence in the kinetic mass transfer
NSRT	number of terms in the rate equation
XDIAG	**prepc**–calcite kinetics considered; **preps**–Silica kinetics considered; **dissg**-glass dissolution considered

Card 12e: additional constants used in the rate equation–<u>disregard if</u> NPARAM = 0

variable	CDAT(i), i = 1,NPARAM
column	1–80
format	8e10.2

CDAT	Array for rate equation parameters

Card 12f: designation of elements participating in the rate equation (not used for glass kinetics)–4 sets of data per card

variable	INXI(i)	INX2(i)	i = 1, NSRT
column	1–5	6–10	
format	4(2I5)		

INX1, INX2	designate the binary components of each term of the rate equation in the list of species of the problem. Forward reaction terms get a positive sequence number while back reactions elements are negative. 0 specifies a non-existing term. When both INX1 and INX2 are zero it implies just a value of 1.0 for the binary expression.

Card 12g: parameters for calculation of reaction terms in the rate equation–one card for each thermodynamic parameter

variable	CONK(i,j) j = 1,4 ; i = 1, NSRT
column	11–80
format	10x, 7f10.3

CONK	coefficients in function for long of reaction constants

Card 13a: omit if IPART = 0, Delta-13 for water and rock if IPART = 1.

variable	SHIFTR	SHIFTS
columns	11–20	21–30
format	10X, 2F10.3	

SHIFTR	delta C-13 in the sail water
SHIFTS	delta C-13 of the rocks

Card 13b: omit if IPART = 0, 4 cards are read here

variable	CONP(i,j) j = 1,2, i = 1,4
column	11–80
format	10X, 2F10.3

CONP	coefficients in function for log of equilibrium con-

Card 14: Integer data are read on this card: the number of basis species, the number of species which are sorbed following either an ion exchange mechanism or a surface complexation model, the number of aqueous complexes, the number of solid precipitates and the position of H^+ and e^- in the list of basis species which follows.

variable	NUM(1)	NUM(2)	NUM(3)	NUM(4)	NUM(5)	LNH	LNO
columns	11–20	21–30	31–40	41–50	51–60	61–70	71–80
format	I10	I10	I10	I10	I10	I10	I10

NUM(1)	number of basis species (OH^- must be included in the count if the dissociation of water is to be incorporated in the model)
NUM(2)	number of species (both basis species and complexes) which are sorbed according to an ion exchange mechanism
NUM(3)	number of species (both basis species and complexes) which are sorbed according to a surface complexation model
NUM(4)	number of aqueous complexes
NUM(5)	number of solid precipitates
LNH	position of H^+ in the list of basis species: 0-dissociation of water not included
LNO	position of e^- in the list of basis species: 0-redox not considered

Card 15: If any of the species are sorbed according to an ion exchange mechanism, then the user must specify the number of equivalents of sites per liter of solution that are available for sorption. This value is obtained from the cation exchange capacity (CEC), usually given in meq/g of solid, by multiplying the CEC by $10^{-3}\phi_s(1 - \phi)/\phi$, where ϕ is the porosity and ρ_s is the solid density (g/cm^3). This card should be omitted if none of the species is sorbed according to an ion exchange mechanism.

variable	CECNS
columns	11–20
format	F10.3

CECNS	concentration of sites in [equivalents/$\underline{l}$ solution] available for sorption in ion exchange model

Card 16a,b,c: If any of the species is sorbed according to the surface complexation model, then the user must specify the 3 parameters described below. **This set of cards are omitted if none of the species if sorbed according to the surface complexation model.**

variable	C1	C2	AREA	SOH
columns	11–20	21–30	41–50	51–60
format	F10.3	F10.3	F10.3	F10.3

C1	capacitance between the surface plane and the beta plane [$\mu\underline{F}/\underline{L}^2$]
C2	capacitance between the beta plane and the diffuse layer [$\mu\underline{F}/\underline{L}^2$]
AREA	sorption area [$\underline{L}^2/\underline{l}$ solution]
SOH	concentration of sites in [equivalents/$\underline{l}$ solution] available for forming complexes in surface complexation model

Card 16b: surface complexation parameters

variable	CON(1)	CON(2)	CON(3)	CON(4)
columns	11–20	21–30	31–40	41–50
format	F10.3	F10.3	F10.3	F10.3

CON(i)	coefficients in function for log of SOH^{2+} dissociation

Card 16b: surface complexation parameters

variable	CON(1)	CON(2)	CON(3)	CON(4)
columns	11–20	21–30	31–40	41–50
format	F10.3	F10.3	F10.3	F10.3

CON(i)	coefficient in function for log of SOH dissociation constant

Card 17a-e: This set of cards reads information about the basis species and constants involved in the processes of ion exchange or surface complexation. **Card 17a** is always read; both **Card 17b,c** and **Card 17c,d** are omitted if the basis species is not sorbed (ISORPI = ISORPE = 0). **Card 17b,c** are is read if the basis species is sorbed according to an ion exchange mechanism (ISORPI = 1); **Card 17d-e** is read if the basis species is sorbed according to a surface complexation model (ISORPE = 1). **GUESSI** and **GUESSB** can all be left blank if the program is to provide the guesses (IG = 1 on **Card 2**).

Card 17a

variable	DUM	VJ	KCODE	INDEXI	BI	GUESSI
columns	1–10	11–15	16–20	21–25	26–35	36–45
format	A10	F5.2	I5	I5	F10.3	F10.3

Card 17a continued

variable	INDEXB	BC	GUESSB	ISORPI	ISORPE
columns	46–50	51–60	61–70	71–75	76–80
format	I5	F10.3	F10.3	I5	I5

DUM	name of the species
VJ	charge of the species
KCODE	An integer code that assumes location numbers when kinetics or C-13 fractionation is considered. For glass dissolution consideration, the KCODE for species Na$^+$, H$^+$, K$^+$, Ca$^+$ and Mg^{+2} assumes 1, 2, 3, 4, and 5 respectively. For isotope studies KCODE assumes 32, 31, 33 and 34. For CO3^{2-}, KI, HCO3$^-$ and H_2CO_3 respectively.
INDEXI	type of initial condition: 0-the total concentration of the species will be given (including sorbed phase, aqueous phase and precipitate); 1-only the concentration of the basis species will be given; 2-the combined concentration of the basis species and the aqueous complexes will be given (if a species can form a precipitate, option 3 must be used instead of option 2 to allow for the inclusion of the solubility product if necessary) 3-the combined concentration of the **basis species, aqueous complexes** and **precipitates** will be given; 4-the initial concentration will be determined from a charge balance;
BI	value of the initial concentration according to the value of **INDEXI** [moles/l]
GUESSI	guess for the initial concentration according to the value of **INDEXI** [moles/l]
INDEXB	takes the same values as **INDEXI** and describes what type of concentration is being specified at the boundary
BC	value of the concentration at the boundary [moles/l] according to the value of **INDEXB** [moles/l]; if a solid is leaching into the ground water at the inner boundary, the value of BC for the first species making up this solid is the amount of the solid while the value of BC need not be specified for the other species making up the solid
GUESSB	guess for the boundary concentration of the basis species [moles/l]; **can be left blank if the program is to provide the guess (IG = 1 on Card 2)**
ISORPI	code to indicate whether the species is sorbed according to an ion exchange mechanism: 0-is not sorbed; 1-is sorbed

ISORPE	code to indicate whether the species is sorbed according to a surface complexation model: 0-is not sorbed; 1-is sorbed

Card 17b: ion exchange sorption

variable	GUESSI	GUESSB
columns	11–20	21–30
format	F10.3	F10.3

GUESSI	guess for the amount of this species which is sorbed initially [moles/l]; **can be left blank if the program is to provide the guess (IG = 1 on Card 2)**
GUESSB	guess for the amount of this species which is sorbed initially at the boundary conditions [moles/l]; **can be left blank if the program is to provide the guess (IG = 1 on Card 2)**

Card 17c: ion exchange equilibrium parameters

variable	CON(i)	CON(i)	CON(3)	CON(4)
column	11–20	21–30	31–40	41–50
format	F10.3	F10.3	F10.3	F10.3

CON(i)	coefficients in function for log of equilibrium constant

Card 17d: surface complexation

variable	GUESSI	GUESSB	AH
columns	11–20	21–30	31–40
format	F10.3	F10.3	10.3

GUESSI	guess for the amount of this species which is sorbed initially [moles/l]; **can be left blank if the program is to provide the guess (IB = 1 on Card 2)**
GUESSB	guess for the amount of this species which is sorbed initially at the inner boundary; [moles/l] **can be left blank if the program is to provide the guess (IB = 1 on Card 2)**
AH	amount of charge which is displaced or sorbed for each surface complexation reaction with a SOH site: +1–a H^+ ion is displaced, 0–used for the H^+ ion,

-1-a H^+ ion is sorbed

Card 17e: surface complexation equilibrium parameters

variable	CON(1)	CON(2)	CON(3)	CON(4)
column	11-20	21-30	31-40	41-50
format	F10.3	F10.3	F10.3	F10.3

CON(i)	coefficients in function for log of equilibrium constant

Cards 18a,b: These two cards are omitted if water dissociation is not included in the simulation (LNH = 0 on Card 14). If water dissociation is allowed, information bout OH must be read immediately after all of the information about the basis species has been read. OH must be included in the number of basis species (see NUM(1)) read on **Card** 14. If **Card 18** is used, **Card 18** must be included.

Card 18a

variable	DUM	H2OC(1)	H2OC(2)	H2OC(3)	H2OC(4)
columns	1-10	11-20	21-30	31-40	41-50
format	A10	F10.3	F10.3	F10.3	F10.3

DUM	name of the species
H2OC(i)	coefficients in function for log of dissociation constant of water

Card 18b

variable	GUESSI	GUESSB
columns	11-20	21-30
format	F10.3	F10.3

GUESSI	guess for the amount of this species which is sorbed initially [moles/l]; **can be left blank if the program is to provide the guess (IG = 1 on Card 2)**
GUESSB	guess for the amount of this species which is sorbed initially at the boundary [moles/l]; **can be left blank if the program is to provide the guess (IG = 1 on Card 2)**

Cards 19a–g: These seven cards are omitted if NUM(4) = 0 on **Card 14**, that is, if no aqueous complexes form. If the aqueous complex is sorbed according to an ion exchange mechanism, then **Cards 19a-e** are read; if the complex is sorbed according to a surface complexation reaction, then **Cards 19a-c** and **f,g** are read. If the complex is not sorbed, then only **Cards 19a-c** are read.

Card 19a

variable	DUM	KCODE	S(J), J = 1, NUM(1)
columns	1–10	11–20	21–70,/,1–75
format	A10	I10.3	10F5.1,/,15F5.1

DUM	name of the complex
KCODE	integer code used in kinetic options. For details see information about the basis species given above
S	array which holds the stoichiometric coefficients for the reaction, read in the same order in which the basis species are read

Card 19b: complexation equilibrium constant parameters

variable	CON(1)	CON(2)	CON(3)	CON(4)
column	11–20	21–30	31–40	41–50
format	F10.3	F10.3	F10.3	F10.3

CON(i)	coefficients in function for log of formation of the complex

Card 19c

variable	GUESSI	GUESSB	ISORPI	ISORPE
columns	11–20	21–30	31–35	36–40
format	F10.3	F10.3	I5	I5

GUESSI	guess for the initial concentration of this complex [moles/l]; can be left blank if the program is to provide the guess (IG = 1 on Card 2)
GUESSB	guess for the concentration of this complex at the boundary [moles/l]; can be left blank if the program is to provide the guess (IG = 1 on Card 2)
ISORPI	code to indicate whether the complex is sorbed according to an ion exchange mechanism:

	0-is not sorbed; 1-is sorbed
ISORPE	code to indicate whether the complex is sorbed according to a surface complexation model: 0-is not sorbed; 1-is sorbed

Card 19d: ion exchange

variable	GUESSI	GUESSB
columns	11–20	21–31
format	F10.3	F10.3

GUESSI	guess for the amount of this complex which is sorbed initially [moles/l]; **can be left blank if the program is to provide the guess (IG = 1 on Card 2)**
GUESSB	guess for the amount of the complex which is sorbed at the boundary conditions [moles/l]; **can be left blank if the program is to provide the guess (IG = 1 on Card 2)**

Card 19e: ion exchange equilibrium parameters

variable	CON(1)	CON(2)	CON(3)	CON(4)
column	11–20	21–30	31–40	41–50
format	F10.3	F10.3	F10.3	F10.3

CON(i)	coefficients in function for log of thermodynamic equilibrium constant

Card 19f: surface complexation parameters

variable	GUESSI	GUESSB	AH
columns	11–20	21–30	31–40
format	F10.3	I10.3	F10.3

GUESSI	guess for the amount of this complex which is sorbed initially [moles/l]; **can be left blank if the program is to provide the guess (IG = 1 on Card 2)**
GUESSB	guess for the amount of this complex which is sorbed at the boundary [moles/l]; **can be left blank if the program is to provide the guess (IG = 1 on Card 2)**
AH	amount of charge which is displaced or sorbed for each surface complexation reaction with a SOH site: +1-a H^+ ion is displaced, 0-used for the H^+ ion,

-1-a H^+ ion is sorbed

Card 19g: surface complexation equilibrium parameters

variable	CON(1)	CON(2)	CON(3)	CON(4)
column	11-20	21-30	31-40	41-50
format	F10.3	F10.3	F10.3	F10.3

CON(i)	constants in function for log of thermodynamic surface exchange constant

Cards 20a,b,c: These three cards are omitted if NUM(5) = 0 on Card 14, that is if no precipitates form. The concentration units for the precipitate are in equivalent moles/l of solution.

Card 19a

variable	DUM	KCODE	SS(J), J = 1, NUM(1)
columns	1-10	11-20	21-70,/,1-75
format	A10	I10.3	10F5.1,/,15F5.1

DUM	name of the precipitate
SS	array which holds the stoichiometric coefficients for the reaction, read in the same order in which the basis species are read
KCODE	integer code that assumes the rank of the solid in the list of species as they appear in the input

Card 20b: solubility product parameters

variable	CON(1)	CON(2)	CON(3)	CON(4)
column	11-20	21-30	31-40	41-50
format	F10.3	F10.3	F10.3	F10.3

CON(i)	coefficients in function for log of solubility product

Card 20c

variable	GUESSI	GUESSB
columns	11-20	21-30
format	F10.3	F10.3

GUESSI	guess for the initial concentration of this precipitate [moles/l]; can be left blank if the program is to provide the guess (IG = 1 on Card 2)
GUESSI	guess for the concentration of this precipitate at the boundary [moles/l]; can be left blank if the program is to provide the guess (IG = 1 on Card 2)

APPENDIX 12

FORTRAN IV LISTING OF CHMTRNS

```
c
c  ******************************************************************
c                            CHMTRNS CODE
c              (CHEMICAL TRANSPORT NUMERICAL SIMULATOR)
C  ******************************************************************
c  This version of the CHEMTRN code(originally developed by C. W. Miller
c  and L. V. Benson, 1983) called CHMTRNS includes kinetic dissolution/
c  precipitation, redox, temprature variation, and C-13 fractionation
c  options.
c
c  Authors       Noorishad, J., Carnahan, C. L.
c                earth sciences division
c                lawrence berkeley lab
c                berkeley, ca 94720
c                         and
c                Benson, L. V
c                U.S. Geological Survey
c                Denver, Colorado
c
c  Description   CHMTRNS simulates solute transport for steady one-dim.
c                fluid flow by convection and diffusion/dispersion in a
c                saturated porous medium based on the assumption of loca
c                chemical equilibrium*. the chemical interactions includ
c                in the model are aqueous-phase complexation, solid-phas
c                ion exchange of bare ions and complexes, solid-phase
c                sorption of bare ions and complexes using the surface
c                complexation model, and precipitation/dissolution of
c                solids.
c  *             The code is also capable of handling kinetic
c  *             dissolution/precipitation for Calcite and Silica and
c  *             kinetic dissolution for Glass. Also, options for redox
c  *             and C-13 fractionation as well as non-isothermal
c  *             conditions are included .
c
c  solution procedure      for chemical transport the finite differnce
c                          approaximation along with Newton-Raphson
c                          method is used. for the heat transport a
c                          direct LU solution of the crank-Nicholson
c                          approaximation is employed.
c
c
c
c
c   description of input variables
c
c          title = title of run up to 80 characters
c
c          ulength = unit of length used in the problem
c          utime  =  unit of time used in the problem
c
c          itype  = 0 if one-dimensional flow is modelled
c                 = 1 if radial flow is modelled
c          idynam = 0   static distribution of species calculated
c                 = 1   dynamic transport calculation
c          ig     =  0 user provides the initial guesses
c          ipart  # 0 C-13 fractionation considered
c          gascon = gas constant
c          farcon = Faradays constant
c                 =  1 program provides the initial guesses
c          nsrt   = 0 if kinetic option is not used
```

```
c             = n number of terms in the k.rate equation            chmtrns.62
c      nparam = m number of additional data for rate equation       chmtrns.63
c      sarea  =   a parameter representing surface effects          chmtrns.64
c      conk(1,k) = parameters of the themodynamic constants of the  chmtrns.65
c                  kinetic reactions                                chmtrns.66
c      shiftr    = Delta 13 of the aqueous carbonate solution       chmtrns.67
c      shifts    = Delta 13 of the solid carbonate                  chmtrns.68
c      conp(j,k) = parameters of the thermodynamic constants of     chmtrns.69
c                  fractionation.                                   chmtrns.70
c                                                                   chmtrns.71
c      in1, in2, in3 = number of iterations done in equilibrium spec chmtrns.72
c             subroutine until printout is desired.  if the speciati chmtrns.73
c             concludes before, this value is obtained, there is no  chmtrns.74
c             in1 is for the initialization calculation, in2 is for  chmtrns.75
c             calculation for the boundary or influx conditions, and chmtrns.76
c             there is an equilibrium speciation calculation when a  chmtrns.77
c             forms.  if the values are set to 100 or greater, no pr chmtrns.78
c             will occur here.                                       chmtrns.79
c                                                                   chmtrns.80
c      nmax = maximum number of grid points. limited to 100 points a chmtrns.81
c             present                                                chmtrns.82
c      xmax = spatial distance over which problem is solved         chmtrns.83
c      gc        = constant in the mesh generation equation.        chmtrns.84
c                 if gc=0, a uniform grid is generated of           chmtrns.85
c                 spacing 1/(nmax-1).                               chmtrns.86
c      rw  =  position where a flux boundary condition is imposed fo chmtrns.87
c             the radial flow problem with a flux condition at the i chmtrns.88
c             boundary                                               chmtrns.89
c                                                                   chmtrns.90
c      kmax = number of printouts desired                           chmtrns.91
c      dtini  = first time step used, usually about 1.e-4 or 1.e-5 t chmtrns.92
c               for the initial large changes which usually occur    chmtrns.93
c      dtmax   = maximum time step desired.  if convergence is withi chmtrns.94
c                time step is multiplied by 1.75 each time.  however chmtrns.95
c                other criteria such as numerical dispersion which m chmtrns.96
c                considered. usually there is a maximum time step wh chmtrns.97
c                be used.                                            chmtrns.98
c                                                                   chmtrns.99
c      tprint = array containing printout times                     chmtrns.100
c                                                                   chmtrns.101
c      doa = dispersivity (it is multiplied by the velocity         chmtrns.102
c            in the program)                                        chmtrns.103
c      dob = diffusion coefficient                                  chmtrns.104
c      vo = average fluid velocity for 1-d case.                    chmtrns.105
c         = one radiant flux for radial case                        chmtrns.106
c      ibnd(1) = 0  constant concentration at inner boundary        chmtrns.107
c              = 1  constant flux at inner boundary                 chmtrns.108
c      ibnd(2) = 0  constnat concentration equal to initial conditon chmtrns.109
c                   outer boundary                                  chmtrns.110
c                1  no flux at outer boundary                       chmtrns.111
c      ibndt(1) = 0 constant temp at inner boundary                 chmtrns.112
c                 1 constant heat flux at inner boundary            chmtrns.113
c      ibndt(2) = 0 constant temp at outer boundary                 chmtrns.114
c      itemp    = 0 isothermal                                      chmtrns.115
c               = 1 transient                                       chmtrns.116
c               = 2 stady state                                     chmtrns.117
c      tempi    =   initial temp or internal temp for steady state  chmtrns.118
c      tempo    =   external temp for the steady state case         chmtrns.119
c      hcw      =   heat capacity of fluid per unit volume          chmtrns.120
c                   porous medium                                   chmtrns.121
```

```
c          hcm      =   heat capacity of solid per unit volume of       chmtrns.122
c                       porous medium                                   chmtrns.123
c          tcm      =   thermal conductivity of saturated porous medium chmtrns.124
c                                                                       chmtrns.125
c          lnh      = the position of the H+ ion in basis species list  chmtrns.126
c                       0, when water dissociatin is not considered     chmtrns.127
c          lno      =   the place of e- in the list of basis elements   chmtrns.128
c                                                                       chmtrns.129
c                                                                       chmtrns.130
c          leach =  position in the list of precipitates of the solid wh chmtrns.131
c                  has been put in contact with the groundwater.  the eq chmtrns.132
c                  condition used is that the total concentration of thi chmtrns.133
c                  precipitate is given and that the species making up t chmtrns.134
c                  precipitate are entering the groundwater equivalent t chmtrns.135
c                  stoichiometric coefficients; equals 0 it no leaching  chmtrns.136
c          nb   = the number of the node where the solid was placed in c chmtrns.137
c                 with the groundwater.  in this version it must be at n chmtrns.138
c                 if no solid is put in contact with the groundwater, nb chmtrns.139
c                                                                       chmtrns.140
c          num(1) = number of basis species (must include hydroxyl ion i chmtrns.141
c                   dissociation is included)                           chmtrns.142
c          num(2) = number of cations and complexes sorbed by ion exchan chmtrns.143
c          num(3) = number of ions and complexes sorbed by surface compl chmtrns.144
c                   model (value is increased by 2 in program to allow f chmtrns.145
c                   inclusion of negatively charged surface site and pos chmtrns.146
c                   charged surface site)                               chmtrns.147
c          num(4) = number of aqueous complexes                         chmtrns.148
c          num(5) = number of precipitates that could form              chmtrns.149
c                                                                       chmtrns.150
c          cecns = concentration of sites available for sorption in ion  chmtrns.151
c                  model; expressed in equivalents/liter solution       chmtrns.152
c                                                                       chmtrns.153
c          c1  = capacitance between the surface plane and the beta plan chmtrns.154
c                surface complexation model (microfarads/length2)       chmtrns.155
c          c2  = capacitance between the beta plane and the diffuse laye chmtrns.156
c                complexation model (microfarads/length2)               chmtrns.157
c          area = area of sites for sorption in surface complexation in  chmtrns.158
c                 length2/liter solution                                chmtrns.159
c          soh  = concentration of sites for forming complexes in surfac chmtrns.160
c                 complexation model; in equivalents/liter solution     chmtrns.161
c                                                                       chmtrns.162
c          dum  = array containing the names of the basis species, compl chmtrns.163
c                 the precipitates                                      chmtrns.164
c          vj = array containing ion valences and complex valences; the  chmtrns.165
c              valances are computed from the valences of the basis spec chmtrns.166
c          kcode(j) =   code for the rate generation of basis species   chmtrns.167
c                       used presently for Glass dissociation option    chmtrns.168
c          con(i,j) =   parameters of the thermodynamic constants in    chmtrns.169
c                       mass action equations                           chmtrns.170
c           h2oc(i) =   parameters of the thermodynamic constants for   chmtrns.171
c                       water dissociation                              chmtrns.172
c          indexi = array giving type of initial conditions used for    chmtrns.173
c                   each particular species                             chmtrns.174
c                   value of 0 means total concentration is specified   chmtrns.175
c                   value of 1 means basis species concentration is give chmtrns.176
c                   value of 2 means total aqueous concentration is     chmtrns.177
c                   given excluding any precipitates (however, if a spec chmtrns.178
c                   form a precipitate, the value of 3 should be used in chmtrns.179
c                   2 to allow for the inclusion of the solubility produ chmtrns.180
c                   necessary);                                         chmtrns.181
```

```
c                 value of 3 means total aqueous concentration plus     chmtrns.182
c                 any precipitates is given                             chmtrns.183
c                 value of 4 means that the concentration of this       chmtrns.184
c                 species comes from a charge balance                   chmtrns.185
c         bi = array of initial conditions                              chmtrns.186
c         guessi = guess of equilibrated initial conditions             chmtrns.187
c                 for all the species                                   chmtrns.188
c         indexb = array giving type of conditions used to determine    chmtrns.189
c                 boundary or influx conditions; values of 0 to 4 are   chmtrns.190
c                 same as given in indexi above                         chmtrns.191
c         bc = array of boundary conditions                             chmtrns.192
c         guessb = guess of equilibrated boundary or influx conditions  chmtrns.193
c                 for all the species                                   chmtrns.194
c         isorpi  = array containing information about whether the basi chmtrns.195
c                  or the complex sorbs via ion exchange;               chmtrns.196
c                  0 - no sorption                                      chmtrns.197
c                  1 - sorption                                         chmtrns.198
c         isorpe  = array containing information about whether the basi chmtrns.199
c                  or complex sorbs via surface complexation            chmtrns.200
c                  0 - no sorption                                      chmtrns.201
c                  1 - sorption                                         chmtrns.202
c               mass action equations describing sorption via ion exch  chmtrns.203
c         pke  = array containing negative log of the equilibrium const chmtrns.204
c               mass action equations describing sorption via surface   chmtrns.205
c                                                                       chmtrns.206
c         s(j,i) = the stoichiometric coefficients for the aqueous      chmtrns.207
c                 phase complexes; given for 1 mole of the complex and  chmtrns.208
c                 same order as the basis species are given             chmtrns.209
c                                                                       chmtrns.210
c                                                                       chmtrns.211
c         h2ok  =  negative log of the dissociation of water            chmtrns.212
c                                                                       chmtrns.213
c         ss(j,i) = stoichiometric coefficients for precipitates; given chmtrns.214
c                  same order as the list of the basis species          chmtrns.215
c                                                                       chmtrns.216
c         ah(i)    =  number of h ions released or taken up by the surfa chmtrns.217
c                    in surface complexation model                      chmtrns.218
c                    +1 - if h+ ion is released                         chmtrns.219
c                     0 - special case for h+ ion                       chmtrns.220
c                    -1 - if h+ ion is sorbed                           chmtrns.221
c                                                                       chmtrns.222
c                                                                       chmtrns.223
c    description of other important variables used in the program       chmtrns.224
c                                                                       chmtrns.225
c         cdat(i) = array containing pk of needed equilib. constants    chmtrns.226
c                  for calculation of reaction constants of the rate    chmtrns.227
c                  equation and some other needed data                  chmtrns.228
c         inx1(j),inx2(j) = array designating the binary components     chmtrns.229
c                          of each term of the rate equattion           chmtrns.230
c          reactsk(i,j,n) = array containing reaction constants         chmtrns.231
c                     in the rate equation                              chmtrns.232
c        cn(j)  = array containing all the unknown species. the order i chmtrns.233
c                those in node 1, then those in node 2, etc. the order  chmtrns.234
c                at each node is first the basis species (the last bas  chmtrns.235
c                is always oh if water dissociation is included), then  chmtrns.236
c                basis species and complexes via simple ion exchange,   chmtrns.237
c                basis species and complexes via surface complexation,  chmtrns.238
c                and the last are the precipitates.                     chmtrns.239
c        co(j)  = array containing all the old values of the species in chmtrns.240
c                order as listed for cn(j)                              chmtrns.241
```

```
c         r(i,n)  = residues from the equations are stored here.  n is t chmtrns.242
c                   i is the number of the equation.  the order of the e chmtrns.243
c                   at each node is (1)transport equations, (2)the site  chmtrns.244
c                   (3)mass action equations for sorption via ion exchan chmtrns.245
c                   (4) site constraint for sorption via surface complex chmtrns.246
c                   action equations for surface complexation, (6) mass  chmtrns.247
c                   for formation of complexes in the aqueous phase, and chmtrns.248
c                   products                                             chmtrns.249
c         ac(i,n)  =  activities of the basis species and complexes in t chmtrns.250
c                    phase are stored here                               chmtrns.251
c           equconst(j,n) array containing the equilibrium constants     chmtrns.252
c                        first for sorbed species via ion exchange,      chmtrns.253
c                       then for species sorbed via surface complexation chmtrns.254
c                       then for aqueous phase complexation,             chmtrns.255
c                       then solubility products                         chmtrns.256
c         a(i,j,n)  = the diagonal matrix of the block tridiagonal jacob chmtrns.257
c                     it is change in residue divided by the change in c chmtrns.258
c                     at the node n                                      chmtrns.259
c         b(i,j,n)  = off diagonal matrix of the block tridiagonal jacob chmtrns.260
c         c(i,j,n)  = off diagonal matrix of the block tridiagonal jacob chmtrns.261
c                                                                        chmtrns.262
      dimension cn(3000),co(3000),dum(20),title(20),pki(30),pke(30)      chmtrns.263
      dimension cm(25),wconst(25),bi(25),bc(25),tprint(25),num(20),      chmtrns.264
     1index1(10),index2(25),indexb(25),cmax(25),ibnd(2),bc2(25),         chmtrns.265
     2iip(3000),guessi(30),guessb(30),indexi(25),com(100),ibndt(2)       chmtrns.266
      dimension tmp(100),tmpold(100),eh(100)                             chmtrns.267
      common/ab/vj(75),ac(75,100),eqconst(25,100),h2ok(100),alfa(4,100),    /ab/.2
     1s(25,25),ss(25,25),ah(25),h2oc(8),con(25,8),conk(20,8),conp(4,2),     /ab/.3
     2 ilbli(25),ilble(25),q(10,100)                                        /ab/.4
      common/edl/c1,c2,area,soh                                            /edl/.2
      common/edl2/sigo,sigb,sigd,psio,psibeta,psid                        /edl2/.2
      common/nj/isorpi(30),isorpe(30),iion,ieion                            /nj/.2
      common/ad/x(200),delt,doa,dob,vo,cecns                                /ad/.2
      common/const/rel,absv                                              /const/.2
      common/lcm/a(25,25,100),b(25,25,100)                                 /lcm/.2
      common/lcm1/r(25,100)                                               /lcm1/.2
      common/lcm2/c(25,25,100)                                            /lcm2/.2
      common/oxred/lno,lnh                                               /oxred/.2
      common/spec/leach1(5),nb,pini(5)                                    /spec/.2
      common/crate/rate(50),kcode(25),inx1(5,25),inx2(5,25),cdat(5,25),  /crate/.2
     1 reactsk(5,10,100),nsrt(5),sarea(5),nparam(5),xkinet(5),xmax,time, /crate/.3
     2 gc,excode,tpcode,kinr,xdiag(5),shift(4,100),shftsol(100)          /crate/.4
c                                                                        chmtrns.280
      data redox/5Hredox/                                                chmtrns.281
      data xtemp/5Htempx/                                                chmtrns.282
      data xkinet/5Hdissg,5Hexkin,5Htpkin,5Hprepc,5Hpreps/               chmtrns.283
      data blank/5Hblank/                                                chmtrns.284
      data rel/1.e-02/,absv/1.e-12/                                      chmtrns.285
      call link(" unit5=(input,open,text),read5,unit6=(output,create,hc) chmtrns.286
     x,print6 //")                                                       chmtrns.287
c                                                                        chmtrns.288
c                                                                        chmtrns.289
      mdim=25                                                            chmtrns.290
      ipart=0                                                            chmtrns.291
      kinr=0                                                             chmtrns.292
      ktsum=0                                                            chmtrns.293
      oxcode=0.0                                                         chmtrns.294
      tpcode=0.0                                                         chmtrns.295
      excode=0.0                                                         chmtrns.296
c  **************************                                            chmtrns.297
```

```
c  read in input parameters                                             chmtrns.298
c  **************************                                           chmtrns.299
      read(5,1000) title                                                chmtrns.300
         write(6,999) title                                             chmtrns.301
      read 1010, ulength,utime                                          chmtrns.302
         write (6,1011) ulength,utime                                   chmtrns.303
      read 910,itype,idynam,ig,ipart                                    chmtrns.304
      read 910,in1,in2,in3                                              chmtrns.305
c     read parameters of the numerical calculations                     chmtrns.306
      read 918, climit,errsqr,iterj,iterm,iterd,dtmult,ndiv             chmtrns.307
      read 920, nmax,xmax,gc,rw                                         chmtrns.308
      read 920, kmax,dtini,dtmax                                        chmtrns.309
      read 900, (tprint(k),k=1,kmax)                                    chmtrns.310
      read 900, doa,dob,vo                                              chmtrns.311
      read 910, ibnd(1),ibnd(2)                                         chmtrns.312
      read 910, ibndt(1),ibndt(2)                                       chmtrns.313
      read 910, itemp                                                   chmtrns.314
      if(itemp.eq.0) read 900,temp                                      chmtrns.315
      if(itemp.eq.1) read 900,temp,tempi,hcw,hcm,tcm                    chmtrns.316
      if(itemp.eq.2) read 900,tempi,tempo                               chmtrns.317
      read 910, leach, nb                                               chmtrns.318
c      kinetic rate information                                         chmtrns.319
    1 read 510,word                                                     chmtrns.320
      if(word.eq.redox) go to 9                                         chmtrns.321
      if( word.eq.xkinet(2) ) go to 11                                  chmtrns.322
      if( word.eq.xkinet(3) ) go to 12                                  chmtrns.323
      if( word.eq.blank ) go to 3                                       chmtrns.324
c                                                                       chmtrns.325
    9 write(6,3000)                                                     chmtrns.326
      oxcode=1.0                                                        chmtrns.327
      go to 13                                                          chmtrns.328
   11 write(6,3200)                                                     chmtrns.329
      excode=1.0                                                        chmtrns.330
      go to 2                                                           chmtrns.331
   12 write(6,3300)                                                     chmtrns.332
      tpcode=1.0                                                        chmtrns.333
      go to 2                                                           chmtrns.334
c        read redox constants R and F                                   chmtrns.335
   13 read 900, gascon, farcon                                          chmtrns.336
      go to 1                                                           chmtrns.337
c                                                                       chmtrns.338
c     read binary components of the rate equations and constants        chmtrns.339
c     Here,the order for the solids with reversible reactions must      chmtrns.340
c     be the same as those in the input list. Also, parameters for      chmtrns.341
c     the Glass dissolution option are the last ones to be read.        chmtrns.342
c     The position of H+ in the list of species must be number 2 if     chmtrns.343
c     Glass dissolution option is invoked.                              chmtrns.344
    2 kinr=kinr+1                                                       chmtrns.345
      i=kinr                                                            chmtrns.346
      read 915,nparam(i),sarea(i),nsrt(i),xdiag(i)                      chmtrns.347
      if(nparam(i).ne.0) read 916,(cdat(i,j),j=1,nparam(i))             chmtrns.348
      if(nsrt(i).eq.0) go to 1                                          chmtrns.349
      read 917, (inx1(i,j),inx2(i,j), j=1,nsrt(i) )                     chmtrns.350
      do 14 j=1,nsrt(i)                                                 chmtrns.351
      l=j+ktsum                                                         chmtrns.352
      read 900, (conk(l,k), k=1,4)                                      chmtrns.353
   14 continue                                                          chmtrns.354
      ktsum=ktsum+nsrt(i)                                               chmtrns.355
c                                                                       chmtrns.356
      go to 1                                                           chmtrns.357
```

```
c                                                                       chmtrns.358
    3 continue                                                          chmtrns.359
c     read C-13 partition parameters                                    chmtrns.360
      if(ipart.eq.0.0) go to 65                                         chmtrns.361
      read 900, shiftr, shifts                                          chmtrns.362
      do 66 j=1,4                                                       chmtrns.363
      read 900, (conp(j,k), k=1,2)                                      chmtrns.364
   66 continue                                                          chmtrns.365
   65 continue                                                          chmtrns.366
      read 910, (num(i),i=1,5), lnh, lno                                chmtrns.367
         num(10)=num(1)+num(4)                                          chmtrns.368
         num(11)=num(10)+num(5)                                         chmtrns.369
         if (num(2).eq.0) cecns=0.                                      chmtrns.370
      if (num(2).gt.0) read 900, cecns                                  chmtrns.371
         if (num(3).eq.0) soh=0.                                        chmtrns.372
         if (num(3).eq.0) go to 5                                       chmtrns.373
      read 900, c1, c2, area, soh                                       chmtrns.374
c     read exchange surface site dissociation information               chmtrns.375
c     to be used for aqueous species surface complexation               chmtrns.376
      read 900, ( con((num(2)+1),j), j=1,4 )                            chmtrns.377
      read 900, ( con((num(2)+2),j), j=1,4 )                            chmtrns.378
         num(3)=num(3)+2                                                chmtrns.379
         dum(num(11)+1)=2hso                                            chmtrns.380
         dum(num(11)+2)=4hsoh2                                          chmtrns.381
         ilble(1)=num(11)+1                                             chmtrns.382
         ilble(2)=num(11)+2                                             chmtrns.383
    5 continue                                                          chmtrns.384
         num(6)=num(1)+num(2)                                           chmtrns.385
         num(7)=num(6)+num(3)                                           chmtrns.386
         num(8)=num(7)+num(4)                                           chmtrns.387
         num(9)=num(8)+num(5)                                           chmtrns.388
         num(12)=num(2)+num(3)                                          chmtrns.389
         num(14)=nmax                                                   chmtrns.390
         num(15)=num(12)+num(4)+num(5)                                  chmtrns.391
         ntotp=num(9)                                                   chmtrns.392
         leach1(1)=leach                                                chmtrns.393
         if (leach1(1).ne.0) go to 7                                    chmtrns.394
c                                                                       chmtrns.395
      do 6 i=2,5                                                        chmtrns.396
    6 leach1(i)=0                                                       chmtrns.397
c                                                                       chmtrns.398
    7 if (lnh.eq.0) go to 8                                             chmtrns.399
      s0=0                                                              chmtrns.400
      s1=1                                                              chmtrns.401
    8 continue                                                          chmtrns.402
c  ************************************                                 chmtrns.403
c  write out some of the constants used                                 chmtrns.404
c  ************************************                                 chmtrns.405
      if ((ibnd(1).eq.0).and.(ibnd(2).eq.1)) write(6,2060)              chmtrns.406
      if ((ibnd(1).eq.1).and.(ibnd(2).eq.1)) write(6,2061)              chmtrns.407
      if ((ibnd(1).eq.0).and.(ibnd(2).eq.0)) write(6,2062)              chmtrns.408
      if ((ibnd(1).eq.1).and.(ibnd(2).eq.0)) write(6,2063)              chmtrns.409
      if (itype.eq.0) write(6,600) doa,dob,vo,nmax                      chmtrns.410
      if (itype.eq.1) write(6,601) doa,dob,vo,nmax                      chmtrns.411
      write(6,610) kmax,(tprint(i),i=1,kmax)                            chmtrns.412
      write(6,612) climit,errsqr,iterj,iterm,iterd,dtmult,ndiv          chmtrns.413
      write(6,611) cecns,soh                                            chmtrns.414
      if(itemp.eq.0) write (6,2065) temp                                chmtrns.415
      if(itemp.eq.1) write (6,2067) temp,hcw,hcm,tcm                    chmtrns.416
      if((itemp.eq.1).and.(ibndt(1).eq.0)) write (6,2068) tempi         chmtrns.417
```

```
      if((itemp.eq.1).and.(ibndt(1).eq.1)) write (6,2069) tempi
      if(itemp.eq.2) write (6,2070) tempi,tempo
      if(itemp.eq.1) write (6,2073)
         ii=0
         ie=3
         li=0
         lb=0
c  ******************************************************
c  read in information on bare ions and type of sorption
c  ******************************************************
c     Note:if Glass dissolution option is activated use 1,2,3,4, and 5
c          for kcode(j) of Na, H, K, Ca, and Mg, respectively, regardles
c          of the order of appearence of these species in the input list
c       H+ must be at position number 2 in the input for Glass option.
c     Also, if C-13 partition is included use 31 and 32 for kcode(j)
c     of CO3 and neg.KI respectively.
         m1=num(1)
         nspecm=num(1)
         if (lnh.gt.0) nspecm=num(1)-1
c
      do 17 j=1,nspecm
      read 950, dum(j),vj(j),kcode(j),indexi(j),bi(j),guessi(j),
     1 indexb(j),bc(j),guessb(j),isorpi(j),isorpe(j)
         if (indexi(j).eq.4) li=j
         if (indexb(j).eq.4) lb=j
c
c  if the species is sorbed via ion exchange, read in equilibrium consta
c  and guesses if they are being provided
c
         if (isorpi(j).eq.0) go to 15
         ii=ii+1
         ilbli(ii)=j
         k=ii+num(1)
      read 960, guessi(k),guessb(k)
      read 900, (con(ii,ji), ji=1,4)
c
c  if the species is sorbed via surface complexation, read in equilibriu
c  constant and number of h ions sorbed or released from the surface, an
c  guesses if being provided
c
   15    if (isorpe(j).eq.0) go to 17
         if (j.eq.lnh) iej=3
         if (j.eq.lnh) go to 16
         ie=ie+1
         iej=ie
   16    continue
         ilble(iej)=j
         k=iej+num(6)
      read 960, guessi(k),guessb(k),ah(iej)
         iejn=num(2)+iej
      read 900, (con(iejn,ji), ji=1,4)
   17 continue
c
c  ************************************************
c  write out the conditons on the basis species
c  ************************************************
       iie=ie-2
       if (num(3).eq.0) iie=0
      write (6,800) nspecm,ii,iie
        iion=ii
```

```
       ieion=ie                                                         chmtrns.478
       ii=0                                                             chmtrns.479
       ie=3                                                             chmtrns.480
c                                                                       chmtrns.481
      do 20 j=1,nspecm                                                  chmtrns.482
       if (isorpi(j).eq.1) ii=ii+1                                      chmtrns.483
       if (j.eq.lnh.and.isorpe(j).eq.1) iek=3                           chmtrns.484
       if (j.eq.lnh.and.isorpe(j).eq.1) go to 19                        chmtrns.485
       if (isorpe(j).eq.1) ie=ie+1                                      chmtrns.486
       iek=ie                                                           chmtrns.487
   19 continue                                                          chmtrns.488
      write(6,805) dum(j),vj(j),kcode(j),indexi(j),bi(j),indexb(j),bc(j) chmtrns.489
   20 continue                                                          chmtrns.490
c                                                                       chmtrns.491
       ncmplxp=num(4)                                                   chmtrns.492
c  ****************************************************************     chmtrns.493
c  if water dissociation is considered, need to read and write out      chmtrns.494
c  information about the oh ion                                         chmtrns.495
c  ****************************************************************     chmtrns.496
       if (lnh.eq.0) go to 21                                           chmtrns.497
      read 974, dum(num(1)),(h2oc(j), j=1,4)                            chmtrns.498
       vj(num(1))=-1.0                                                  chmtrns.499
       indexi(num(1))=1                                                 chmtrns.500
       indexb(num(1))=1                                                 chmtrns.501
      read 980,guessi(num(1)),guessb(num(1))                            chmtrns.502
      isorpi(num(1))=0                                                  chmtrns.503
      isorpe(num(1))=0                                                  chmtrns.504
       ncmplxp=num(4)+1                                                 chmtrns.505
       bi(num(1))=guessi(num(1))                                        chmtrns.506
       bc(num(1))=guessb(num(1))                                        chmtrns.507
   21 if (num(4).eq.0) go to 35                                         chmtrns.508
c  *****************************************************                chmtrns.509
c  read in information on complexes and type of sorption                chmtrns.510
c  *****************************************************                chmtrns.511
c  NOTE: If C-13 fractionation is included use 33 and 34 for            chmtrns.512
c        KCODE(knu) HCO3- and H2CO3 respectively.                       chmtrns.513
       m4=num(4)                                                        chmtrns.514
c                                                                       chmtrns.515
      do 27 i=1,m4                                                      chmtrns.516
       kk=i+num(1)                                                      chmtrns.517
       knu=num(7)+i                                                     chmtrns.518
      read 972, dum(kk), kcode(knu), (s(j,i),j=1,m1)                    chmtrns.519
         in=num(12)+i                                                   chmtrns.520
      read 900, (con(in,j), j=1,4)                                      chmtrns.521
       vj(kk)=0.                                                        chmtrns.522
         do 22 j=1,m1                                                   chmtrns.523
   22    vj(kk)= vj(kk)+(s(j,i)*vj(j))                                  chmtrns.524
       ic=kk+num(12)                                                    chmtrns.525
      read 980, guessi(ic),guessb(ic),isorpi(kk),isorpe(kk)             chmtrns.526
c                                                                       chmtrns.527
c  if complex is sorbed via ion exchange need to read in equilibrium con chmtrns.528
c                                                                       chmtrns.529
       if (isorpi(kk).eq.0) go to 25                                    chmtrns.530
       ii=ii+1                                                          chmtrns.531
       k=ii+num(1)                                                      chmtrns.532
      read 960, guessi(k),guessb(k)                                     chmtrns.533
       ilble(ii)=ic                                                     chmtrns.534
      read 900, (con(ii,j), j=1,4)                                      chmtrns.535
c                                                                       chmtrns.536
c  if complex is sorbed via surface complexation need to read in equilib chmtrns.537
```

```
c  constant and number of h ions sorbed or released by the surface        chmtrns.538
c                                                                         chmtrns.539
   25  if (isorpe(kk).eq.0) go to 27                                      chmtrns.540
       ie=ie+1                                                            chmtrns.541
       ilble(ie)=ic                                                       chmtrns.542
       k=ie+num(6)                                                        chmtrns.543
      read 960, guessi(k),guessb(k),ah(ie)                                chmtrns.544
         ien=num(2)+ie                                                    chmtrns.545
      read 900, (con(ien,j), j=1,4)                                       chmtrns.546
   27 continue                                                            chmtrns.547
c                                                                         chmtrns.548
c                                                                         chmtrns.549
c  write out information about the complexes                              chmtrns.550
c                                                                         chmtrns.551
       icmplx=ii-iion                                                     chmtrns.552
       iecmplx=ie-ieion                                                   chmtrns.553
      write (6,802) ncmplxp,icmplx,iecmplx                                chmtrns.554
       if (ii.ne.num(2)) write (6,990) ii,num(2),ie,num(3)                chmtrns.555
       ii=iion                                                            chmtrns.556
       ie =ieion                                                          chmtrns.557
      if(lnh.gt.0) kcode(num(1))=0                                        chmtrns.558
      if(lnh.gt.0) write(6,820) dum(num(1)),vj(num(1)),kcode(num(1)),     chmtrns.559
     1 (s0, j=1,nspecm), s1                                               chmtrns.560
c                                                                         chmtrns.561
      do 30 i=1,m4                                                        chmtrns.562
       if (isorpi(i).eq.1) ii=ii+1                                        chmtrns.563
       if (isorpe(i).eq.1) ie=ie+1                                        chmtrns.564
       kk=i+num(1)                                                        chmtrns.565
       inp=num(12)+i                                                      chmtrns.566
      write(6,820) dum(kk),vj(kk),kcode(kk),(s(j,i),j=1,m1)               chmtrns.567
   30 continue                                                            chmtrns.568
c                                                                         chmtrns.569
   35 continue                                                            chmtrns.570
c  ***********************************                                    chmtrns.571
c  read in information on precipitates                                    chmtrns.572
c  ***********************************                                    chmtrns.573
       if (num(5).eq.0) go to 41                                          chmtrns.574
      write (6,840) num(5)                                                chmtrns.575
       m5=num(5)                                                          chmtrns.576
c                                                                         chmtrns.577
      do 40 i=1,m5                                                        chmtrns.578
       kk=num(10)+i                                                       chmtrns.579
      knu=num(8)+i                                                        chmtrns.580
       kl=kk+num(12)                                                      chmtrns.581
      read 972,dum(kk),kcode(knu),(ss(j,i),j=1,m1)                        chmtrns.582
       inun=num(12)+num(4)+i                                              chmtrns.583
      read 900, (con(inun,j), j=1,4)                                      chmtrns.584
      read 960,guessi(kl),guessb(kl)                                      chmtrns.585
c                                                                         chmtrns.586
c  if a solid is being placed in contact with the initial groundwater     chmtrns.587
c  need to determine the species making up this solid                     chmtrns.588
c                                                                         chmtrns.589
       if (i.ne.leach1(1)) go to 39                                       chmtrns.590
       ileach=1                                                           chmtrns.591
         do 38 j=1,m1                                                     chmtrns.592
         if (ss(j,i).eq.0) go to 38                                       chmtrns.593
         ileach=ileach+1                                                  chmtrns.594
         leach1(ileach)=j                                                 chmtrns.595
   38    continue                                                         chmtrns.596
c  ********************************************                           chmtrns.597
```

```
c  write out information about the precipitates                          chmtrns.598
c  ********************************************                          chmtrns.599
   39 write(6,971) dum(kk),kcode(knu),                                   chmtrns.600
     1             (ss(j,i),j=1,m1)                                      chmtrns.601
   40 continue                                                           chmtrns.602
   41 continue                                                           chmtrns.603
      m15=num(15)                                                        chmtrns.604
      delt=dtini                                                         chmtrns.605
      ntot=num(8)                                                        chmtrns.606
      neq=num(9)*nmax                                                    chmtrns.607
c     *********************************************************          chmtrns.608
c     write out kinetic rate information,and other related data          chmtrns.609
c     *********************************************************          chmtrns.610
      if( kinr.eq.0 ) go to 43                                           chmtrns.611
      ktsum=0                                                            chmtrns.612
      do 42 i=1,kinr                                                     chmtrns.613
      if( xdiag(i).eq.xkinet(4) ) write(6,901)                           chmtrns.614
      if( xdiag(i).eq.xkinet(5) ) write(6,902)                           chmtrns.615
      if( xdiag(i).eq.xkinet(1) ) write(6,903)                           chmtrns.616
      print 976, nsrt(i),nparam(i),sarea(i),xdiag(i)                     chmtrns.617
      if(nparam(i).ne.0) print 977,(cdat(i,j),j=1,nparam(i))             chmtrns.618
      if(nsrt(i).eq.0) go to 42                                          chmtrns.619
      print 919, (inx1(i,j),inx2(i,j),j=1,nsrt(i) )                      chmtrns.620
      write(6,905)                                                       chmtrns.621
      do 44 j=1,nsrt(i)                                                  chmtrns.622
      l=j+ktsum                                                          chmtrns.623
      print 900,(conk(l,k), k=1,4)                                       chmtrns.624
   44 continue                                                           chmtrns.625
      ktsum=ktsum+nsrt(i)                                                chmtrns.626
   42 continue                                                           chmtrns.627
   43 continue                                                           chmtrns.628
c     *******************************************************            chmtrns.629
c     print of C-13 partition parameters                                 chmtrns.630
c     *******************************************************            chmtrns.631
      if(ipart.eq.0) go to 68                                            chmtrns.632
      write(6,615)                                                       chmtrns.633
      write(6,616) shiftr,shifts                                         chmtrns.634
      write(6,905)                                                       chmtrns.635
      do 67 j=1,4                                                        chmtrns.636
      print 900, (conp(j,k), k=1,2)                                      chmtrns.637
   67 continue                                                           chmtrns.638
   68 continue                                                           chmtrns.639
c     **************************************************************     chmtrns.640
c      write out redox constants                                         chmtrns.641
c     **************************************************************     chmtrns.642
      if(oxcode.eq.1.0) print 978, gascon, farcon                        chmtrns.643
c  ************                                                          chmtrns.644
c  set up grid                                                           chmtrns.645
c  ************                                                          chmtrns.646
      call setup(xmax,nmax,itype,gc,rw)                                  chmtrns.647
      index=0                                                            chmtrns.648
c  ******************************                                        chmtrns.649
c  equilibrate initial conditions                                        chmtrns.650
c  ******************************                                        chmtrns.651
c                                                                        chmtrns.652
c     assign initial temp. to nodes                                      chmtrns.653
c                                                                        chmtrns.654
      if(itemp.lt.2) go to 46                                            chmtrns.655
      call gradt(nmax,tempi,tempo,itype,x,tmp)                           chmtrns.656
                                                                         chmtrns.657
```

```
      go to 47
   46 do 45 n=1,nmax
   45 tmp(n)=temp
      if(itemp.eq.1) tmp(1)=tempi
      if(itemp.eq.0) tb=temp
      if(itemp.eq.1) tb=tempi
   47 continue
      n2=2
      if(itemp.eq.2) n2=nmax
      call eqcon(1,n2,con,conk,h2oc,m15,kinr,lnh,tmp,h2ok,eqconst,
     1          nsrt,reactsk,alfa,conp,ipart,0 )
      m9=num(9)
c
c  determine here whether the program or the user is
c  providing the first guesses
c
      do 165 n=2,n2
      if(ig.eq.1) call iguess(com,bi,num,n,lnh)
      do 50 j=1,m9
      ac(j,n)=1.0
      if(ig.eq.1) cm(j)=com(j)
      if (ig.eq.1) go to 50
      cm(j)=guessi(j)
   50 continue
      if ((num(3).eq.0).or.(ig.eq.1) ) go to 51
      nn2=num(2)+2
      nn1=num(2)+1
      cm(num(6)+1)=eqconst(nn2,n)*soh/cm(lnh)
      cm(num(6)+2)=cm(lnh)*soh/eqconst(nn1,n)
      cm(num(6)+2)=cm(num(6)+2)/10.
   51 continue
c
      ctot=0.
c  *********************************************************************
c  determine the equilibrium concentrations for the initial conditions
c  *********************************************************************
      call eqlib(cm,bi,index1,indexi,num,n,ss,eqconst,ctot,ac,
     1li,lnh,in1)
c  ******************************
c  out initial conditions
      if(n.ne.2) go to 165
      if(n2.eq.nmax) write(6,765) n
      write(6,760) temp
      write (6,780)
      ii=0
      ie=0
      ileach=2
c
c  calcualte conventional kd's here, the total amount of the
c  species in the aqueous phase
c
      do 80 j=1,m1
      if( cm(j).eq.0.0 ) cm(j)=1.0e-30
      cmtotaq=cm(j)
      if (isorpi(j).eq.0) sorb=0.
      if (isorpi(j).eq.0) go to 55
      ii=ii+1
      sorb=cm(ii+num(1))
   55 continue
      if (isorpe(j).eq.0) sorbe=0.
```

```
      if (isorpe(j).eq.0) go to 56
      ie=ie+1
      sorbe=cm(ie+num(6)+3)
      if (j.eq.lnh) sorbe=cm(3+num(6))
      if (j.eq.lnh) ie=ie-1
   56 continue
      if (num(4).eq.0) go to 61
      do 60 i=1,m4
   60 cmtotaq=cmtotaq+s(j,i)*cm(num(7)+i)
      dk=  sorb/cmtotaq
      dke=sorbe/cmtotaq
c
c  if a solid is going to be placed in contact with the groundwater,
c  need to calculate how much of each species that makes up this
c  solid already exists in the groundwater
c
   61 if (leach1(1).eq.0) go to 79
      if (j.ne.leach1(ileach)) go to 70
      pini(ileach-1)=cmtotaq
      indexb(j)=3
      ileach=ileach+1
   70 continue
   79 if (indexi(j).eq.4) indexb(j)=2
      if (indexi(j).eq.4) bc(j)=cmtotaq
c
c  write out this information
c
   80 write (6,700) dum(j),cmtotaq,cm(j),sorb,sorbe,ac(j,2),dk,dke
c
      if (num(4).eq.0) go to 91
      write (6,740)
c
      do 90 i=1,m4
      kk=i+num(1)
      if (isorpi(kk).eq.0) sorb=0.
      if (isorpi(kk).eq.0) go to 85
      ii=ii+1
      sorb=cm(ii+num(1))
   85 continue
      if (isorpe(kk).eq.0) sorbe=0.
      if (isorpe(kk).eq.0) go to 86
      ie=ie+1
      sorbe=cm(ie+num(6)+3)
   86 continue
      ik=i+num(7)
c
c  write out information about the aqueous phase complexes
c
   90 write (6,750) dum(kk),vj(kk),cm(ik),sorb,sorbe
c
   91 continue
      do 95 j=1,m9
      cn((n-1)*num(9)+j)=cm(j)
   95 continue
c
      if (li.ne.0) go to 141
c
c  a species at the boundary may be calculated from a charge balance
c  therefore it is necessary to know the charge that
c  is not being considered in these calculations and which is being
```

```
c  made up by species not participating in the chemical changes          chmtrns.778
c                                                                        chmtrns.779
      ctot=0.                                                            chmtrns.780
c                                                                        chmtrns.781
      do 130 j=1,m1                                                      chmtrns.782
  130 ctot=cm(j)*vj(j)+ctot                                              chmtrns.783
c                                                                        chmtrns.784
      if (num(4).eq.0) go to 141                                         chmtrns.785
c                                                                        chmtrns.786
      do 140 i=1,m4                                                      chmtrns.787
  140 ctot=ctot+cm(num(7)+i)*vj(num(1)+i)                                chmtrns.788
c                                                                        chmtrns.789
  141 if (num(5).eq.0) go to 160                                         chmtrns.790
c                                                                        chmtrns.791
      write (6,841)                                                      chmtrns.792
c                                                                        chmtrns.793
      do 150 i=1,m5                                                      chmtrns.794
      ii=num(10)+i                                                       chmtrns.795
      write (6,845) dum(ii),cm(ii+num(12))                               chmtrns.796
c                                                                        chmtrns.797
c  if a solid is being placed in contact with the initial groundwater    chmtrns.798
c  need to calculate how much of each particular species in this         chmtrns.799
c  solid was initially in the groundwater; this calculation includes     chmtrns.800
c  the aqueous phase complexes                                           chmtrns.801
c                                                                        chmtrns.802
      if (leach1(1).eq.0) go to 150                                      chmtrns.803
      ileach=2                                                           chmtrns.804
c                                                                        chmtrns.805
      do 145 j=1,nspecm                                                  chmtrns.806
      if (ss(j,i).eq.0) go to 145                                        chmtrns.807
      if (j.ne.leach1(ileach)) go to 145                                 chmtrns.808
      pini(ileach-1)=pini(ileach-1)+ss(j,i)*cm(num(8)+i)                 chmtrns.809
      ileach=ileach+1                                                    chmtrns.810
  145 continue                                                           chmtrns.811
c                                                                        chmtrns.812
  150 continue                                                           chmtrns.813
  160 continue                                                           chmtrns.814
c                                                                        chmtrns.815
c  if a surface complexation model is used for sorption,                 chmtrns.816
c  then write out the charges developed at the layers and                chmtrns.817
c  the potentials between these layers                                   chmtrns.818
c                                                                        chmtrns.819
      if (num(3).eq.0) go to 161                                         chmtrns.820
      write (6,597) sigo,sigb,sigd,psio,psibeta,psid                     chmtrns.821
      write (6,598) cm(num(6)+1),cm(num(6)+2)                            chmtrns.822
  161 continue                                                           chmtrns.823
      ctot2 = -ctot                                                      chmtrns.824
      write (6,596) ctot2                                                chmtrns.825
  165 continue                                                           chmtrns.826
      if(n2.eq.nmax) go to 115                                           chmtrns.827
      do 110 n=1,nmax                                                    chmtrns.828
      do 110 j=1,m9                                                      chmtrns.829
      cn((n-1)*num(9)+j)=cm(j)                                           chmtrns.830
      ac(j,n)=ac(j,2)                                                    chmtrns.831
  110 continue                                                           chmtrns.832
  115 continue                                                           chmtrns.833
c     Note: The subroutine shiftb is only used for calculation of rechar chmtrns.834
c           water make up, in a pre-processing stage of C-13 partition   chmtrns.835
c           studies. In such run we only consider co2(g), H+ and OH- as  chmtrns.836
c           basis species and H2CO3, HCO3-, and CO3-- as complexes, in   chmtrns.837
```

```
c           this order.                                                 chmtrns.838
c                                                                       chmtrns.839
      if(idynam.eq.0.and.ipart.eq.1) call shiftb(cm,num,alfa,shiftr)    chmtrns.840
c  ********************************                                     chmtrns.841
c  equilibrate boundary conditions                                      chmtrns.842
c  ********************************                                     chmtrns.843
c                                                                       chmtrns.844
c  decide if the program is supplying the first guesses                 chmtrns.845
c  for the species concentrations                                       chmtrns.846
c                                                                       chmtrns.847
      if(ig.eq.1) call iguess(com,bc,num,n,lnh)                         chmtrns.848
      do 170 j=1,m9                                                     chmtrns.849
      ac(j,1)=1.0                                                       chmtrns.850
      if(ig.eq.1) cm(j)=com(j)                                          chmtrns.851
      if (ig.eq.1) go to  170                                           chmtrns.852
      cm(j)=guessb(j)                                                   chmtrns.853
  170 continue                                                          chmtrns.854
      if ((num(3).eq.0).or.(ig.eq.1)) go to 171                         chmtrns.855
      nn2=num(2)+2                                                      chmtrns.856
      nn1=num(2)+1                                                      chmtrns.857
      cm(num(6)+1)=eqconst(nn2,1)*soh/cm(lnh)                           chmtrns.858
      cm(num(6)+2)=cm(lnh)*soh/eqconst(nn1,1)                           chmtrns.859
      cm(num(6)+2)=cm(num(6)+2)/1000.                                   chmtrns.860
  171 continue                                                          chmtrns.861
c  ******************************************************************   chmtrns.862
c  determine the equilibrium distribution of species for the boundary   chmtrns.863
c  or influx conditions                                                 chmtrns.864
c  *******************************************************************  chmtrns.865
      call eqlib(cm,bc,index1,indexb,num,1,ss,eqconst,ctot,ac,          chmtrns.866
     1lb,lnh,in2)                                                       chmtrns.867
      tbound=tb                                                         chmtrns.868
      if(itemp.eq.2) tbound=tmp(1)                                      chmtrns.869
      write (6,790) tbound                                              chmtrns.870
      write (6,780)                                                     chmtrns.871
      ii=0                                                              chmtrns.872
      ie=0                                                              chmtrns.873
      nend=(nmax-1)*num(9)                                              chmtrns.874
c  ******************************************                           chmtrns.875
c  write out the boundary or influx conditions                          chmtrns.876
c  ******************************************                           chmtrns.877
c                                                                       chmtrns.878
c  also calculate conventional kd's, and the total aqueous              chmtrns.879
c  phase concentration of the species                                   chmtrns.880
c                                                                       chmtrns.881
      do 180 j=1,m1                                                     chmtrns.882
      if (ibnd(2).eq.0) bc2(j)=cn(nend+j)                               chmtrns.883
      if(cm(j).eq.0.0) cm(j)=1.0e-35                                    chmtrns.884
      cmtotaq=cm(j)                                                     chmtrns.885
      if (isorpi(j).eq.0) sorb=0.                                       chmtrns.886
      if (isorpi(j).eq.0) go to 172                                     chmtrns.887
      ii=ii+1                                                           chmtrns.888
      sorb=cm(ii+num(1))                                                chmtrns.889
  172 continue                                                          chmtrns.890
      if (isorpe(j).eq.0) sorbe=0.                                      chmtrns.891
      if (isorpe(j).eq.0) go to 173                                     chmtrns.892
      ie=ie+1                                                           chmtrns.893
      sorbe=cm(ie+num(6)+3)                                             chmtrns.894
      if (j.eq.lnh) sorbe=cm(num(6)+3)                                  chmtrns.895
      if (j.eq.lnh) ie=ie-1                                             chmtrns.896
  173 continue                                                          chmtrns.897
```

```
      if (num(4).eq.0) go to 176
      do 175 i=1,m4
  175 cmtotaq=cmtotaq+s(j,i)*cm(num(7)+i)
      dk=  sorb/cmtotaq
      dke=sorbe/cmtotaq
  176 bbc=cmtotaq
      if (num(5).eq.0) go to 179
      do 177 i=1,m5
      bbc=bbc+ss(j,i)*cm(num(8)+i)
  177 continue
c  *******************************************
c  set inner boundary condition
c  *******************************************
c
c  set up the actual boundary conditions for the inner
c  boundary
c
  179 if (ibnd(1).eq.0) bc(j)=bbc
      if (ibnd(1).eq.1 ) bc(j)=vo*cmtotaq
      if((ibnd(1).eq.1).and.(itype.eq.1)) bc(j)=vo*cmtotaq/x(1)
      index2(j)=0
  180 write (6,700) dum(j),cmtotaq,cm(j),sorb,sorbe,ac(j,1),dk,dke
c
c  do calculations for the complexes
c
      if (num(4).eq.0) go to 191
      write (6,740)
c
      do 190 i=1,m4
      kk=i+num(1)
      if (isorpi(kk).eq.0) sorb=0.
      if (isorpi(kk).eq.0) go to 192
      ii=ii+1
      sorb=cm(ii+num(1))
  192 continue
      if (isorpe(kk).eq.0) sorbe=0
      if (isorpe(kk).eq.0) go to 193
      ie=ie+1
      sorbe=cm(num(6)+ie+3)
  193 continue
      ik=i+num(7)
  190 write (6,750) dum(kk),vj(kk),cm(ik),sorb,sorbe
c
  191 continue
c  ******************************************************
c  if flux condtion is used, then first gird is the same
c  as the rest of the grids
c  *******************************************************
      if(ibndt(1).eq.1) tmp(1)=temp
      if (ibnd(1).eq.1) go to 201
      n=1
c
      do 200 j=1,m9
      cn((n-1)*num(9)+j)=cm(j)
  200 continue
c
  201  continue
c
c  write out information about the precipitates
c
```

```
      if (num(5).eq.0) go to 210                                        chmtrns.958
      write (6,841)                                                     chmtrns.959
c                                                                       chmtrns.960
      do 205 i=1,m5                                                     chmtrns.961
      ii=num(10)+i                                                      chmtrns.962
      write (6,845) dum(ii),cm(ii+num(12))                              chmtrns.963
  205 continue                                                          chmtrns.964
c                                                                       chmtrns.965
  210 continue                                                          chmtrns.966
      ctot1=0.                                                          chmtrns.967
c                                                                       chmtrns.968
c   do a check of the charge balance                                    chmtrns.969
c                                                                       chmtrns.970
      do 215 j=1,m1                                                     chmtrns.971
  215 ctot1=cm(j)*vj(j)+ctot1                                           chmtrns.972
c                                                                       chmtrns.973
      if (num(4).eq.0) go to 221                                        chmtrns.974
c                                                                       chmtrns.975
      do 220 i=1,m4                                                     chmtrns.976
  220 ctot1=ctot1+cm(num(7)+i)*vj(num(1)+i)                             chmtrns.977
c                                                                       chmtrns.978
  221 continue                                                          chmtrns.979
      ctot2 = -ctot1                                                    chmtrns.980
      write (6,596) ctot2                                               chmtrns.981
c                                                                       chmtrns.982
c  if surface complexation is used for sorption,  write                 chmtrns.983
c  out the charges developed at the surfaces and the                    chmtrns.984
c  potentials between the surfaces for the boundary conditions          chmtrns.985
c                                                                       chmtrns.986
      if (num(3).eq.0) go to 231                                        chmtrns.987
      write (6,597) sigo,sigb,sigd,psio,psibeta,psid                    chmtrns.988
      write (6,598) cm(num(6)+1),cm(num(6)+2)                           chmtrns.989
  231 continue                                                          chmtrns.990
      time=0.                                                           chmtrns.991
      k1=1                                                              chmtrns.992
      delt=dtini                                                        chmtrns.993
      if (idynam.eq.0) go to 500                                        chmtrns.994
c  ************************************************************         chmtrns.995
c  calculate new values of the concentration as a function of time      chmtrns.996
c  ************************************************************         chmtrns.997
c                                                                       chmtrns.998
c  set old values of concentration = to new values                      chmtrns.999
c                                                                       chmtrns.1000
      iijj=0                                                            chmtrns.1001
      call second(ts)                                                   chmtrns.1002
  235 do 240 n=1,nmax                                                   chmtrns.1003
      tmpold(n)=tmp(n)                                                  chmtrns.1004
      do 240 j=1,m9                                                     chmtrns.1005
  240 co((n-1)*num(9)+j)=cn((n-1)*num(9)+j)                             chmtrns.1006
c                                                                       chmtrns.1007
      time=time+delt                                                    chmtrns.1008
      iijj=iijj+1                                                       chmtrns.1009
      iter3=0                                                           chmtrns.1010
c                                                                       chmtrns.1011
c     calculate temperatures                                            chmtrns.1012
c                                                                       chmtrns.1013
  242 if(iter3.gt.ndiv) go to 500                                       chmtrns.1014
      if(iijj.eq.1.and.itemp.eq.0) go to 245                            chmtrns.1015
      if(itemp.ne.1) go to 250                                          chmtrns.1016
      call tmpcal(ibndt(1),nmax,hcm,hcw,vo,tcm,doa,itype,tb,delt,       chmtrns.1017
```

```
     1            x,tmpold,tmp )
  245 continue
c
c     calculate equilibrium constants
c
      call eqcon(1,nmax,con,conk,h2oc,m15,kinr,lnh,tmp,h2ok,eqconst,
     1          nsrt,reactsk,alfa,conp,ipart,1)
  250 iter=0
      iter2=0
c
c  calculate the residues of the equations for the new guess of
c  concentrations at this time
c
      call res(cn,co,bc,bc2,r,num,mdim,lnh,ibnd,itype,0)
c
c  calculate the jacobian
c
  260 call jacobi(a,b,c,co,cn,bc,bc2,num,r,mdim,lnh,ibnd,itype,
     1            tpcode,kcode )
c  the subroutines decbt and solbt invert the jacobian
c
  263 call decbt(num(9),nmax,mdim,a,b,c,iip,ier)
  270 continue
      iter=iter+1
      print 915,iter
      call solbt(num(9),nmax,mdim,a,b,c,r,iip)
c
      do 280 i=1,m9
  280 cmax(i)=0.
c
c  *****************************************************************
c  determine the new value of the concentrations from the old values
c  and the inversion of the jacobian
c  *****************************************************************
      do 291 n=1,nmax
      do 290 j=1,m9
      cold=cn((n-1)*num(9)+j)
      cn((n-1)*num(9)+j)=cn((n-1)*num(9)+j)-r(j,n)
      if(j.eq.lno) eh(n)=-alog10(cn((n-1)*num(9)+j))/( farcon/
     1                 (2.3*gascon*(tmp(n)+273.15)) )
     if (cn((n-1)*num(9)+j).lt.0.) cn((n-1)*num(9)+j)=cold/10.
      npnsne=num(7)+1
      if((j.gt.num(6)).and.(j.lt.npnsne).and.
     1(cn((n-1)*num(9)+j).gt.soh)) cn((n-1)*num(9)+j)=soh
      cmax(j)=amax1(cn((n-1)*num(9)+j),cmax(j)  )
  290 continue
  291 continue
c
c  *******************************************************************
c  determine if the changes in the concentrations are within the error
c  range
c  *******************************************************************
      err=0.
c
      do 300 n=1,nmax
      do 300 i=1,m9
      aaa=r(i,n)
      bbb=cmax(i)
      if(bbb.lt.climit) go to 300
      aaa=aaa/bbb
```

```
      err=err + aaa*aaa
  300 continue
      err=sqrt(err)
c
      if(err.lt.errsqr) go to 310
c  *********************************************************************
c  if convergence is not obtained calcualte the residues for the next gu
c  *********************************************************************
      call res(cn,co,bc,bc2,r,num,mdim,lnh,ibnd,itype,0)
      if(iter.lt.iterj) go to 270
  301 continue
      iter2=iter2+1
      if(iter2.lt.iterd) go to 260
      iter3=iter3+1
c  *********************************************************************
c  if convergence is not obtained within 7 iterations reduce the time st
c  *********************************************************************
      time=time-delt
      delt=delt/2.0
      write (6,2022) delt
      k6=1
c
c  set new values equal to old values and start this time step over
c
      do 306 n=1,nmax
      tmp(n)=tmpold(n)
      do 306 j=1,m9
  306 cn((n-1)*num(9)+j)=co((n-1)*num(9)+j)
      time=time+delt
      go to 242
c
c  convergence is obtained, so start next time step
c
  310 continue
      if(ipart.ne.0) call shiftc(cn,num,shftsol,shift,shiftr,shifts,
     1              alfa,nmax,kcode )
c  *********************************************************************
c
      call second(te)
      deltc=te-ts
      if( (time-tprint(k1)).ge.0. ) go to 401
      if(iter.lt.iterm) delt=delt*dtmult
      if (delt.gt.dtmax) delt=dtmax
      if(iter.lt.iterm.and.delt.lt.dtmax) write(6,2023) delt,time
      write (6,2024) iter
      go to 235
  401 continue
      write(6,2080) deltc
      if(iter.lt.iterm.and.delt.lt.dtmax) write(6,2023) delt,time
c  *********************************************************
c  write out the species concentrations
c  *********************************************************
      k1=k1+1
      write (6,530) time,delt
      npri=nmax/9
      i1=1
      i5=1
      i4=9
      k2p=0
      i3=0
```

```
  405 continue                                                          chmtrns.1138
      k1p=k2p+1                                                         chmtrns.1139
      k2p=k1p+(i4-1)                                                    chmtrns.1140
      i2=i3                                                             chmtrns.1141
      i3=i2+num(9)*i4                                                   chmtrns.1142
      write (6,620) (x(k),k=k1p,k2p)                                    chmtrns.1143
c                                                                       chmtrns.1144
c  basis species                                                        chmtrns.1145
c                                                                       chmtrns.1146
      do 410 j=1,m1                                                     chmtrns.1147
      i22=i2+j                                                          chmtrns.1148
      write (6,625) dum(j),(cn(k),k=i22,i3,m9)                          chmtrns.1149
  410 continue                                                          chmtrns.1150
c                                                                       chmtrns.1151
      if(ipart.ne.0) write(6,630) (shftsol(k),k=k1p,k2p)                chmtrns.1152
      if(ipart.ne.0) write(6,631) (shift(1,k),k=k1p,k2p)                chmtrns.1153
      if(ipart.ne.0) write(6,632) (shift(2,k),k=k1p,k2p)                chmtrns.1154
      if(ipart.ne.0) write(6,633) (shift(3,k),k=k1p,k2p)                chmtrns.1155
  630 format('shftsol',6x,9(1pe12.4))                                   chmtrns.1156
  631 format('shift0' ,6x,9(1pe12.4))                                   chmtrns.1157
  632 format('shift1' ,6x,9(1pe12.4))                                   chmtrns.1158
  633 format('shift2' ,6x,9(1pe12.4))                                   chmtrns.1159
      if(lno.ne.0) write (6,625) dum(lno), (eh(k), k=k1p,k2p)           chmtrns.1160
      if(itemp.ne.0) write(6,625) xtemp,(tmp(k), k=k1p,k2p)             chmtrns.1161
c  sorbed species via ion exchange                                      chmtrns.1162
c                                                                       chmtrns.1163
      if (num(2).eq.0) go to 420                                        chmtrns.1164
      write (6,621)                                                     chmtrns.1165
      m2=num(2)                                                         chmtrns.1166
c                                                                       chmtrns.1167
      do 415 j=1,m2                                                     chmtrns.1168
      i22=i2+num(1)+j                                                   chmtrns.1169
      kk=ilbli(j)                                                       chmtrns.1170
      write (6,625) dum(kk),(cn(k),k=i22,i3,m9)                         chmtrns.1171
  415 continue                                                          chmtrns.1172
c                                                                       chmtrns.1173
c  sorbed species via surface complexation                              chmtrns.1174
c                                                                       chmtrns.1175
  420 if (num(3).eq.0) go to 435                                        chmtrns.1176
      write (6,622)                                                     chmtrns.1177
      m3=num(3)                                                         chmtrns.1178
c                                                                       chmtrns.1179
      do 430 j=1,m3                                                     chmtrns.1180
      i22=i2+num(6)+j                                                   chmtrns.1181
      kk=ilble(j)                                                       chmtrns.1182
      if (kk.gt.num(1)) kk=kk-num(2)-num(3)                             chmtrns.1183
      if ((j.eq.1).or.(j.eq.2)) kk=ilble(j)                             chmtrns.1184
      write (6,625) dum(kk),(cn(k),k=i22,i3,m9)                         chmtrns.1185
  430 continue                                                          chmtrns.1186
c                                                                       chmtrns.1187
c  aqueous phase complexes                                              chmtrns.1188
c                                                                       chmtrns.1189
  435 if (num(4).eq.0) go to 445                                        chmtrns.1190
      write(6,623)                                                      chmtrns.1191
c                                                                       chmtrns.1192
      do 440 i=1,m4                                                     chmtrns.1193
      i22=i2+num(7)+i                                                   chmtrns.1194
      kk=i+num(1)                                                       chmtrns.1195
      write (6,625) dum(kk),(cn(k),k=i22,i3,m9)                         chmtrns.1196
  440 continue                                                          chmtrns.1197
```

```
c                                                                       chmtrns.1198
c  precipitates                                                         chmtrns.1199
c                                                                       chmtrns.1200
  445 if (num(5).eq.0) go to 470                                        chmtrns.1201
      write (6,624)                                                     chmtrns.1202
c                                                                       chmtrns.1203
      do 460 i=1,m5                                                     chmtrns.1204
      kk=num(10)+i                                                      chmtrns.1205
      i22=i2+num(8)+i                                                   chmtrns.1206
      write (6,625) dum(kk),(cn(k),k=i22,i3,m9)                         chmtrns.1207
  460 continue                                                          chmtrns.1208
c                                                                       chmtrns.1209
  470 continue                                                          chmtrns.1210
      i1=i1+1                                                           chmtrns.1211
      if (i1.le.npri) go to 405                                         chmtrns.1212
      npri2=npri*9                                                      chmtrns.1213
      if (npri2.eq.nmax) go to 490                                      chmtrns.1214
      i4=nmax-npri2                                                     chmtrns.1215
      if (i5.eq.5) go to 490                                            chmtrns.1216
      i5=5                                                              chmtrns.1217
      go to 405                                                         chmtrns.1218
  490 if (k1.gt.kmax) go to 500                                         chmtrns.1219
      go to 235                                                         chmtrns.1220
  500 continue                                                          chmtrns.1221
      stop                                                              chmtrns.1222
c                                                                       chmtrns.1223
c  format statements for the main program                               chmtrns.1224
c                                                                       chmtrns.1225
  510 format(10x,a5)                                                    chmtrns.1226
  530 format ('1',4x,'time=',1pe12.5,10x,'delt=',1pe12.4,10x,           chmtrns.1227
     1'concentration of species in moles/liter solution',/)             chmtrns.1228
  596 format (//,4x,'to have a net balance of zero charge, other '      chmtrns.1229
     1'species must be present (although not participating in the',/,   chmtrns.1230
     2 4x, 'chemical reactions) with a total concentration times ',     chmtrns.1231
     2 'charge of ',g10.3,/)                                            chmtrns.1232
  597 format (///,5x,'sigo=',(1pe10.3),5x,'sigb=',(1pe10.3),5x,'sigd=', chmtrns.1233
     1 (1pe10.3),/,5x,'psio=',(1pe10.3),5x,'psibeta=',(1pe10.3),5x,     chmtrns.1234
     2'psid=',(1pe10.3),/)                                              chmtrns.1235
  598 format (4x,'so- =',1pe12.4,10x,'soh2+ =',1pe12.4,//)              chmtrns.1236
  600 format(//,4x,'dispersion coefficient=',f4.1,'v+'e10.2,5x,         chmtrns.1237
     1'fluid velocity =',g10.3,5x, 'number of grid points =',i5,/)      chmtrns.1238
  601 format(//,4x,'dispersion coefficient=',f4.1,'v+'e10.2,5x,         chmtrns.1239
     1'fluid velocity =',f10.3,'/ r',5x,                                chmtrns.1240
     2 'number of grid points =',i5,/)                                  chmtrns.1241
  610 format(/,4x,'number of print outs =',i5,/,4x,                     chmtrns.1242
     1 'given at the following times= ',8g10.4,/,34x,9g10.4)            chmtrns.1243
  611 format (/,4x,'number of sites(ion-exchange)',f10.4,10x,           chmtrns.1244
     1'number of sites(edl model)',f10.4)                               chmtrns.1245
  612 format(/,'errsqr= 'e10.2,5x,'climit= 'e10.2,5x,'iterj= 'i5,5x,    chmtrns.1246
     1 'iterm= 'i5,5x,'iterd= 'i5,5x,'dtmult= 'e10.2,5x,'ndiv= 'i5)     chmtrns.1247
  615 format(/,10x,'C-13 Fractionation parameters:')                    chmtrns.1248
  616 format(/,10x,'shiftr= 'f10.4,10x,'shifts= 'f10.4)                 chmtrns.1249
  620 format (//,4x,55x,'distance',/,4x,'species',1x,9f12.3,/)          chmtrns.1250
  622 format (/,4x,'sorbed(edl)',/)                                     chmtrns.1251
  623 format(/,4x,'complexes',/)                                        chmtrns.1252
  621 format(/,1x,'sorbed(ionex)',/)                                    chmtrns.1253
  624 format(/,2x,'precipitates',/)                                     chmtrns.1254
  625 format (3x,a8,2x,9(1pe12.4))                                      chmtrns.1255
  700 format(' 'a8,2g15.4,4x,2g11.4,6x,3g15.4)                          chmtrns.1256
  740 format(/' complex        valence       aqueous conc.   sorption(iochmtrns.1257
```

```
   1)      sorption(edl)')                                              chmtrns.1258
750 format(' 'a8,5g15.4)                                                chmtrns.1259
760 format('1',60x,'initial conditions',/,63x,'temp= ',f7.2             chmtrns.1260
   1 //)                                                                chmtrns.1261
765 format(60x,' for nodal point no ',i5,//)                            chmtrns.1262
780 format(' component',5x,'  total     ',5x,                           chmtrns.1263
   1'bare ion    ',5x,'sorbed concentration',8x,'act. coef.',5x,        chmtrns.1264
   2'kd(ion ex)        kd(edl)      ',/,13x,'aqueous conc.',4x,         chmtrns.1265
   3 'concentration',3x,'  (ionex)     (edl) ',/)                       chmtrns.1266
790 format('1',30x,'boundary or influx conditions'/,40x,                chmtrns.1267
   1 'temp= ',f7.2)                                                     chmtrns.1268
800 format (////,2x,i4,'       bare ions',10x,i4,5x,'sorbing ions',     chmtrns.1269
   1 ' (ionex)',10x,i4,5x,'sorbing ions (edl model)',///,               chmtrns.1270
   2 'component charge ',5x,'kinetic',                                  chmtrns.1271
   3 5x,'initial condition',7x,'boundary condition',/,                  chmtrns.1272
   4 25x,'code',7x,'type',7x,'value',9x,'type',7x,'value',/)            chmtrns.1273
802 format (///,2x,i4,'     complexes',10x,i4,7x,'sorbing complexes',   chmtrns.1274
   1'(ionex)',10x,i4,5x,'sorbing complexes (edl model)',///,            chmtrns.1275
   2'complex   charge  ',5x,'kcode',25x,'stoichiometry',/)              chmtrns.1276
805 format(a8,f6.0,6x,i5,6x,i5,4x,1pe12.4,4x,i5,4x,1pe12.4,10x,         chmtrns.1277
   1 )                                                                  chmtrns.1278
820 format(a8,f6.0,5x,i5,5x,10(0pf5.0) )                                chmtrns.1279
840 format (///,2x,i5,7x,'solid precipitates',//,2x,'precipitate',      chmtrns.1280
   1 2x,'kinetic code',10x,'stoichiometry',/)                           chmtrns.1281
841 format (//,2x,'precipitate',3x,'equivalent moles',                  chmtrns.1282
   1'/liter solution',/)                                                chmtrns.1283
845 format (1x,a8,g10.3)                                                chmtrns.1284
900 format (10x,7f10.3)                                                 chmtrns.1285
901 format(/,'Calcite dissolution information:',/)                      chmtrns.1286
902 format(/,'Silicate-water reaction information:',/)                  chmtrns.1287
903 format(/,'Glass dissolution information:',/)                        chmtrns.1288
905 format(/,'coefficients in log function of thermodynamic constants' chmtrns.1289
   1 )                                                                  chmtrns.1290
910 format (10x,7i10)                                                   chmtrns.1291
915 format(80x,/10x,i10,f10.3,i10,5x,a5)                                chmtrns.1292
916 format(10x,7e10.2)                                                  chmtrns.1293
917 format(10x,4(2i5))                                                  chmtrns.1294
919 format(/,'binary element nos. of the terms of the rate equation',   chmtrns.1295
   1          /,10x,4(2i5) )                                            chmtrns.1296
918 format(10x,2e10.3,3i10,e10.3,i10)                                   chmtrns.1297
920 format (10x,i10,6f10.3)                                             chmtrns.1298
950 format(a8,f5.0,2i5,2f10.3,i5,2f10.3,2i5)                            chmtrns.1299
960 format(10x,3f10.3,f5.1)                                             chmtrns.1300
970 format (a8,f10.3,10f5.1,/,20f5.1)                                   chmtrns.1301
971 format(a8,6x,i5,9x,10(0pf5.1) )                                     chmtrns.1302
972 format(a8,i5,10f5.1,/,20f5.1 )                                      chmtrns.1303
974 format(a8,7e10.3)                                                   chmtrns.1304
978 format(/,'redox constants',/,'gas constant R= ',e12.3,              chmtrns.1305
   1         10x,'Faraday constant F= ',e12.3 )                         chmtrns.1306
979 format(a8,10f5.1,/,20f5.1 )                                         chmtrns.1307
975 format(' binary print of species no. participating in each'/        chmtrns.1308
   1       ' term of rate formulla and the reaction constants ',//      chmtrns.1309
   2        4( 2i5,f10.3 )/)                                            chmtrns.1310
976 format(/' no of terms in the rate formulla = 'i5 ,//                chmtrns.1311
   1       ' no of other additional data needed = ' i5 ,//              chmtrns.1312
   2        ' representative surface parameter = 'e12.3,//              chmtrns.1313
   3        ' xdiag = 'a5,//)                                           chmtrns.1314
977 format( ' additional data for rate calculation',// 10e12.3 )        chmtrns.1315
980 format (10x,2f10.3,2i5)                                             chmtrns.1316
990 format (/,4x,'your number of sorbing ions input does not equal the chmtrns.1317
```

```
     1 number that was listed initially',2i5,10x,2i5)                   chmtrns.1318
  999 format ('1',20a4)                                                 chmtrns.1319
 2065 format(1x,'this simulation is isothermal, temperature =',f5.0)    chmtrns.1320
 2067 format(1x,'simulation of mixing with heat transport, with initial  chmtrns.1321
     1temperature =',f5.0,/,4x,'heat capacities of fluid and solid are', chmtrns.1322
     2g10.2,2x,'and',g10.2,/,4x,'thermal conductivity of solid',g10.2)   chmtrns.1323
 2068 format(1x,'inner boundary condition is constant temperature =',f5. chmtrns.1324
     10)                                                                chmtrns.1325
 2069 format(1x,'inner boundary condition is constant flux of fluid with chmtrns.1326
     1 temperature =',f5.0)                                             chmtrns.1327
 2070 format(1x,'fixed temperature gradient:'/,4x,f10.3,2x,'at inner bou chmtrns.1328
     1ndary',/,4x,f10.3,2x,'at outer boundary')                         chmtrns.1329
 2073 format(1x,'temperature is held at initial value at outer boundary' chmtrns.1330
     1)                                                                 chmtrns.1331
 1000 format (20a4)                                                     chmtrns.1332
 1001 format (10x,5(1pe12.4,/))                                         chmtrns.1333
 1010 format(10x,2a8)                                                   chmtrns.1334
 1011 format (/,4x,'unit of length is',2x,a8,10x,'unit of time is'      chmtrns.1335
     1,2x,a8)                                                           chmtrns.1336
 2000 format (/,4x,'the activity product is',1pe10.3,5x,                chmtrns.1337
     1'which now exceeds the solubility product of',1pe10.3)            chmtrns.1338
 2021 format (//,4x,'precipitation at node',i5,5x,'for species',a8,     chmtrns.1339
     1 'at time',1pe11.3)                                               chmtrns.1340
 2022 format (/,4x,'convergence is not obtained, so time step is '      chmtrns.1341
     1 , 'divided by 2,   delt=',1pe12.4)                               chmtrns.1342
 2023 format (/,4x,'convergence is obtained in less than 6 ',           chmtrns.1343
     1'iterations so time step is increased by 1.75',/,4x,'delt=',      chmtrns.1344
     11pe12.4,10x, 'at time=',1pe12.4)                                  chmtrns.1345
 2024 format(i10)                                                       chmtrns.1346
 2050 format (/,4x,'at node',i5,4x,'species',a8,'has redissolved',/,    chmtrns.1347
     1'  the concentration of this species at time' 1pe8.2,4x, 'is',    chmtrns.1348
     2 e12.4,/,'and the concentration of this species at time',e8.2,    chmtrns.1349
     3 4x,'is', e12.4)                                                  chmtrns.1350
 2060 format (/,4x,'inner boundary is constant concentration',10x,      chmtrns.1351
     1'outer boundary is zero flux')                                    chmtrns.1352
 2061 format (/,4x,'inner boundary is constant flux',10x,'outer',       chmtrns.1353
     1' boundary is zero flux')                                         chmtrns.1354
 2062 format (/,4x,'inner boundary is constant concentration',10x,      chmtrns.1355
     1'outer boundary is constant initial concentration')               chmtrns.1356
 2063 format (/,4x,'inner boundary is constant flux',10x,'outer'        chmtrns.1357
     1' boundry is constant initial concentration')                     chmtrns.1358
 2080 format('calculation time= ',e15.5,'seconds')                      chmtrns.1359
 3000 format(/,'redox is considered'/)                                  chmtrns.1360
 3100 format( /,'kinetics of reversible silica dissolution considered'/) chmtrns.1361
 3200 format(/,'kinetics of non-rev. silica dissolution considered',/)   chmtrns.1362
 3300 format(/,'kinetics of rev. precipitate dissolution considered',/)  chmtrns.1363
      end                                                               chmtrns.1364
      subroutine iguess(cm,cbasis,num,n,lnh)                              iguess.2
c                                                                         iguess.3
c  *********************************************************************  iguess.4
c  this subroutine calculates the first guess of the values of the        iguess.5
c  basis species, complexes, sorbed components and precipitates           iguess.6
c  for an equilibrium distribution of species calculation. it assumes     iguess.7
c  that the concentration of the basis species is equal to value set      iguess.8
c  for the initial condition for that species, i.e. it is set equal to    iguess.9
c  bi(j); the concentrations of the complexes and the sorbed components   iguess.10
c  are then calculated from the mass action equations assuming these      iguess.11
c  basis species concentrations.  the concentration of the precipitates   iguess.12
c  set equal to 0 for the first guess. the subroutine does not necessari  iguess.13
c  make the best guess.  the user can read in guesses instead of calling  iguess.14
```

```
c  this routine.  set ig=1 if you want to call this routine, otherwise s iguess.15
c  it equal to 0.                                                        iguess.16
c  ********************************************************************* iguess.17
      dimension cm(1),num(1),cbasis(25)                                  iguess.18
      common/ab/vj(75),ac(75,100),eqconst(25,100),h2ok(100),alfa(4,100),   /ab/.2
     1s(25,25),ss(25,25),ah(25),h2oc(8),con(25,8),conk(20,8),conp(4,2),    /ab/.3
     2 ilbli(25),ilble(25),q(10,100)                                       /ab/.4
      common/edl/c1,c2,area,soh                                           /edl/.2
      common/ad/x(200),delt,doa,dob,vo,cecns                               /ad/.2
      m5=num(5)                                                          iguess.22
      nspecm=num(1)                                                      iguess.23
      if (lnh.gt.0) nspecm=num(1)-1                                      iguess.24
c                                                                        iguess.25
c  set the concentration of the basis species equal to the concentration iguess.26
c  read in for that basis species even though that value may be for      iguess.27
c  the total aqueous phase concentration of that species.                iguess.28
c                                                                        iguess.29
      do 10 j=1,nspecm                                                   iguess.30
   10 cm(j)=cbasis(j)                                                    iguess.31
      if(lnh.gt.0) cm(num(1))=h2ok(n)/cm(lnh)                            iguess.32
c                                                                        iguess.33
c  calculate the sorbed concentrations from the mass action equations    iguess.34
c                                                                        iguess.35
      if (num(2).eq.0) go to 100                                         iguess.36
      cm(num(1)+1)=cecns                                                 iguess.37
      amtotal=0                                                          iguess.38
      j1=ilbli(1)                                                        iguess.39
      m2=num(2)                                                          iguess.40
      do 20 j=2,m2                                                       iguess.41
      jj=ilbli(j)                                                        iguess.42
      cm(num(1)+j)=(eqconst(j,n)/eqconst(1,n))**(1./vj(j1))*cm(num(1)+1) iguess.43
     1**(vj(jj)/vj(j1))/(cm(j1)**(vj(jj)/vj(j1)))                        iguess.44
      cm(num(1)+j)=cm(jj)*cm(num(1)+j)                                   iguess.45
   20 continue                                                           iguess.46
  100 if (num(3).eq.0) go to 200                                         iguess.47
      ij1=num(2)+1                                                       iguess.48
      ij2=num(2)+2                                                       iguess.49
      cm(num(6)+3)=soh                                                   iguess.50
      kk=num(6)                                                          iguess.51
      cm(kk+2)=cm(lnh)*soh*0.1/eqconst(num(2)+1,n)                       iguess.52
      cm(kk+1)=eqconst(num(2)+2,n)*cm(kk+3)/cm(lnh)                      iguess.53
      if (num(3).lt.4) go to 200                                         iguess.54
      m3=num(3)                                                          iguess.55
      do 120 i=4,m3                                                      iguess.56
      jj=ilble(i)                                                        iguess.57
      ij=num(2)+i                                                        iguess.58
      cm(kk+i)=eqconst(ij,n)*cm(kk+3)/((cm(lnh))**ah(i))                 iguess.59
  120 continue                                                           iguess.60
  200 continue                                                           iguess.61
c                                                                        iguess.62
c  calculate the concentration of the complexes                          iguess.63
c  using the mass action equations                                       iguess.64
c                                                                        iguess.65
      if (num(4).eq.0) go to 300                                         iguess.66
      m4=num(4)                                                          iguess.67
      do 250 i=1,m4                                                      iguess.68
      ccm=eqconst(num(12)+i,n)                                           iguess.69
      m1=num(1)                                                          iguess.70
      do 220 j=1,m1                                                      iguess.71
      if (s(j,i).eq.0.) go to 220                                        iguess.72
```

```
      ccm=ccm*cm(j)**s(j,i)                                             iguess.73
  220 continue                                                          iguess.74
      cm(num(7)+i)=ccm                                                  iguess.75
  250 continue                                                          iguess.76
c                                                                       iguess.77
c  set the first guess for the precipitates to be 0                     iguess.78
c                                                                       iguess.79
  300 if (num(5).eq.0) go to 400                                        iguess.80
      do 350 j=1,m5                                                     iguess.81
  350 cm(num(8)+j)=0.0                                                  iguess.82
  400 return                                                            iguess.83
      end                                                               iguess.84
      subroutine eqlib(cm,wconst,index1,index2,num,n,ss,eqconst,ctot,    eqlib.2
c                                                                        eqlib.3
c                                                                        eqlib.4
     1ac,lchrg,lnh,in)                                                   eqlib.5
c  ********************************************************************* eqlib.6
c  this subroutine calculates the equilibrium distribution of species.   eqlib.7
c  calls the subroutine reseq which calculates the residues of the equil eqlib.8
c  equations.  the equilibrium distribution of species is calculated giv eqlib.9
c  the type of conditions set for each species, i.e. total concentration eqlib.10
c  species, bare ion concentration, etc.  this set of conditions must eq eqlib.11
c  the number of basis species. the mass action expressions, solubility  eqlib.12
c  equations, and the site constraint equation are used to calculate the eqlib.13
c  concentrations of the rest of the species.                            eqlib.14
c  ********************************************************************* eqlib.15
      dimension cm(1),wconst(1),z(25,25),y(25),index1(1),index2(1)       eqlib.16
     1, ss(25,1),eqconst(25,1),ac(75,1),iterp(35),num(1)                 eqlib.17
      common/edl/c1,c2,area,soh                                          /edl/.2
      common/spec/leach1(5),nb,pini(5)                                  /spec/.2
c                                                                        eqlib.20
      mdim=25                                                            eqlib.21
      ntot=num(8)                                                        eqlib.22
c                                                                        eqlib.23
      if (num(3).eq.0) soh=20.                                           eqlib.24
      iter=1                                                             eqlib.25
      if (num(5).eq.0) go to 2                                           eqlib.26
      m5=num(5)                                                          eqlib.27
      do 1 i=1,m5                                                        eqlib.28
    1 iterp(i)=1                                                         eqlib.29
    2 iter=iter+1                                                        eqlib.30
c                                                                        eqlib.31
c  if convergence is not obtained within 100 iterations, the             eqlib.32
c  calculations are stopped                                              eqlib.33
c                                                                        eqlib.34
      if (iter.gt.100) go to 9                                           eqlib.35
      nsol=0                                                             eqlib.36
c                                                                        eqlib.37
c  the solubility product for a precipitate is only included when        eqlib.38
c  the product of the activities of the species making up that           eqlib.39
c  precipitate exceeds this product;  to aviod oscillations between      eqlib.40
c  the precipitate wanting to form or not, the solubility product        eqlib.41
c  is included only when it is exceeded for two calculations             eqlib.42
c                                                                        eqlib.43
      if (num(5).eq.0) go to 40                                          eqlib.44
      do 4 i=1,m5                                                        eqlib.45
      aksol=1.0                                                          eqlib.46
      m1=num(1)                                                          eqlib.47
      do 5 j=1,m1                                                        eqlib.48
      if(cm(j).eq.0.0) cm(j)=1.0e-35                                     eqlib.49
```

```
      aksol=aksol*(ac(j,n)*cm(j))**ss(j,i)                              eqlib.50
    5 continue                                                          eqlib.51
c                                                                       eqlib.52
c  if a solid has been put in contact with the groundwater at the       eqlib.53
c  first node point, the solubility product of that precipitate will    eqlib.54
c  always be included for the distribution of species at the inner      eqlib.55
c  boundary condition.                                                  eqlib.56
c                                                                       eqlib.57
      if ((leach1(1).eq.i).and.(n.eq.nb)) index1(i)=1                   eqlib.58
      if ((leach1(1).eq.i).and.(n.eq.nb)) go to 77                      eqlib.59
      index1(i)=0                                                       eqlib.60
      ij=num(12)+num(4)+i                                               eqlib.61
      ek=0.001*eqconst(ij,n)                                            eqlib.62
      if ((eqconst(ij,n).lt.aksol).or.(cm(ntot+i).gt.1.e-16)) iterp(i)  eqlib.63
     1 =iterp(i)+1                                                      eqlib.64
      if((eqconst(ij,n).lt.aksol).or.(cm(ntot+i).gt.1.e-16)) go to 76   eqlib.65
      iterp(i)=0                                                        eqlib.66
   76 if (iterp(i).ge.2) index1(i)=1                                    eqlib.67
   77 if (index1(i).eq.1) nsol=nsol+1                                   eqlib.68
      m9=num(9)                                                         eqlib.69
      if(iter.gt.in) write (6,900) i,aksol,eqconst(ij,n)                eqlib.70
  900 format (4x,'precipitate no.',4x,i5,10x,'precipitation check',     eqlib.71
     1 10x,'product of concentrations is',                              eqlib.72
     24x,e12.4,/,4x,'solubility product is ',4x,e12.5)                  eqlib.73
    4 continue                                                          eqlib.74
   40 continue                                                          eqlib.75
c                                                                       eqlib.76
c  residues of the constraint equations for the initial and boundary    eqlib.77
c  conditions and the mass action and site constraint equation are      eqlib.78
c  calculated in the subroutine reseq                                   eqlib.79
c                                                                       eqlib.80
      call reseq(wconst,cm,index1,index2,y,n,ctot,mdim,num,lchrg,lnh,   eqlib.81
     1 nsol,0)                                                          eqlib.82
      ntotp=ntot+nsol                                                   eqlib.83
c     print 4000 , ntot, nsol, ntotp                                    eqlib.84
 4000 format(3i10)                                                      eqlib.85
      if (iter.gt.in) write (6,1010) (y(i),i=1,ntotp)                   eqlib.86
 1010 format (/,4x,'residues are',/,4(8e12.4,/))                        eqlib.87
c                                                                       eqlib.88
c  the jacobian for the equilibrium distribution of species; i.e. for th eqlib.89
c  equations in subroutine reseq is calculated here                     eqlib.90
c                                                                       eqlib.91
      call jacob2(z,cm,wconst,index1,index2,y,num,n,ctot,nsol,lchrg,    eqlib.92
     1lnh,mdim)                                                         eqlib.93
c                                                                       eqlib.94
c  from the jacobian and the residues for the last guess, calculate the eqlib.95
c  new guess for the all the species                                    eqlib.96
c                                                                       eqlib.97
      call simq(z,y,ntotp,mdim)                                         eqlib.98
c                                                                       eqlib.99
  101 format(4x,i5,10x,'values for this iteration',/,/,4x,              eqlib.100
     1'species no.',10x,'old value',                                    eqlib.101
     2 10x,'new value',10x,'error',/)                                   eqlib.102
      if (iter.gt.in) write (6,101) iter                                eqlib.103
      do 6 j=1,ntot                                                     eqlib.104
      cc1=cm(j)                                                         eqlib.105
      cm(j)=cm(j)-y(j)                                                  eqlib.106
      if (iter.gt.in) write (6,1000) j,cc1,cm(j),y(j)                   eqlib.107
 1000 format (/,4x,i10,10x,1pe12.4,9x,1pe12.4,9x,1pe12.4)               eqlib.108
c                                                                       eqlib.109
```

```
c  the concentration of the species is not allowed to go negative or     eqlib.110
c  for the surface complexation model, the sorbed concentration cannot    eqlib.111
c  be greater than the total number of sites                              eqlib.112
c                                                                         eqlib.113
      if(cm(j).lt.0.) cm(j)=cc1/10.                                       eqlib.114
      if ((j.gt.num(6)).and.(j.lt.num(7)).and.(cm(j).gt.soh)) cm(j)=soh   eqlib.115
    6 continue                                                            eqlib.116
c                                                                         eqlib.117
      index=0                                                             eqlib.118
c                                                                         eqlib.119
c  calculate the amount of the precipitates;  only precipitates that are  eqlib.120
c  forming are included in this calculation                               eqlib.121
c                                                                         eqlib.122
      if (num(5).eq.0) go to 66                                           eqlib.123
      do 65 i=1,m5                                                        eqlib.124
      if (index1(i).eq.0) go to 65                                        eqlib.125
      index=index+1                                                       eqlib.126
      cc1=cm(ntot+i)                                                      eqlib.127
      cm(ntot+i)=cm(ntot+i)-y(ntot+index)                                 eqlib.128
      ii=ntot+i                                                           eqlib.129
      iii=ntot+index                                                      eqlib.130
      if (iter.gt.in) write (6,1000) i,cc1,cm(ii),y(iii)                  eqlib.131
      if (cm(ntot+i).lt.0.) iterp(i)=0                                    eqlib.132
      if (cm(ntot+i).lt.0.) cm(ntot+i)=0.                                 eqlib.133
   65 continue                                                            eqlib.134
   66 continue                                                            eqlib.135
c                                                                         eqlib.136
c  check convergence;  if within the criteria set, then end the           eqlib.137
c  iterations, otherwise continue                                         eqlib.138
c                                                                         eqlib.139
      do 7 j=1,ntotp                                                      eqlib.140
      if (cm(j).lt.1.e-30) go to 7                                        eqlib.141
      if(abs(y(j)/cm(j)).gt.1.0e-04) go to 2                              eqlib.142
    7 continue                                                            eqlib.143
      return                                                              eqlib.144
    9 write(6,100) iter                                                   eqlib.145
  100 format(' no convergence in eqlib after iter ='i3)                   eqlib.146
      stop                                                                eqlib.147
      end                                                                 eqlib.148
      subroutine simq(z,y,n,mdim)                                          simq.2
c                                                                          simq.3
c                                                                          simq.4
c  ****************************************************                    simq.5
c  inversion of a matrix; it is used in calculating                        simq.6
c  an equilibrium distribution of species such as                          simq.7
c  in the initial conditions and the boundary conditions.                  simq.8
c  the jacobian is stored in z while the residues are in                   simq.9
c  y.  this  subroutine solves the equation zx=y for x                     simq.10
c  ****************************************************                    simq.11
      dimension z(mdim,1),y(1)                                             simq.12
c     provision for n=1                                                    simq.13
      if(n.ne.1) go to 50                                                  simq.14
      y(1)=y(1)/z(1,1)                                                     simq.15
      return                                                               simq.16
   50 continue                                                             simq.17
c     element of elimination                                               simq.18
      n1=n-1                                                               simq.19
      do 10 m=1,n1                                                         simq.20
      zmax=0.                                                              simq.21
      imax=0                                                               simq.22
```

```
c     find max of column                                                 simq.23
      do 20 i=m,n                                                        simq.24
      if(abs(z(i,m)).le.zmax) go to 20                                   simq.25
      imax=i                                                             simq.26
      zmax=abs(z(i,m))                                                   simq.27
   20 continue                                                           simq.28
c     error return                                                       simq.29
      if(imax.ne.0) go to 30                                             simq.30
      write(6,1000)                                                      simq.31
      write(6,900) i,m                                                   simq.32
      write (6,990) (z(i,m),i=1,n)                                       simq.33
   30 continue                                                           simq.34
c     row interchange                                                    simq.35
      if(imax.eq.m) go to 35                                             simq.36
      v=y(m)                                                             simq.37
      y(m)=y(imax)                                                       simq.38
      y(imax)=v                                                          simq.39
      do 40 j=m,n                                                        simq.40
      v=z(m,j)                                                           simq.41
      z(m,j)=z(imax,j)                                                   simq.42
      z(imax,j)=v                                                        simq.43
   40 continue                                                           simq.44
   35 continue                                                           simq.45
c     diagonalize                                                        simq.46
      m1=m+1                                                             simq.47
      do 70 i=m1,n                                                       simq.48
      v=z(i,m)/z(m,m)                                                    simq.49
      y(i)=y(i)-v*y(m)                                                   simq.50
      do 70 j=m,n                                                        simq.51
      z(i,j)=z(i,j)-v*z(m,j)                                             simq.52
   70 continue                                                           simq.53
   10 continue                                                           simq.54
c     back substitute                                                    simq.55
      y(n)=y(n)/z(n,n)                                                   simq.56
      n1=n-1                                                             simq.57
      do 100 k=1,n1                                                      simq.58
      i=n-k                                                              simq.59
      i1=i+1                                                             simq.60
      do 90 j=i1,n                                                       simq.61
   90 y(i)=y(i)-y(j)*z(i,j)                                              simq.62
  100 y(i)=y(i)/z(i,i)                                                   simq.63
      return                                                             simq.64
  900 format (4x,2i10)                                                   simq.65
  990 format (4x,5(19e12.4))                                             simq.66
 1000 format(' singular jacobian matrix')                                simq.67
      end                                                                simq.68
      subroutine setup(xmax,nmax,itpe,gc,rw)                            setup.2
c                                                                       setup.3
c                                                                       setup.4
c  ******************************************************************** setup.5
c   calculates the grid based on an exponential law;if gc is set        setup.6
c   to 0, a uniform grid is calculated.                                 setup.7
c  ******************************************************************** setup.8
      common/ad/x(200),delt,doa,dob,vo,cecns                             /ad/.2
c                                                                       setup.10
      dn=1./float(nmax-1)                                               setup.11
                                                                        setup.12
      x(1)=0.                                                           setup.13
      if(itype.eq.1) x(1)=rw                                            setup.14
                                                                        setup.15
```

```
      x1=x(1)                                                           setup.16
      do 10 i=2,nmax                                                    setup.17
      pip=float(i-1)                                                    setup.18
      if(gc.gt.0.) go to 5                                              setup.19
c     calcultion for uniform grid                                       setup.20
c                                                                       setup.21
      x(i)=xmax*dn*pip+x1                                               setup.22
      go to 10                                                          setup.23
c                                                                       setup.24
c     calculation for variable grid                                     setup.25
c                                                                       setup.26
    5 x(i)=xmax*(gc**((pip)*dn)-1.)/(gc-1.)+x1                          setup.27
   10 continue                                                          setup.28
      write(6,190) (x(n),n=1,nmax)                                      setup.29
  190 format(/55x,' ** spatial grid ** ',//,(' ',10g13.4))              setup.30
      return                                                            setup.31
      end                                                               setup.32
      subroutine res(cn,co,bc,bc2,r,num,mdim,lnh,ibnd,itype,jacob)        res.2
c                                                                         res.3
c                                                                         res.4
c                                                                         res.5
c                                                                         res.6
c                                                                         res.7
c  *******************************************************************    res.8
c  calculates the residues of the transport equations, the site           res.9
c  constraint equation, the mass action equations, and the boundary       res.10
c  conditions. the boundary conditions are stored in the arrays bc        res.11
c  and bc2.  if boundary conditions other than the concentration and      res.12
c  flux conditions are of interest, the user would have to supply how     res.13
c  the values of bc and bc2 would change.  for the concentration boundar  res.14
c  condition at the inner node, bc stores the total concentration of a    res.15
c  species in the aqueous phase and in the form of a precipitate.  for t  res.16
c  flux condition at this boundary, bc stores the fluid velocity times    res.17
c  the total aqueous phase concentration of each species.  for the outer  res.18
c  boundary, for the concentration condition, bc2 stores the basis        res.19
c  species concentration of the initial conditions, and for the no flux   res.20
c  condition it is not used.                                              res.21
c  the subroutine is called from both the main program and the subroutin  res.22
c  jacobi.                                                                res.23
c  *******************************************************************    res.24
      dimension cn(1),co(1),bc(1),r(mdim,1),num(1),ibnd(1),bc2(1)         res.25
      common/ab/vj(75),ac(75,100),eqconst(25,100),h2ok(100),alfa(4,100),   /ab/.2
     1s(25,25),ss(25,25),ah(25),h2oc(8),con(25,8),conk(20,8),conp(4,2),    /ab/.3
     2 ilbli(25),ilble(25),q(10,100)                                       /ab/.4
      common/ac/elconc(100)                                                /ac/.2
      common/nj/isorpi(30),isorpe(30),iion,ieion                           /nj/.2
      common/ad/x(200),delt,doa,dob,vo,cecns                               /ad/.2
      common/crate/rate(50),kcode(25),inx1(5,25),inx2(5,25),cdat(5,25),  /crate/.2
     1 reactsk(5,10,100),nsrt(5),sarea(5),nparam(5),xkinet(5),xmax,time, /crate/.3
     2 gc,excode,tpcode,kinr,xdiag(5),shift(4,100),shftsol(100)          /crate/.4
      nmax=num(14)                                                        res.31
      nmaxm=nmax-1                                                        res.32
      m1=num(1)                                                           res.33
      m2=num(2)                                                           res.34
      m3=num(3)                                                           res.35
      m4=num(4)                                                           res.36
      m5=num(5)                                                           res.37
      ntot=num(9)                                                         res.38
c                                                                         res.39
c  calculate the activities of the aqueous phase species                  res.40
```

```
c                                                                       res.41
      call active(num,cn,vj,ac,elconc,75,1,num(14))                     res.42
c                                                                       res.43
c  *********************************************************            res.44
c   inner boundary condition set here                                   res.45
c **********************************************************            res.46
      n=1                                                               res.47
      n1=2                                                              res.48
c                                                                       res.49
      if( ibnd(1).eq.1 ) n1=1                                           res.50
c                                                                       res.51
c  residue calculations for internal boundary drichlet condition        res.52
c                                                                       res.53
      if(n1.eq.1) go to 10                                              res.54
      do 20 j=1,ntot                                                    res.55
   20 r(j,1)=1.0e-30                                                    res.56
      go to 25                                                          res.57
c                                                                       res.58
c  calculate the residues for the mass action equations                 res.59
c                                                                       res.60
c                                                                       res.61
c                                                                       res.62
   10 call algebrc(num,cn,co,r,1,1,mdim,elconc,lnh,0,jcob)              res.63
   25 continue                                                          res.64
c                                                                       res.65
c ************************************************************          res.66
c  calculation of changes in the porous medium                          res.67
c *************************************************************         res.68
      do 170 n=n1,nmaxm                                                 res.69
      neven=(n-1)*ntot                                                  res.70
      nplus=neven+ntot                                                  res.71
      nminus=neven-ntot                                                 res.72
      ii=0                                                              res.73
      ie=3                                                              res.74
      do 40 i=1,50                                                      res.75
      rate(i)=0.0                                                       res.76
   40 continue                                                          res.77
      if(kinr.eq.0) go to 45                                            res.78
      if(excode.eq.0) go to 45                                          res.79
      call srate(n,num,co,cn,ac,neven,delt,0)                           res.80
   45 continue                                                          res.81
      do 78 j=1,m1                                                      res.82
      vv=vo                                                             res.83
      if (itype.eq.1) vv=vo/x(n)                                        res.84
      dd=doa*vv+dob                                                     res.85
c                                                                       res.86
c  for a flux condition at the inner boundary and for                   res.87
c  radial flow, the dispersion coefficient and the fluid velocity       res.88
c  for the first grid are calculated here                               res.89
c                                                                       res.90
      if (itype.eq.1) vv=vv-( dob/x(n) )                                res.91
      if (n.eq.1) go to 50                                              res.92
c                                                                       res.93
c  calculate the residues of the transport equations for nodes          res.94
c  2 through nmax-1 including the changes in the basis species          res.95
c                                                                       res.96
      r(j,n)=(cn(neven+j)-co(neven+j))/delt-dd*((cn(nplus+j)-           res.97
     1cn(neven+j))/(x(n+1)-x(n))-(cn(neven+j)-cn(nminus+j))             res.98
     2/(x(n)-x(n-1)))/(0.5*(x(n+1)-x(n-1)))+vv*(cn(nplus+j)-            res.99
     3 cn(nminus+j))/(x(n+1)-x(n-1))+rate(kcode(j))                     res.100
```

```
      go to 51
c
c  for the flux boundary condition at the first node point,
c  calculate the changes in the first node point for the
c  basis species
c
   50 r(j,1)=(cn(neven+j)-co(neven+j))/delt
      r(j,1)=r(j,1)-2.0*dd*(cn(nplus+j)-cn(neven+j))/((x(2)-x(1))**2)
      r(j,1)=r(j,1)+(vv*cn(neven+j)-bc(j))/(x(2)-x(1))
     1      *(2.0+vv*(x(2)-x(1))/dd)+rate(kcode(j))
   51  continue
c
c  include the change in the sorption of the basis species
c  for  nodes 2 thru nmax-1 or 1 thru nmax-1 depending on the
c  boundary conditions; first is for sorption via ion exchange
c  next is for any sorption via surface complexation
c
      if (num(2).eq.0) go to 68
      if (isorpi(j).eq.0) go to 68
      ii=ii+1
      ip=num(1)+ii
      r(j,n)=r(j,n)+(cn(neven+ip)-co(neven+ip))/delt
   68 continue
      if (num(3).eq.0) go to 69
      if (isorpe(j).eq.0) go to 69
      if (j.eq.lnh) go to 70
      ie=ie+1
      ip=num(6)+ie
      r(j,n)=r(j,n)+(cn(neven+ip)-co(neven+ip))/delt
      go to 69
   70 ip=num(6)
c
c  for sorption via surface complexation, need to consider the
c  changes in the soh and soh2+ sites
c
      r(j,n)=r(j,n)+2*(cn(neven+ip+2)-co(neven+ip+2))/delt
     1  +(cn(neven+ip+3)-co(neven+ip+3))/delt
c
c  next include changes in the complexes at the node points;
c  time changes, advection and dispersion
c
   69 if (num(4).eq.0) go to 72
      do 71 i=1,m4
      ip=num(7)+i
       if (n.eq.1) go to 52
      r(j,n)=r(j,n)+s(j,i)*(cn(neven+ip)-co(neven+ip))/delt-s(j,i)*
     1dd*((cn(nplus+ip)-cn(neven+ip))/(x(n+1)-x(n))-(cn(neven
     2 +ip)-cn(nminus+ip))/(x(n)-x(n-1)))/(0.5*(x(n+1)-x(n-1)))+
     1 s(j,i)*vv*(cn(nplus+ip)-cn(nminus+ip))/(x(n+1)-x(n-1))
      go to 53
   52  continue
c
c  special case for considering these last changes at the first
c  node point if flux boundary condition at the inner boundary
c
       r(j,1)=r(j,1)+s(j,i)*(cn(neven+ip)-co(neven+ip))/delt
      r(j,1)=r(j,1)-2.0*s(j,i)*dd*(cn(nplus+ip)-cn(neven+ip))/(
     1  (x(2)-x(1))**2.0)
      r(j,1)=r(j,1)+s(j,i)*vv*cn(neven+ip)*(2.0/(x(2)-x(1))+vv/dd)
   53 continue
```

```
c
c  need to also include the changes in the sorption of the complexes
c  whether via ion exchange or surface complexation
c
      if (num(2).le.iion) go to 568
      if (isorpi(j+num(1)).eq.0) go to 568
      ii=ii+1
      ip=num(1)+ii
      r(j,n)=r(j,n)+s(j,i)*(cn(neven+ip)-co(neven+ip))/delt
  568 continue
      if (num(3).le.ieion) go to 71
      if (isorpe(num(1)+j).eq.0) go to 71
      ie=ie+1
      ip=num(6)+ie
      r(j,n)=r(j,n)+s(j,i)*(cn(neven+ip)-co(neven+ip))/delt
   71 continue
   72 continue
c
c  include the changes in the precipitates at a node point
c
      if (num(5).eq.0) go to 78
      do 74 i=1,m5
      ip=num(8)+i
      r(j,n)=r(j,n)+ss(j,i)*(cn(neven+ip)-co(neven+ip))/delt
   74 continue
   78 continue
c
c  when water dissociation is included, need to actually calculate
c  the changes for the dummy variable which is the concentration of
c  the h+ ion minus the concentration of the oh- ion
c
      if (lnh.ne.0) r(lnh,n)=r(lnh,n)-r(num(1),n)
      if(lnh.ne.0) r(num(1),n)=h2ok(n)-ac(lnh,n)*cn(neven+lnh)*
     1                    ac(num(1),n)*cn(neven+num(1))
c
c  calculate the changes in the site constraint, the mass action,
c  and the solubility products at the node points
c
  170 continue
c
c
      call algebrc(num,cn,co,r,n1,nmax,mdim,elconc,lnh,0,jcob)
c
c  calculation of the conditions at the last node point
c
      n=nmax
      neven=(n-1)*ntot
      nminus=(n-2)*ntot
      if (ibnd(2).eq.0) go to 210
c
c  no flux condition
c
      do 200 j=1,m1
      r(j,n)=( cn(neven+j)-cn(nminus+j) )/( x(nmax)-x(nmax-1) )
  200 continue
      go to 220
c
c  concentration boundary condition
c
  210 do 215 j=1,m1
```

```
  215 r(j,n)=cn(neven+j)-bc2(j)                                          res.221
  220 continue                                                           res.222
c                                                                        res.223
      if (lnh.ne.0) r(lnh,n)=r(lnh,n)-r(num(1),n)                        res.224
      if(lnh.ne.0) r(num(1),n)=h2ok(n)-ac(lnh,n)*cn(neven+lnh)*          res.225
     1                    ac(num(1),n)*cn(neven+num(1))                  res.226
c                                                                        res.227
c  calculate the residues for the algebric equations for the             res.228
c  conditions at the last node point                                     res.229
c                                                                        res.230
c                                                                        res.231
c  residue calculations for external boundary drichlet condition         res.232
c                                                                        res.233
      if( ibnd(2).ne.0 ) go to 225                                       res.234
      do 230 j=1,ntot                                                    res.235
  230 r(j,nmax)=1.0e-30                                                  res.236
      go to 235                                                          res.237
  225 call algebrc(num,cn,co,r,nmax,nmax,mdim,elconc,lnh,0,jcob)         res.238
  235 continue                                                           res.239
      return                                                             res.240
      end                                                                res.241
      subroutine active(num,cn,vj,ac,elconc,mdim,nbegin,nend)         active.2
c                                                                     active.3
c  **********************************************************         active.4
c  calculates the activities based on the davies equation             active.5
c  the ionic strength at each node point is stored in elconc,         active.6
c  the activities are in the matrix ac(i,n)                           active.7
c  **********************************************************         active.8
      dimension num(1),cn(1),vj(1),ac(mdim,1),elconc(1)               active.9
      common/oxred/lno,lnh                                           /oxred/.2
c                                                                     active.11
  200 format(i10)                                                     active.12
      m1=num(1)                                                       active.13
      m4=num(4)                                                       active.14
      ntot=num(9)                                                     active.15
      if (nbegin.eq.nend) ntot=0                                      active.16
      do 50 n=nbegin,nend                                             active.17
c                                                                     active.18
c  calculation of ionic strength                                      active.19
c                                                                     active.20
      ctot=0.                                                         active.21
      si=0.                                                           active.22
      do 10 j=1,m1                                                    active.23
      if(j.eq.lno) go to 10                                           active.24
      if (cn((n-1)*ntot+j) .lt.0.) go to 60                           active.25
      ctot=ctot+vj(j)*cn((n-1)*ntot+j)                                active.26
      si=si+vj(j)**2*cn((n-1)*ntot+j)                                 active.27
   10 continue                                                        active.28
      if (num(4).eq.0) go to 16                                       active.29
      do 15 i=1,m4                                                    active.30
      if (cn((n-1)*ntot+num(7)+i).lt.0) go to 60                      active.31
      ctot=ctot+vj(num(1)+i)*cn((n-1)*ntot+num(7)+i)                  active.32
   15 si=si+vj(num(1)+i)**2*cn((n-1)*ntot+num(7)+i)                   active.33
   16 continue                                                        active.34
c                                                                     active.35
c  although the net charge of the species being considered is in the  active.36
c  calculations may not be equal to zero, for an accurate calculation active.37
c  of the ionic strength, all the chemical species must be considered active.38
c  therefore, it is assumed here that the balance of the charge is made active.39
c  up by a chemical species of a +1 or -1 charge and a concentration  active.40
```

```
c  of ctot.  if this is not reasonable, then the actual calculation       active.41
c  should be done with a net charge of 0.                                 active.42
c                                                                         active.43
      si=si+abs(ctot)                                                     active.44
      cionst=0.5*si                                                       active.45
      elconc(n)=cionst                                                    active.46
      si=sqrt(cionst)                                                     active.47
      fi=0.5*(si/(1.+si)-0.3*cionst)                                      active.48
c                                                                         active.49
c  calculation of activity coefficients                                   active.50
c                                                                         active.51
      do 20 j=1,m1                                                        active.52
      if(j.eq.lno) go to 20                                               active.53
      g=fi*vj(j)**2                                                       active.54
      ac(j,n)=10.**(-g)                                                   active.55
   20 continue                                                            active.56
      if (num(4).eq.0) go to 50                                           active.57
      do  25 i=1,m4                                                       active.58
      gcx=fi*vj(num(1)+i)**2                                              active.59
   25 ac(num(1)+i,n)=10.**(-gcx)                                          active.60
   50 continue                                                            active.61
      return                                                              active.62
   60 write (6,100)                                                       active.63
  100 format (/,4x,'the concentration is less than zero which is not'     active.64
     1 ' allowed, therefore the program has stopped',/)                   active.65
      return                                                              active.66
      end                                                                 active.67
      subroutine algebrc(num,cn,co,r,n1,n2,mdim,elconc,lnh,k1,jcob)      algebrc.2
c                                                                        algebrc.3
c                                                                        algebrc.4
c  **********************************************************            algebrc.5
c  calculates the residue for the site constraint equations              algebrc.6
c  and the mass action equations describing the sorption                 algebrc.7
c  reactions and the formation of complexes and precipitates             algebrc.8
c  **********************************************************            algebrc.9
c                                                                        algebrc.10
      dimension num(1),cn(1),r(mdim,1),residue(25),elconc(1)             algebrc.11
      dimension co(1)                                                    algebrc.12
      common/ab/vj(75),ac(75,100),eqconst(25,100),h2ok(100),alfa(4,100),    /ab/.2
     1s(25,25),ss(25,25),ah(25),h2oc(8),con(25,8),conk(20,8),conp(4,2),     /ab/.3
     2 ilbli(25),ilble(25),q(10,100)                                        /ab/.4
      common/nj/isorpi(30),isorpe(30),iion,ieion                            /nj/.2
      common/ad/x(200),delt,doa,dob,vo,cecns                                /ad/.2
      common/crate/rate(50),kcode(25),inx1(5,25),inx2(5,25),cdat(5,25),  /crate/.2
     1 reactsk(5,10,100),nsrt(5),sarea(5),nparam(5),xkinet(5),xmax,time, /crate/.3
     2 gc,excode,tpcode,kinr,xdiag(5),shift(4,100),shftsol(100)          /crate/.4
      m1=num(1)                                                          algebrc.17
      m2=num(2)                                                          algebrc.18
      m3=num(3)                                                          algebrc.19
      m4=num(4)                                                          algebrc.20
      m5=num(5)                                                          algebrc.21
      nmax=num(14)                                                       algebrc.22
      nmaxm=nmax-1                                                       algebrc.23
      kk=k1                                                              algebrc.24
      do 200 n=n1,n2                                                     algebrc.25
      nstart=(n-1)*num(9)                                                algebrc.26
c                                                                        algebrc.27
c  kinetic rate calculations for dissolution                             algebrc.28
c                                                                        algebrc.29
      do 15 i=1,50                                                       algebrc.30
```

```
      rate(i)=0.0                                                       algebrc.31
   15 continue                                                          algebrc.32
      if(k1.eq.1.or.tpcode.eq.0) go to 25                               algebrc.33
      call srate(n,num,co,cn,ac,nstart,delt,1)                          algebrc.34
      ddelx= xmax/nmaxm                                                 algebrc.35
      if(gc.eq.0.0) go to 20                                            algebrc.36
      ddelx=( x(n+1)-x(n) )/2.0+( x(n)-x(n-1) )/2.0                     algebrc.37
      if( n.eq.1 ) ddelx= ( x(n+1)-x(n) )/2.0                           algebrc.38
      if( n.eq.nmax ) ddelx=( x(n)-x(n-1) )/2.0                         algebrc.39
      go to 25                                                          algebrc.40
   20 if( n.eq.1.or.n.eq.nmax ) ddelx=ddelx/2.0                         algebrc.41
   25 continue                                                          algebrc.42
c                                                                       algebrc.43
c  site constant equation for sorption via ion exchange                 algebrc.44
c                                                                       algebrc.45
      if (kk.eq.1) nstart=0                                             algebrc.46
      if (num(2).eq.0) go to 45                                         algebrc.47
      r(num(1)+1,n)=cecns                                               algebrc.48
      amtotal=0.                                                        algebrc.49
      do 35 j=1,m2                                                      algebrc.50
      jj=ilbli(j)                                                       algebrc.51
      r(num(1)+1,n)=r(num(1)+1,n)-vj(jj)*cn(nstart+num(1)+j)            algebrc.52
   35 amtotal=amtotal+cn(nstart+num(1)+j)                               algebrc.53
c                                                                       algebrc.54
c  mass action equations describing sorption via ion exchange           algebrc.55
c                                                                       algebrc.56
      if (num(2).eq.1) go to 45                                         algebrc.57
      j1=ilbli(1)                                                       algebrc.58
      do 40 j=2,m2                                                      algebrc.59
      jj=ilbli(j)                                                       algebrc.60
      aj=(cn(nstart+num(1)+j)/amtotal)**vj(j1)                          algebrc.61
      a1=(cn(nstart+num(1)+1)/amtotal)**vj(jj)                          algebrc.62
   36 r(num(1)+j,n)=(eqconst(j,n)*cn(nstart+jj)*ac(jj,n))**vj(j1)*      algebrc.63
     1 a1-(eqconst(1,n)*cn(nstart+j1)*ac(j1,n))**vj(jj)*aj              algebrc.64
   40 continue                                                          algebrc.65
   45 continue                                                          algebrc.66
      if (num(3).eq.0) go to 52                                         algebrc.67
c                                                                       algebrc.68
c  for soption via surface complexation, mass action equations          algebrc.69
c  are solved in subroutine redl                                        algebrc.70
c                                                                       algebrc.71
      call redl(cn,residue,nstart,n,elconc(n),lnh,num)                  algebrc.72
      do 50 i=1,m3                                                      algebrc.73
      r(num(6)+i,n)=residue(i)                                          algebrc.74
   50 continue                                                          algebrc.75
   52 continue                                                          algebrc.76
c                                                                       algebrc.77
c  dissociation equation for water                                      algebrc.78
c                                                                       algebrc.79
      if (num(4).eq.0) go to  58                                        algebrc.80
c                                                                       algebrc.81
c  mass actions equations for formation of complexes in the aqueous    algebrc.82
c  phase                                                                algebrc.83
c                                                                       algebrc.84
      do 56 i=1,m4                                                      algebrc.85
      cm=eqconst(num(12)+i,n)                                           algebrc.86
      do 54 j=1,m1                                                      algebrc.87
      if (s(j,i).eq.0) go to 54                                         algebrc.88
      cm=cm*(ac(j,n)*cn(nstart+j))**s(j,i)                              algebrc.89
   54 continue                                                          algebrc.90
```

```
   56 r(num(7)+i,n)=ac(num(1)+i,n)*cn(nstart+num(7)+i)-cm                algebrc.91
   58 continue                                                           algebrc.92
      if (num(5).eq.0) go to 170                                         algebrc.93
c                                                                        algebrc.94
c  solubility product equations for the formations of precipitates       algebrc.95
c                                                                        algebrc.96
      do 168 i=1,m5                                                      algebrc.97
      kk=nstart+num(8)+i                                                 algebrc.98
      kn=num(8)+i                                                        algebrc.99
      if( tpcode.eq.0.0 ) go to 150                                      algebrc.100
      if(kcode(kn).eq.0) go to 150                                       algebrc.101
      if(rate(kn).eq.0.or.k1.eq.1) go to 150                             algebrc.102
      r(kn,n)=ddelx*rate(kn)+(cn(kk)-co(kk))/delt                        algebrc.103
c------watch this when having different mass productions---------        algebrc.104
      if(xdiag(i).eq.xkinet(5)) r(kn,n)=rate(kn)+(cn(kk)-co(kk))/delt    algebrc.105
      go to 168                                                          algebrc.106
  150 continue                                                           algebrc.107
      aksol=1.0                                                          algebrc.108
      do 162 j=1,m1                                                      algebrc.109
      if (ss(j,i).lt.1.e-15) go to 162                                   algebrc.110
      aksol=aksol*(ac(j,n)*cn(nstart+j))**ss(j,i)                        algebrc.111
  162 continue                                                           algebrc.112
      if( jcob.eq.0) q(i,n)=aksol                                        algebrc.113
      r(kn,n)=aksol-eqconst(num(12)+num(4)+i,n)                          algebrc.114
      if(cn(kk).lt.1.0e-20.and.r(kn,n).lt.0.) r(kn,n)=cn(kk)             algebrc.115
                                                                         algebrc.116
  168 continue                                                           algebrc.117
  170 continue                                                           algebrc.118
  200 continue                                                           algebrc.119
      return                                                             algebrc.120
      end                                                                algebrc.121
      subroutine srate(n,num,co,cn,ac,neven,delt,kp)                       srate.2
c  ****************************************************************        srate.3
c   This subroutine calculates the rate equations for the revrsible        srate.4
c  Calcite and Silica dissolutions according to equations proposed         srate.5
c  by Plummer et al (1978) and Rimstidt and Barnes (1980). Also,           srate.6
c  imperical Glass dissolution rates suggested by Claassen and             srate.7
c  White, (1979) are calculated here.                                      srate.8
c  ****************************************************************        srate.9
      common/crate/rate(50),kcode(25),inx1(5,25),inx2(5,25),cdat(5,25),  /crate/.2
     1 reactsk(5,10,100),nsrt(5),sarea(5),nparam(5),xkinet(5),xmax,time, /crate/.3
     2 gc,excode,tpcode,kinr,xdiag(5),shift(4,100),shftsol(100)          /crate/.4
      common/oxred/lno,lnh                                               /oxred/.2
      dimension num(1),co(1),cn(1),ac(75,1),rsk(25)                        srate.12
c                                                                          srate.13
      ii=1                                                                 srate.14
      if( kp.eq.1 ) go to 200                                              srate.15
c   **********************************************************:            srate.16
c     kinetic rate calculation for non-rev. silica dissolution             srate.17
c   **********************************************************             srate.18
c                                                                          srate.19
  100 if( excode.ne.1.0 ) go to 300                                        srate.20
      phyd=-alog10(cn(neven+2))                                            srate.21
      phydo=-alog10(co(neven+2))                                           srate.22
      ddtph=(phyd-phydo)/delt                                              srate.23
      cddtph=2.3*(time)**(0.5)*ddtph                                       srate.24
      xlna=-0.17*(phyd-5.75)-13.73                                         srate.25
      xlna=10.0**(xlna)                                                    srate.26
      xlk=-14.54                                                           srate.27
      xlk=10.0**(xlk)                                                      srate.28
```

```
      xlca=-0.50*(phyd-6.25)-14.15
      xlca=10.0**(xlca)
      xlmg=-0.4*(phyd-5.25)-13.88
      xlmg=10.0**(xlmg)
c     the unit of sigma is per liter which may be wrong!
c      assign rate to designated species
      sigma=sarea(kinr)
      rate(1)=-sigma*xlna*(0.5*(time)**(-0.5)-0.17*cddtph)
      rate(3)=-sigma*xlk*(0.5*(time)**(-0.5))
      rate(4)=-2.0*sigma*xlca*(0.5*(time)**(-0.5)-0.5*cddtph)
      rate(5)=-2.0*sigma*xlmg*(0.5*(time)**(-0.5)-0.4*cddtph)
      rate(2)=-(rate(1)+rate(3)+rate(4)+rate(5))
      go to 300
c   ***************************************************************
c     kinetic rate calculation for rev. carbonate dissolution
c   ***************************************************************
  200 if(xdiag(ii).ne.xkinet(4)) go to 250
c-----assign data
c
      skp=10.0**( -cdat(ii,1) )
      bk1=10.0**( -cdat(ii,2) )
      bk2=10.0**( -cdat(ii,3) )
      bkc=10.0**( -cdat(ii,4) )
      tfactor=  ( +cdat(ii,5) )
      xrate=0.0
c-----calculate reaction constant
c
      do 150 i=1,nsrt(ii)
       if( reactsk(ii,i,n).ne.1.0 ) go to 155
      ni2=inx1(ii,2)
      ni4=inx1(ii,4)
c-----calculate back reaction const.in the last rate term
      ahs=( bk1*bk2*ac(ni4,n)*cn(neven+ni4)*ac(ni2,n)*
     1                        cn(neven+ni2)/bkc)**0.5
      fiahs=( rsk(2)*ac(ni2,n)*cn(neven+ni2)+rsk(3) )/ahs+ skp
      rsk(i)=-bk2*fiahs/bkc
      go to 171
  155 rsk(i)=( +reactsk(ii,i,n) )
c-----calculate rate according to the formulla
  171 ccn1=1.0
      ccn2=1.0
      in1=inx1(ii,i)
      in2=inx2(ii,i)
      if( in1.ne.0 ) ccn1=ac(in1,n)*cn(neven+in1)
      if( in2.ne.0 ) ccn2=ac(in2,n)*cn(neven+in2)
      xrate=xrate+rsk(i)*ccn1*ccn2
  150 continue
c  assign rate to designated precipitant replacing its m-action
      rate(num(8)+ii)=tfactor*sarea(ii)*xrate
      ii=ii+1
c   *************************************************************
c    kinetic rate calculation fo rev. silica dissolution
c   *************************************************************
  250 continue
      coef=1.0
      xrate=0.0
      if( xdiag(ii).ne.xkinet(5) ) go to 300
c
      do 10 i=1,nsrt(ii)
      if(inx1(ii,i).lt.0.or.inx2(ii,i).lt.0) coef=-1.0
```

```
      ccn1=1.0                                                             srate.89
      ccn2=1.0                                                             srate.90
      in1=iabs(inx1(ii,i))                                                 srate.91
      in2=iabs(inx2(ii,i))                                                 srate.92
      if(in1.ne.0) ccn1=cn(neven+in1)                                      srate.93
      if(in2.ne.0) ccn2=cn(neven+in2)                                      srate.94
      rsk(i)=(+reactsk(ii,i,n) )                                           srate.95
      xrate=xrate+coef*rsk(i)*ccn1*ccn2                                    srate.96
   10 continue                                                             srate.97
c     assign rate to designated precipitate                                srate.98
      rate(num(8)+ii)=+sarea(ii)*xrate                                     srate.99
  400 format(5e12.4)                                                       srate.100
  300 continue                                                             srate.101
      return                                                               srate.102
      end                                                                  srate.103
      subroutine jacobi(a,b,c,co,cn,bc,bc2,num,ro,mdim,lnh,ibnd,itype,    jacobi.2
     1                tpcode,kcode)                                       jacobi.3
c                                                                         jacobi.4
c  *********************************************************************  jacobi.5
c  calculation of the jacobian which is the change in the residues        jacobi.6
c  of the transport equations, site constraint equation and the           jacobi.7
c  mass action equations for a change in the concentrations.  this subro  jacobi.8
c  considers the residues for the time dependent equations                jacobi.9
c  it is called from the main program chemtrn only                        jacobi.10
c  *********************************************************************  jacobi.11
      dimension ro(mdim,1),cn(1),co(1),cno(100),bc(1),num(1),ibnd(1)      jacobi.12
      dimension a(mdim,mdim,1),b(mdim,mdim,1),c(mdim,mdim,1),bc2(1)       jacobi.13
      dimension kcode(1)                                                  jacobi.14
      common/ab/vj(75),ac(75,100),eqconst(25,100),h2ok(100),alfa(4,100),    /ab/.2
     1s(25,25),ss(25,25),ah(25),h2oc(8),con(25,8),conk(20,8),conp(4,2),     /ab/.3
     2 ilbli(25),ilble(25),q(10,100)                                        /ab/.4
      common/const/rel,absv                                              /const/.2
      common/lcm3/rn(25,100)                                             /lcm3/.2
c                                                                         jacobi.18
      ntot=num(9)                                                         jacobi.19
      nmax=num(14)                                                        jacobi.20
      do 50 k=1,3                                                         jacobi.21
      do 50 j=1,ntot                                                      jacobi.22
      do 30 n=k,nmax,3                                                    jacobi.23
      cno(n)=cn((n-1)*ntot+j)                                             jacobi.24
   30 cn((n-1)*ntot+j)=cn((n-1)*ntot+j)*(1.+rel)+absv                     jacobi.25
      call res(cn,co,bc,bc2,rn,num,mdim,lnh,ibnd,itype,1)                 jacobi.26
      do 45 n=k,nmax,3                                                    jacobi.27
      do 40 i=1,ntot                                                      jacobi.28
      kk=(n-1)*ntot+i                                                     jacobi.29
      if(i.le.num(8)) go to 33                                            jacobi.30
      if(tpcode.eq.1.0.and.kcode(i).ne.0) go to 33                        jacobi.31
      b(i,j,n-1)=0.0                                                      jacobi.32
      a(i,j,n)  =0.0                                                      jacobi.33
      c(i,j,n+1)=0.0                                                      jacobi.34
      if(j.gt.i) go to 40                                                 jacobi.35
      kp=i-num(8)                                                         jacobi.36
      ip=num(12)+num(4)+kp                                                jacobi.37
      if(i.eq.j) go to 32                                                 jacobi.38
      if((q(kp,n).le.eqconst(ip,n)).and.(cn(kk).le.1.e-20)) go to 40      jacobi.39
      a(i,j,n)=(rn(i,n)-ro(i,n))/(cn((n-1)*ntot+j)-cno(n))                jacobi.40
      go to 40                                                            jacobi.41
   32 continue                                                            jacobi.42
      if((q(kp,n).gt.eqconst(ip,n)).or.(cno(n).gt.1.e-20))go to 34        jacobi.43
      a(i,j,n)=1.0                                                        jacobi.44
```

```
   34 if(n.eq.1.and.ibnd(1).eq.0.) a(i,i,1)=1.0                                jacobi.45
      if(n.eq.nmax.and.ibnd(2).eq.0.) a(i,i,nmax)=1.0                          jacobi.46
      go to 40                                                                 jacobi.47
   33 continue                                                                 jacobi.48
      if (n.eq.1) c(i,j,1)=0.                                                  jacobi.49
      if (n.lt.2) go to 35                                                     jacobi.50
      b(i,j,n-1)=(rn(i,n-1)-ro(i,n-1))/(cn((n-1)*ntot+j)-                      jacobi.51
     1cno(n))                                                                  jacobi.52
   35 a(i,j,n)=(rn(i,n)-ro(i,n))/(cn((n-1)*ntot+j)-                            jacobi.53
     1cno(n))                                                                  jacobi.54
      if(n.eq.1.and.ibnd(1).eq.0.) a(i,i,1)=1.0                                jacobi.55
      if (n.eq.nmax) b(i,j,n)=0.                                               jacobi.56
      if(n.eq.nmax.and.ibnd(2).eq.0.) a(i,i,nmax)=1.0                          jacobi.57
      if (n.eq.nmax) go to  40                                                 jacobi.58
      c(i,j,n+1)=(rn(i,n+1)-ro(i,n+1))/(cn((n-1)*ntot+j)-cno(n))               jacobi.59
   40 continue                                                                 jacobi.60
      cn((n-1)*ntot+j)=cno(n)                                                  jacobi.61
   45 continue                                                                 jacobi.62
   50 continue                                                                 jacobi.63
      return                                                                   jacobi.64
      end                                                                      jacobi.65
      subroutine jacob2(z,cm,wconst,index1,index2,re,num,n,ctot,nsol,          jacob2.2
c                                                                              jacob2.3
     1lchrg,lnh,mdim)                                                          jacob2.4
c  *******************************************************************         jacob2.5
c  calculation of the jacobian  done for the calculation of the                jacob2.6
c  equilibrium conditions; it is called from the subroutine eqlib only         jacob2.7
c  *******************************************************************         jacob2.8
      dimension re(1),ren(25),cm(1),wconst(1),index1(1),                       jacob2.9
     1z(25,25),index2(1),num(1)                                                jacob2.10
      common/ab/vj(75),ac(75,100),eqconst(25,100),h2ok(100),alfa(4,100),         /ab/.2
     1s(25,25),ss(25,25),ah(25),h2oc(8),con(25,8),conk(20,8),conp(4,2),          /ab/.3
     2 ilbli(25),ilble(25),q(10,100)                                             /ab/.4
      common/const/rel,absv                                                   /const/.2
c                                                                              jacob2.13
      ntot=num(9)                                                              jacob2.14
      do 15 j=1,ntot                                                           jacob2.15
      cold=cm(j)                                                               jacob2.16
      cm(j)=cm(j)*(1.+rel)+absv                                                jacob2.17
      call reseq(wconst,cm,index1,index2,ren,n,ctot,mdim,                      jacob2.18
     1 num,lchrg,lnh,nsol,1)                                                   jacob2.19
      do 20 i=1,ntot                                                           jacob2.20
      if(i.le.num(8)) go to 13                                                 jacob2.21
      z(i,j)=0.0                                                               jacob2.22
      if(j.gt.i) go to 20                                                      jacob2.23
      kp=i-num(8)                                                              jacob2.24
      ip=num(12)+num(4)+kp                                                     jacob2.25
      if(i.eq.j) go to 12                                                      jacob2.26
      if((q(kp,n).le.eqconst(ip,n)).and.cm(i).le.1.e-20) go to 20              jacob2.27
      z(i,j)=(ren(i)-re(i))/(cm(j)-cold)                                       jacob2.28
      go to 20                                                                 jacob2.29
   12 continue                                                                 jacob2.30
      if((q(kp,n).gt.eqconst(ip,n)).or.cold.gt.1.e-20)go to 20                 jacob2.31
      z(i,j)=1.0                                                               jacob2.32
      go to 20                                                                 jacob2.33
   13 continue                                                                 jacob2.34
      z(i,j)=(ren(i)-re(i))/(cm(j)-cold)                                       jacob2.35
   20 continue                                                                 jacob2.36
      cm(j)=cold                                                               jacob2.37
   15 continue                                                                 jacob2.38
```

```
      return
      end
      subroutine reseq(wconst,cm,index1,index2,re,n,ctot,mdim,
c
     1 num,lchrg,lnh,nsol,jcob)
c     *****************************************************
c     residues of the equilibrium equations; it is called from
c     the subroutines eqlib and jacob2
c    ******************************************************
      dimension wconst(1),cm(1),index1(1),index2(1),re(1),num(1)
      common/ac/elconc(100)
      common/lcm1/r(25,100)
      common/ab/vj(75),ac(75,100),eqconst(25,100),h2ok(100),alfa(4,100),
     1s(25,25),ss(25,25),ah(25),h2oc(8),con(25,8),conk(20,8),conp(4,2),
     2 ilbli(25),ilble(25),q(10,100)
      common/nj/isorpi(30),isorpe(30),iion,ieion
      common/ad/x(200),delt,doa,dob,vo,cecns
      common/spec/leach1(5),nb,pini(5)
      ntot=num(8)
      ii=0
      ie=3
      m1=num(1)
      m4=num(4)
      m5=num(5)
      m2=num(2)
      m3=num(3)
c
c  calculate the activites of the species in the aqueous phase
c
      call active(num,cm,vj,ac,elconc,75,n,n)
c
c  the first set of equations are setting the conditions to what
c  were specified for each species;
c
   26 do 60 j=1,m1
      r(j,n)=wconst(j)-cm(j)
      if (isorpi(j).eq.0) go to 30
      ii=ii+1
c
c  if index2 .ne. 0 then the amount of the species which is sorbed
c  onto the solid matrix is not included in the specified amount of
c  this species
c
      if (index2(j).ne.0) go to 30
      r(j,n)=r(j,n)-cm(num(1)+ii)
   30 continue
      if (isorpe(j).eq.0) go to 35
      ie=ie+1
      iej=ie
      if (j.eq.lnh) iej=3
      if (j.eq.lnh) ie=ie-1
      if (index2(j).ne.0) go to 35
      r(j,n)=r(j,n)-cm(num(6)+iej)
      if (j.eq.lnh) r(j,n)=r(j,n)-2*cm(num(6)+2)
   35 continue
      if (num(4).eq.0) go to 41
      icmplx=iion
      iecmplx=ieion
c
c  if index2 = 1 then only the concentration of the basis species
```

```
c  itself is being given for this species
c
      if (index2(j).eq.1) go to 41
      do 40 i=1,m4
      kk=i+num(1)
      if (isorpi(kk).eq.0) go to 37
      icmplx=icmplx+1
      if (index2(j).ne.0) go to 37
      r(j,n)=r(j,n)-s(j,i)*cm(num(1)+icmplx)
   37 continue
      if (isorpe(kk).eq.0) go to 38
      iecmplx=iecmplx+1
      if (index2(j).ne.0) go to 38
      r(j,n)=r(j,n)-s(j,i)*cm(num(6)+iecmplx)
   38 continue
   40 r(j,n)=r(j,n)-s(j,i)*cm(num(7)+i)
   41 if (num(5).eq.0) go to 60
c
c  index2 must = 3 for the concentration of the precipitate
c  to be included in the specification of that species;  however
c  if the species may form in a precipitate, than index2 =3 instead
c  of index2 = 2 should be used because of the way the equations are
c  set up at present
c
      if ((index2(j).eq.1).or.(index2(j).eq.2)) go to 60
      do 50 i=1,m5
   50 r(j,n)=r(j,n)-index1(i)*ss(j,i)*cm(num(8)+i)
 1000 format (10x,i5,10x,10(4(1pe12.4),/))
   60 continue
      if (lnh.eq.0) go to 72
c
c  the dissociation of water is included if it has been specified
c
      r(num(1),n)=h2ok(n)-cm(lnh)*ac(lnh,n)*cm(num(1))*ac(num(1),n)
   72 continue
c
c  it is possible to use a charge balance to specify one of the
c  concentrations
c
      if (lchrg.eq.0) go to 77
      r(lchrg,n)=ctot
      do 75 j=1,m1
   75 r(lchrg,n)=r(lchrg,n)-cm(j)*vj(j)
      if (num(4).eq.0) go to 77
      do 76 i=1,m4
   76 r(lchrg,n)=r(lchrg,n)-cm(num(7)+i)*vj(num(1)+i)
c
c  this is for a special condition of leaching of a solid so that
c  the increase in the amount of the species in the aqueous phase
c  making up the solid is in equal amounts; occurs where a solid has
c  been placed in contact with the groundwater.  it can only occur
c  at the first node at present
c
   77 if ((n.ne.nb).or.(leach1(1).eq.0)) go to 80
      r(leach1(2),n)=wconst(leach1(2))-cm(num(8)+leach1(1))
      index1(leach1(1))=1
      ileach=3
      j1=leach1(2)+1
      do 78 j=j1,m1
      if (j.ne.leach1(ileach)) go to 78
```

```
      r(leach1(ileach),n)=wconst(leach1(ileach))-r(leach1(ileach),n)-
     1 pini(ileach-1)-(wconst(leach1(2))-r(leach1(2),n)-
     2 pini(1))
      ileach=ileach+1
   78 continue
   80 continue
c
c  calculation of the residues of the site constraint, and the
c  mass action equations for sorption and formation of complexes
c  in the aqueous phase
c
c-----caution for transferring cm to fill co in agebrc-------
      call algebrc(num,cm,cm,r,n,n,25,elconc,lnh,1,jcob)
c
      if (num(5).eq.0) go to 90
c
c  for any precipitates, include the residues for the solubility product
c  equations
c
      call ressol(index1,cm,num,ntot,r,n,mdim)
   90 nnn=ntot+nsol
      do 100 i=1,nnn
  100 re(i)= r(i,n)
      return
      end
      subroutine ressol(index1,cm,num,ntot,r,n,mdim)
c
c  ***********************************************************
c  residues for the solubility constraint equations; it is
c  only used for the initial equilibrium distribuion of species
c  calculation; the subroutine res also has the solubility
c  product equations for the time changes at the nodes after
c  a precipitate has formed
c     *******************************************************
      dimension cm(1),index1(1),r(mdim,1),num(12)
      common/ab/vj(75),ac(75,100),eqconst(25,100),h2ok(100),alfa(4,100),
     1s(25,25),ss(25,25),ah(25),h2oc(8),con(25,8),conk(20,8),conp(4,2),
     2 ilbli(25),ilble(25),q(10,100)
      index=0
      m5=num(5)
      m1=num(1)
      do 10 i=1,m5
      aksol=1.
      if (index1(i).eq.0) go to 10
      index=index+1
      do 20 j=1,m1
      if (ss(j,i).lt.1.e-15) go to 20
      aksol=aksol*(ac(j,n)*cm(j))**ss(j,i)
   20 continue
      r(ntot+index,n)=aksol-eqconst(num(12)+num(4)+i,n)
c     print 100,r(ntot+index,n),aksol,eqconst(num(12)+num(4)+i,n)
  100 format(3e12.4)
   10 continue
      return
      end
      subroutine redl(conc,residue,np,n,elconc,lnh,num)
c
c  *****************************************************************
c  calculates the residues for the mass action equations, and a site
c  constraint equation for the surface complexation model for sorption;
```

```
c  the actual surface complexation model is described in                 redl.7
c  detail in the paper by j.a. davis, r.o. james, and j.o. leckie        redl.8
c  in j. colloid and interface science, vol 63, no. 3, march, 1978       redl.9
c  ******************************************************************** redl.10
      dimension conc(1),residue(1),num(1)                                redl.11
      common/ab/vj(75),ac(75,100),eqconst(25,100),h2ok(100),alfa(4,100), /ab/.2
     1s(25,25),ss(25,25),ah(25),h2oc(8),con(25,8),conk(20,8),conp(4,2),  /ab/.3
     2 ilbli(25),ilble(25),q(10,100)                                     /ab/.4
      common/edl/c1,c2,area,soh                                         /edl/.2
      common/edl2/sigo,sigb,sigd,psio,psibeta,psid                     /edl2/.2
      data elec/1.602e-19/, avogad/6.025e23/, boltz/1.38e-23/            redl.15
      data dielec/78.84/                                                 redl.16
      data temp/298.15/,faraday/9.6487e4/,epsil/8.85e-12/                redl.17
      arcsinh(x)=alog(x+sqrt(x*x+1))                                     redl.18
      kk=num(6)                                                          redl.19
      m3=num(3)                                                          redl.20
c                                                                        redl.21
c  calculate the charge developed at the surface (sigo) and              redl.22
c  the beta plane (sigb) for the given concentration of sorbed           redl.23
c  species                                                               redl.24
c                                                                        redl.25
      sigo=conc(kk+2+np)-conc(kk+1+np)                                   redl.26
      sigb=0.                                                            redl.27
      if (num(3).eq.3) go to 115                                         redl.28
      do 110 i=4,m3                                                      redl.29
      jj=ilble(i)                                                        redl.30
      if (jj.gt.num(1)) jj=jj-num(12)                                    redl.31
      sigo=sigo - ah(i)*conc(kk+i+np)                                    redl.32
      sigb=sigb+vj(jj)*conc(kk+i+np)                                     redl.33
  110 continue                                                           redl.34
  115 continue                                                           redl.35
 1000 format (4x,5(1pe12.4))                                             redl.36
      sigo=sigo*1.e6*faraday/area                                        redl.37
      sigb=sigb*1.e6*faraday/area                                        redl.38
      sigd=-sigo-sigb                                                    redl.39
      a1=arcsinh(abs(sigd)/(8.*avogad*epsil*boltz*                       redl.40
     1dielec*temp*1.e7*elconc)**0.5)                                     redl.41
c                                                                        redl.42
c  calculate the potentials developed between the surface plane, beta    redl.43
c  plane, diffuse plane, and the bulk solution                           redl.44
c                                                                        redl.45
      psid=-(sigd/abs(sigd))*(2.*boltz*temp/elec)*a1                     redl.46
      psibeta=psid-sigd/c2                                               redl.47
      psio=psibeta+sigo/c1                                               redl.48
      aconst=elec/(boltz*temp)                                           redl.49
      surpln=exp(-aconst*psio)                                           redl.50
      betapln=exp(-aconst*psibeta)                                       redl.51
c                                                                        redl.52
c  site constraint equation for all the sites (soh, soh-, soh2+)         redl.53
c                                                                        redl.54
      residue(1)=soh-conc(kk+1+np)-conc(kk+2+np)-conc(kk+3+np)           redl.55
      if (num(3).lt.4) go to 120                                         redl.56
      do 125 i=4,m3                                                      redl.57
      residue(1)=residue(1)-conc(kk+i+np)                                redl.58
  125 continue                                                           redl.59
  120 continue                                                           redl.60
c                                                                        redl.61
c  mass action equation describing the dissociation of soh2+ site        redl.62
c                                                                        redl.63
      residue(2)=eqconst(num(2)+1,n)*conc(kk+2+np)/surpln-conc(kk+3+np)* redl.64
```

```
     1conc(lnh+np)
c
c  mass action equation describing the dissociation of the soh site
c
      residue(3)=eqconst(num(2)+2,n)*conc(kk+3+np)/surpln-conc(kk+1+np)*
     1conc(lnh+np)
      if (num(3).lt.4) go to 126
c
c  mass action equations for the formation of sites (so-species) or
c  (soh2+ -species)
c
      do 127 i=4,m3
      jj=ilble(i)
      j2=jj
      if (jj.gt.num(1)) j2=jj-num(12)
      quan=conc(kk+i+np)*(conc(lnh+np)*surpln)**ah(i)
      denom=conc(jj+np)*conc(kk+3+np)*betapln**vj(j2)
      residue(i)=(denom)-quan/eqconst(num(2)+i,n)
  127 continue
  126 continue
      return
      end
      subroutine decbt(m,n,mdim,a,b,c,ip,ier)
c
c     ******************************************************************
c   this set of four subroutines are to invert the block tri-diagonal
c   matrix to get the new guesses for the concentrations.  the
c   subroutines involved here are decbt, dec, sol, and solbt and were
c   written by a.c. hindmarsh of lawrence livermore laboratory.  these
c   subroutines are covered in a separate document ucid-30150
c   "solution of block-tridiagonal systems of linear algebraic equations
c     ******************************************************************
      integer m,n,mdim,ip(mdim,n),ier
      dimension a(mdim,mdim,n),b(mdim,mdim,n),c(mdim,mdim,n)
      real dp
      if (m .lt. 1 .or. n .lt. 4) go to 210
      nm1 = n - 1
      nm2 = n - 2
      call dec(m,mdim,a,ip,ier)
      k = 1
      if (ier .ne. 0) go to 200
      do 10 j = 1,m
      call sol(m,mdim,a,b(1,j,1),ip)
      call sol(m,mdim,a,c(1,j,1),ip)
 10     continue
      do 40 j = 1,m
        do 30 i = 1,m
          dp = 0.
          do 20 l = 1,m
 20         dp = dp + c(i,l,2)*c(l,j,1)
          b(i,j,2) = b(i,j,2) - dp
 30       continue
 40     continue
      do 100 k = 2,nm1
        km1 = k - 1
        do 70 j = 1,m
          do 60 i = 1,m
            dp = 0.
            do 50 l = 1,m
 50           dp = dp + c(i,l,k)*b(l,j,km1)
```

```
            a(i,j,k) = a(i,j,k) - dp                                       decbt.40
 60         continue                                                       decbt.41
 70       continue                                                         decbt.42
      call dec(m,mdim,a(1,1,k),ip(1,k),ier)                                decbt.43
        if (ier .ne. 0) go to 200                                          decbt.44
        do 80 j = 1,m                                                      decbt.45
 80       call sol(m,mdim,a(1,1,k),b(1,j,k),ip(1,k))                       decbt.46
 100    continue                                                           decbt.47
      do 130 j = 1,m                                                       decbt.48
        do 120 i = 1,m                                                     decbt.49
          dp = 0.                                                          decbt.50
          do 110 l = 1,m                                                   decbt.51
 110        dp = dp + b(i,l,n)*b(l,j,nm2)                                  decbt.52
          c(i,j,n) = c(i,j,n) - dp                                         decbt.53
 120      continue                                                         decbt.54
 130    continue                                                           decbt.55
      do 160 j = 1,m                                                       decbt.56
        do 150 i = 1,m                                                     decbt.57
          dp = 0.                                                          decbt.58
          do 140 l = 1,m                                                   decbt.59
 140        dp = dp + c(i,l,n)*b(l,j,nm1)                                  decbt.60
          a(i,j,n) = a(i,j,n) - dp                                         decbt.61
 150      continue                                                         decbt.62
 160    continue                                                           decbt.63
      call dec(m,mdim,a(1,1,n),ip(1,n),ier)                                decbt.64
      k = n                                                                decbt.65
      if (ier .ne. 0) go to 200                                            decbt.66
      return                                                               decbt.67
 200  ier = k                                                              decbt.68
      return                                                               decbt.69
 210  ier = -1                                                             decbt.70
      return                                                               decbt.71
      end                                                                  decbt.72
      subroutine solbt(m,n,mdim,a,b,c,y,ip)                                solbt.2
c                                                                          solbt.3
      dimension y(mdim,n)                                                  solbt.4
      integer m,n,ip(mdim,n),mdim                                          solbt.5
      dimension a(mdim,mdim,n),b(mdim,mdim,n),c(mdim,mdim,n)               solbt.6
      real dp                                                              solbt.7
      nm1 = n - 1                                                          solbt.8
      nm2 = n - 2                                                          solbt.9
      call sol(m,mdim,a,y,ip)                                              solbt.10
      do 30 k = 2,nm1                                                      solbt.11
        km1 = k - 1                                                        solbt.12
        do 20 i = 1,m                                                      solbt.13
          dp = 0.                                                          solbt.14
          do 10 j = 1,m                                                    solbt.15
 10         dp = dp + c(i,j,k)*y(j,km1)                                    solbt.16
          y(i,k) = y(i,k) - dp                                             solbt.17
 20       continue                                                         solbt.18
      call sol(m,mdim,a(1,1,k),y(1,k),ip(1,k))                             solbt.19
 30     continue                                                           solbt.20
      do 50 i = 1,m                                                        solbt.21
        dp = 0.                                                            solbt.22
        do 40 j = 1,m                                                      solbt.23
 40       dp = dp + c(i,j,n)*y(j,nm1) + b(i,j,n)*y(j,nm2)                  solbt.24
        y(i,n) = y(i,n) - dp                                               solbt.25
 50     continue                                                           solbt.26
      call sol(m,mdim,a(1,1,n),y(1,n),ip(1,n))                             solbt.27
      do 80 kb = 1,nm1                                                     solbt.28
```

```
        k = n - kb                                                    solbt.29
        kp1 = k + 1                                                   solbt.30
        do 70 i = 1,m                                                 solbt.31
          dp = 0.                                                     solbt.32
          do 60 j = 1,m                                               solbt.33
 60         dp = dp + b(i,j,k)*y(j,kp1)                               solbt.34
          y(i,k) = y(i,k) - dp                                        solbt.35
 70       continue                                                    solbt.36
 80     continue                                                      solbt.37
      do 100 i = 1,m                                                  solbt.38
        dp = 0.                                                       solbt.39
        do 90 j = 1,m                                                 solbt.40
 90       dp = dp + c(i,j,1)*y(j,3)                                   solbt.41
        y(i,1) = y(i,1) - dp                                          solbt.42
 100    continue                                                      solbt.43
      return                                                          solbt.44
      end                                                             solbt.45
      subroutine dec (n, ndim, a, ip, ier)                              dec.2
c                                                                       dec.3
      dimension  ip(n)                                                  dec.4
      dimension a(ndim,n)                                               dec.5
      ier = 0                                                           dec.6
      ip(n) = 1                                                         dec.7
      if (n .eq. 1) go to 70                                            dec.8
      nm1 = n - 1                                                       dec.9
      do 60 k = 1,nm1                                                   dec.10
        kp1 = k + 1                                                     dec.11
        m = k                                                           dec.12
        do 10 i = kp1,n                                                 dec.13
 10       if (abs(a(i,k)) .gt. abs(a(m,k))) m = i                       dec.14
        ip(k) = m                                                       dec.15
        t = a(m,k)                                                      dec.16
        if (m .eq. k) go to 20                                          dec.17
        ip(n) = -ip(n)                                                  dec.18
        a(m,k) = a(k,k)                                                 dec.19
        a(k,k) = t                                                      dec.20
 20     if (t .eq. 0.) go to 80                                         dec.21
        t = 1./t                                                        dec.22
        do 30 i = kp1,n                                                 dec.23
 30       a(i,k) = -a(i,k)*t                                            dec.24
        do 50 j = kp1,n                                                 dec.25
          t = a(m,j)                                                    dec.26
          a(m,j) = a(k,j)                                               dec.27
          a(k,j) = t                                                    dec.28
          if (t .eq. 0.) go to 50                                       dec.29
          do 40 i = kp1,n                                               dec.30
 40         a(i,j) = a(i,j) + a(i,k)*t                                  dec.31
 50       continue                                                      dec.32
 60     continue                                                        dec.33
 70   k = n                                                             dec.34
      if (a(n,n) .eq. 0.) go to 80                                      dec.35
      return                                                            dec.36
 80   ier = k                                                           dec.37
      ip(n) = 0                                                         dec.38
      return                                                            dec.39
      end                                                               dec.40
      subroutine sol (n, ndim, a, b, ip)                                sol.2
c                                                                       sol.3
      dimension b(n), ip(n), a(ndim,n)                                  sol.4
      if (n .eq. 1) go to 50                                            sol.5
```

```
      nm1 = n - 1                                                          sol.6
      do 20 k = 1,nm1                                                      sol.7
        kp1 = k + 1                                                        sol.8
        m = ip(k)                                                          sol.9
        t = b(m)                                                           sol.10
        b(m) = b(k)                                                        sol.11
        b(k) = t                                                           sol.12
        do 10 i = kp1,n                                                    sol.13
 10       b(i) = b(i) + a(i,k)*t                                           sol.14
 20     continue                                                           sol.15
      do 40 kb = 1,nm1                                                     sol.16
        km1 = n - kb                                                       sol.17
        k = km1 + 1                                                        sol.18
        b(k) = b(k)/a(k,k)                                                 sol.19
        t = -b(k)                                                          sol.20
        do 30 i = 1,km1                                                    sol.21
 30       b(i) = b(i) + a(i,k)*t                                           sol.22
 40     continue                                                           sol.23
 50   b(1) = b(1)/a(1,1)                                                   sol.24
      return                                                               sol.25
      end                                                                  sol.26
      subroutine shiftb(cm,num,alfa,shiftr)                             shiftb.2
c                                                                       shiftb.3
c      ******************************************************           shiftb.4
cc     this routine is only called to calculate the initial condition   shiftb.5
c      of recharging groundwater.only used in static mode.              shiftb.6
c      ***************************************************************  shiftb.7
      dimension cm(25),num(1),alfa(4,100)                               shiftb.8
c                                                                       shiftb.9
c     calculate del0,del1,del2,del3                                     shiftb.10
c                                                                       shiftb.11
      shift1=alfa(1,1)*shiftr+1000.0*(alfa(1,1)-1.0)                    shiftb.12
      shift2=alfa(2,1)*shiftr+1000.0*(alfa(2,1)-1.0)                    shiftb.13
      shift3=alfa(3,1)*shiftr+1000.0*(alfa(3,1)-1.0)                    shiftb.14
      shift4=alfa(4,1)+shiftr+1000.0*(alfa(4,1)-1.0)                    shiftb.15
      cm1=cm(num(7)+1)                                                  shiftb.16
      cm2=cm(num(7)+2)                                                  shiftb.17
      cm3=cm(num(7)+3)                                                  shiftb.18
      xmc=cm1+cm2+cm3                                                   shiftb.19
      write(6,400)                                                      shiftb.20
      print 500, (alfa(i,1), i=1,4)                                     shiftb.21
      print 600,shiftr, cm1, cm2, cm3                                   shiftb.22
      shftsol=( cm1*shift1+cm2*shift2+cm3*shift3 )/xmc                  shiftb.23
      print 700, xmc,shftsol,shift1,shift2,shift3,shift4                shiftb.24
  400 format(//,'** Recharge water stable isotop characteristics **')   shiftb.25
  500 format(/'Alfa(1) = 'e12.4,5x,'Alfa(2) = 'e12.4,5x,'Alfa(3) = '    shiftb.26
     1         e12.4,5x,'Alfa(4) = 'e12.4)                              shiftb.27
  600 format(/'shiftr = 'e12.3,5x,'H2CO3 = 'e12.3,5x,'HCO3- = '         shiftb.28
     1 e12.3,5x,'CO3-- = 'e12.3)                                        shiftb.29
  700 format(/'xmc = 'e12.3,5x,'shftsol = 'e12.3,5x,'shift1 = 'e12.3,   shiftb.30
     1         5x,'shift2 = 'e12.3,5x,'shift3 = 'e12.3,5x,              shiftb.31
     2         'shifts = 'e12.3)                                        shiftb.32
c                                                                       shiftb.33
      return                                                            shiftb.34
      end                                                               shiftb.35
      subroutine shiftc(cn,num,shftsol,shift,shiftr,shifts,             shiftc.2
     1                  alfa,nmax,kcode )                               shiftc.3
c                                                                       shiftc.4
c      ***************************************************************** shiftc.5
c      this subroutine calculate variations of del0,del1,del2,del3 as   shiftc.6
```

```
c       the recharging groundwater disolves the carbonate rocks.          shiftc.7
c      *******************************************************************shiftc.8
c                                                                         shiftc.9
      dimension cn(1),shftsol(1),shift(4,1),num(1),alfa(4,1)              shiftc.10
      dimension kcode(1)                                                  shiftc.11
c                                                                         shiftc.12
      do 100 n=1,nmax                                                     shiftc.13
      nstart=(n-1)*num(9)                                                 shiftc.14
      do 200 j=1,num(8)                                                   shiftc.15
      if( kcode(j).le.30) go to 200                                       shiftc.16
      if( kcode(j).eq.31) xmco3=cn(nstart+j)                              shiftc.17
      if( kcode(j).eq.32) xmki =cn(nstart+j)                              shiftc.18
      if( kcode(j).eq.33) xmhco3=cn(nstart+j)                             shiftc.19
      if( kcode(j).eq.34) xmh2co3=cn(nstart+j)                            shiftc.20
  200 continue                                                            shiftc.21
      cn3=xmco3                                                           shiftc.22
      cn4=xmki                                                            shiftc.23
      cnp=xmhco3                                                          shiftc.24
      cnp1=xmh2co3                                                        shiftc.25
      xmc=cn3+cnp+cnp1                                                    shiftc.26
      shftsol(n)=shifts-cn4/xmc                                           shiftc.27
      alfr2=alfa(2,n)/alfa(1,n)                                           shiftc.28
      alfr3=alfa(3,n)/alfa(1,n)                                           shiftc.29
      rxmc=cnp+alfr2*cnp1+alfr3*cn3                                       shiftc.30
      shift(1,n)=(xmc*shftsol(n)-1000.0*(cnp1*(alfr2-1.)+cn3*(alfr3-1.)   shiftc.31
     1 ))/rxmc                                                            shiftc.32
      shift(2,n)=alfr2*shift(1,n)+1000.0*(alfr2-1.0)                      shiftc.33
      shift(3,n)=alfr3*shift(1,n)+1000.0*(alfr3-1.0)                      shiftc.34
  100 continue                                                            shiftc.35
      return                                                              shiftc.36
      end                                                                 shiftc.37
      subroutine eqcon(n1,n2,con,conk,h2oc,m15,kinr,lnh,tmp,h2ok,eqconst   eqcon.2
     1                   ,nsrt,reactsk,alfa,conp,ipart,kt)                 eqcon.3
c                                                                          eqcon.4
c  **********************************************************************  eqcon.5
c  subroutine calculates the kinetic dissolution constants of solids       eqcon.6
c  ,C-13 partition constants,the dissociation of water constants,the       eqcon.7
c  formation constants of aqueous complexes, and the solubility products   eqcon.8
c  of solids at the temperature tmp. the constants are calculated from     eqcon.9
c  expressions of the form                                                 eqcon.10
c                                                                          eqcon.11
c              log k = c1 + c2/tk + c3*(log tk) + c4*tk                    eqcon.12
c                                                                          eqcon.13
c  where tk = tmp + 273.15.                                                eqcon.14
c  **********************************************************************  eqcon.15
c                                                                          eqcon.16
      implicit real*8 (a-h,o-z)                                            eqcon.17
      dimension con(25,8),h2oc(8),eqconst(25,100),h2ok(100),tmp(100),      eqcon.18
     1   conk(20,8),conp(4,2),alfa(4,100),reactsk(5,10,100),               eqcon.19
     2   nsrt(5)                                                           eqcon.20
c                                                                          eqcon.21
      do 40 node=n1,n2                                                     eqcon.22
      tk=tmp(node)+273.15                                                  eqcon.23
      if(kinr.eq.0) go to 20                                               eqcon.24
      if(kt.ne.1) go to 20                                                 eqcon.25
      ktsum=0                                                              eqcon.26
      do 12 i=1,kinr                                                       eqcon.27
      do 11 j=1,nsrt(i)                                                    eqcon.28
      l=j+ktsum                                                            eqcon.29
      term=conk(l,1)+conk(l,2)/tk+conk(l,3)*alog10(tk)+conk(l,4)*tk        eqcon.30
```

```
      reactsk(i,j,node)=10.**term                                         eqcon.31
   11 continue                                                            eqcon.32
      ktsum=ktsum+nsrt(i)                                                 eqcon.33
   12 continue                                                            eqcon.34
c                                                                         eqcon.35
   20 continue                                                            eqcon.36
      if(ipart.eq.0) go to 18                                             eqcon.37
      do 15 j=1,4                                                         eqcon.38
      term=conp(j,1)+conp(j,2)/(tk**2)                                    eqcon.39
      alfa(j,node)=exp(term)                                              eqcon.40
   15 continue                                                            eqcon.41
c                                                                         eqcon.42
   18 continue                                                            eqcon.43
      if (lnh.eq.0) go to 13                                              eqcon.44
      term=h2oc(1)+h2oc(2)/tk+h2oc(3)*alog10(tk)+h2oc(4)*tk               eqcon.45
      h2ok(node)=10.**term                                                eqcon.46
   13 continue                                                            eqcon.47
      do 30 j=1,m15                                                       eqcon.48
      term=con(j,1)+con(j,2)/tk+con(j,3)*alog10(tk)+con(j,4)*tk           eqcon.49
      eqconst(j,node)=10.**term                                           eqcon.50
c     print 100, node, j, eqconst(j,node)                                 eqcon.51
  100 format(2i10,10x,e12.3)                                              eqcon.52
   30 continue                                                            eqcon.53
   40 continue                                                            eqcon.54
      return                                                              eqcon.55
      end                                                                 eqcon.56
c                                                                         eqcon.57
      subroutine gradt(nmax,tempi,tempo,itype,x,tmp)                      gradt.2
c                                                                         gradt.3
c  *********************************************************************  gradt.4
c  this subroutine calculates the fixed temperature gradient when         gradt.5
c  itemp = 2.  steady-state expressions for diffusion of heat in 1-d      gradt.6
c  and radial geometries with fixed temperatures at inner and outer       gradt.7
c  boundaries are used.                                                   gradt.8
c  *********************************************************************  gradt.9
c                                                                         gradt.10
      implicit real*8 (a-h,o-z)                                           gradt.11
      dimension x(101),tmp(101)                                           gradt.12
c                                                                         gradt.13
      xi=x(1)                                                             gradt.14
      xo=x(nmax)                                                          gradt.15
      dt=tempi-tempo                                                      gradt.16
      tmp(1)=tempi                                                        gradt.17
      tmp(nmax)=tempo                                                     gradt.18
      if (itype.eq.1) const=alog(xo/xi)                                   gradt.19
      nm=nmax-1                                                           gradt.20
      do 30 n=2,nm                                                        gradt.21
      xx=x(n)                                                             gradt.22
      if (itype.eq.1) go to 10                                            gradt.23
      xterm=xx/xo                                                         gradt.24
      go to 20                                                            gradt.25
   10 xterm=(alog(xx/xi))/const                                           gradt.26
   20 tmp(n)=tempi-dt*xterm                                               gradt.27
   30 continue                                                            gradt.28
      return                                                              gradt.29
      end                                                                 gradt.30
c                                                                         gradt.31
      subroutine tmpcal(ibnd2,nmax,hcm,hcw,vo,tcm,doa,itype,tb,delt,x,   tmpcal.2
     1tmpold,tmp)                                                        tmpcal.3
c                                                                        tmpcal.4
```

```
c  ********************************************************************
c  this subroutine calculates the temperature at each node for each new
c  time step, using the crank-nicholson method.
c  it solves the tridiagonal system of equations (a)(tmp)=(f), where
c  (a) is the coefficient matrix, by lu decomposition of (a) such that
c  (l)(u)(tmp)=(f).
c  if (a) has the structure
c
c                       (aa1 cc1  0   ... ... ...)
c                       (bb2 aa2 cc2  0   ... ...)
c                       ( 0  bb3 aa3 cc3  0   ...),
c                       (... ... ... ... ... ...)
c                       (... ... ...  0  bbn aan)
c
c  then (l) has the structure
c
c                       ( 1   0  ... ... ... ...)
c                       (be2  1   0  ... ... ...)
c                       ( 0  be3  1   0  ... ...)
c                       (... ... ... ... ... ...)
c                       (... ... ...  0  ben  1 )
c
c  and (u) has the structure
c
c                       (al1 cc1  0  ... ... ...)
c                       ( 0  al2 cc2  0  ... ...)
c                       (...  0  al3 cc3  0  ...).
c                       (... ... ... ... ... ...)
c                       (... ... ... ...  0  aln)
c
c  the forward process solves (l)(g)=(f) for vector (g).
c  the backward process solves (u)(tmp)=(g) for the temperatures (tmp).
c  the solution algorithm was furnished by janet remer.
c  ********************************************************************
c
      implicit real*8 (a-h,o-z)
      dimension aa(100),bb(100),cc(100),al(100),be(100),f(100),g(100),
     1x(100),tmpold(100),tmp(100)
c
c  calculate matrix of coefficients (a) and vector of knowns (f).
c
      nm=nmax-1
      hcmw=hcm+hcw
      vv=vo*hcw/hcmw
      dif=tcm/hcmw
      av=vv*doa
      cap=av+dif
      vm=vv
      vb=vv
c
c     inner boundary
c
      x1=x(1)
      x2=x(2)
      x3=x(3)
      x21=x2-x1
      x31=x3-x1
      x32=x3-x2
      if (ibnd2.eq.1) go to 10
      nin=2
```

```
      if (itype.eq.1) cap=dif+av/x2
      if (itype.eq.1) vm=(vv-dif)/x2
      t1=delt*(cap+vm*x32/2.)/(x31*x21)
      t2=delt*(cap*(1./x32+1./x21)+(vm/2.)*(x32/x21-x21/x32))/x31
      t3=delt*(cap-vm*x21/2.)/(x31*x32)
      aa(2)=1.+t2
      cc(2)=-t3
      f(2)=t1*tb+(1.-t2)*tmpold(2)+t1*tmpold(1)+t3*tmpold(3)
      go to 20
   10 nin=1
      if (itype.eq.1) cap=dif+av/x1
      if (itype.eq.1) vm=(vv-dif)/x1
      if (itype.eq.1) vb=vv/x1
      aa(1)=1.+delt*(cap/(x21*x21)+vb/x21+vm*vb/(2.*cap))
      cc(1)=-cap*delt/(x21*x21)
      f(1)=tmpold(1)+(2.+vm*x21/cap)*vb*tb*delt/x21+cap*delt*(tmpold(2)-
     1tmpold(1))/(x21*x21)-(1.+vm*x21/(2.*cap))*vb*delt*tmpold(1)/x21
c
c     interior nodes
c
   20 ninp=nin+1
      xn=x(nin)
      xp=x(ninp)
      do 30 n=ninp,nm
      xm=xn
      xn=xp
      xp=x(n+1)
      xnm=xn-xm
      xpm=xp-xm
      xpn=xp-xn
      if (itype.eq.1) cap=dif+av/xn
      if (itype.eq.1) vm=(vv-dif)/xn
      tm=delt*(cap+vm*xpn/2.)/(xpm*xnm)
      tn=delt*(cap*(1./xpn+1./xnm)+(vm/2.)*(xpn/xnm-xnm/xpn))/xpm
      tp=delt*(cap-vm*xnm/2.)/(xpm*xpn)
      bb(n)=-tm
      aa(n)=1.+tn
      cc(n)=-tp
      f(n)=tm*tmpold(n-1)+(1.-tn)*tmpold(n)+tp*tmpold(n+1)
      if (n.eq.nm) f(n)=f(n)+tp*tmpold(nmax)
   30 continue
c
c  calculate decomposition matrices
c
      alk=aa(nin)
      al(nin)=alk
      do 40 k=ninp,nm
      bek=bb(k)/alk
      alk=aa(k)-bek*cc(k-1)
      al(k)=alk
      be(k)=bek
   40 continue
c
c  forward substitution
c
      gi=f(nin)
      g(nin)=gi
      do 50 i=ninp,nm
      gi=f(i)-be(i)*gi
      g(i)=gi
```

```
   50 continue                                                          tmpcal.125
c                                                                       tmpcal.126
c  backward substitution                                                tmpcal.127
c                                                                       tmpcal.128
      ti=g(nm)/al(nm)                                                   tmpcal.129
      tmp(nm)=ti                                                        tmpcal.130
      nmm=nm-nin                                                        tmpcal.131
      do 60 j=1,nmm                                                     tmpcal.132
      i=nm-j                                                            tmpcal.133
      ti=(g(i)-cc(i)*ti)/al(i)                                          tmpcal.134
      tmp(i)=ti                                                         tmpcal.135
   60 continue                                                          tmpcal.136
      return                                                            tmpcal.137
      end                                                               tmpcal.138
c                                                                       tmpcal.139
```

LAWRENCE BERKELEY LABORATORY
TECHNICAL INFORMATION DEPARTMENT
UNIVERSITY OF CALIFORNIA
BERKELEY, CALIFORNIA 94720