NUREG/CR-6138
EGG-2719

User's Guide for Simplified Computer Models for the Estimation of Long-Term Performance of Cement-Based Materials

Prepared by
L. E. Plansky, R. R. Seitz

Idaho National Engineering Laboratory
EG&G Idaho, Inc.

Prepared for
U.S. Nuclear Regulatory Commission

NUREG/CR-6138
EGG-2719

User's Guide for Simplified Computer Models for the Estimation of Long-Term Performance of Cement-Based Materials

Prepared by
L. E. Plansky, R. R. Seitz

Idaho National Engineering Laboratory
EG&G Idaho, Inc.

Prepared for
U.S. Nuclear Regulatory Commission

NUREG/CR–6138
EGG–2719

User's Guide for Simplified Computer Models for the Estimation of Long-Term Performance of Cement-Based Materials

Manuscript Completed: December 1993
Date Published: February 1994

Prepared by
L. E. Plansky, R. R. Seitz

Idaho National Engineering Laboratory
Managed by the U.S. Department of Energy

EG&G Idaho, Inc.
Idaho Falls, ID 83415

Prepared for
Division of Regulatory Applications
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, DC 20555–0001
NRC FIN A6858
Under DOE Contract No. DE–AC07–76ID01570

ABSTRACT

This report documents user instructions for several simplified subroutines and driver programs that can be used to estimate various aspects of the long-term performance of cement-based barriers used in low-level radioactive waste disposal facilities. The subroutines are prepared in a modular fashion to allow flexibility for a variety of applications. Three levels of codes are provided: the individual subroutines, interactive drivers for each of the subroutines, and an interactive main driver, CEMENT, that calls each of the individual drivers. The individual subroutines for the different models may be taken independently and used in larger programs, or the driver modules can be used to execute the subroutines separately or as part of the main driver routine. A brief program description is included and user-interface instructions for the individual subroutines are documented in the main report. These are intended to be used when the subroutines are used as subroutines in a larger computer code. User instructions for the drivers and example interactive screens for the main drivers are provided in Appendix A. Examples showing the use of the individual driver routines to execute the different subroutines and test data are included in Appendix B. Programmer notes are provided in Appendix C and the configuration control system is discussed in Appendix D.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

This document provides user instructions for the computer models done in support of the U.S. Nuclear Regulatory Commission program: Performance of Concrete Barriers in Low-Level Waste Disposal, FIN Number A6858.

The computer models are described and user instructions for each of the models are given. Separate and integrated drivers are provided. This permits the models to be run alone, in a series, or as an integrated package. The larger model "CEMENT" represents the latter case. CEMENT is an example of a larger program integrating the individual models into a single program. In this way, the individual models can be used as support subroutines in a larger main computer analysis code or individually, providing maximum flexibility to the analyst. The present version of CEMENT is version 1.01 as described in this user's guide.

FOREWORD

This technical report is a product of the Idaho National Engineering Laboratory under the performance of Concrete Barriers in Low-Level Waste Disposal Project (FIN A6858). The purpose of this research is to provide support to the U.S. Nuclear Regulatory Commission (NRC) staff in their evaluation of degradation, flow, and transport through concrete barriers and transport through cracks that can eventually form in the barriers. The information enhances the capabilities of the NRC staff to conduct independent analyses, as needed, to assess specific aspects of barrier performance for license applications.

NUREG/CR-6138 is not a substitute for NRC regulations and compliance is not required. The approaches and/or methods described in this NUREG/CR are provided for information only. Publication of this report does not necessarily constitute NRC approval or agreement with the information contained herein.

User's Guide for Simplified Computer Models for the Estimation of Long-Term Performance of Cement-Based Materials

1. INTRODUCTION

The purpose of the U.S. NRC Concrete Barriers Program research at the INEL is to investigate and develop mathematical models and computer code models applicable to predicting the degradation or durability of concrete that is to be used as barriers to radionuclide migration in low-level radioactive waste (LLW) repositories. The program originally began at the Idaho National Engineering Laboratory (INEL) for the U.S. NRC in 1988 under the direction of T. J. McCartin of the U.S. NRC Division of Engineering in Washington, D.C. This document describes and provides user documentation for the computer models developed from these analyses. The mathematical models developed or investigated are described in the NUREG documents referenced in this report (Seitz and Walton 1993; Walton 1992; Walton and Seitz 1991; Walton et al. 1990).

The present version of CEMENT is version 1.01 as described in this user's guide. The computer models were developed in a modular fashion to allow a wider range of applicability. Figure 1 shows the overall computer code subroutine flow and organization. The main part of this manual documents the individual subroutines that actually do the calculations. These subroutines can be used as part of a general library of routines or used in other computer codes. Interactive driver routines were also prepared for each individual subroutine to allow a user to execute the routines separately. Finally, a main interactive driver was prepared to allow the user to run the all of the individual routines from a single platform. The main driver and the individual drivers are documented in Appendix A. Appendix A also includes example interactive screens for the main driver. Example interactive screens for each of the drivers for the individual subroutines are provided in Appendix B along with example input and output files for the different routines. Programmer notes are provided in Appendix C and the configuration control system is discussed in Appendix D.

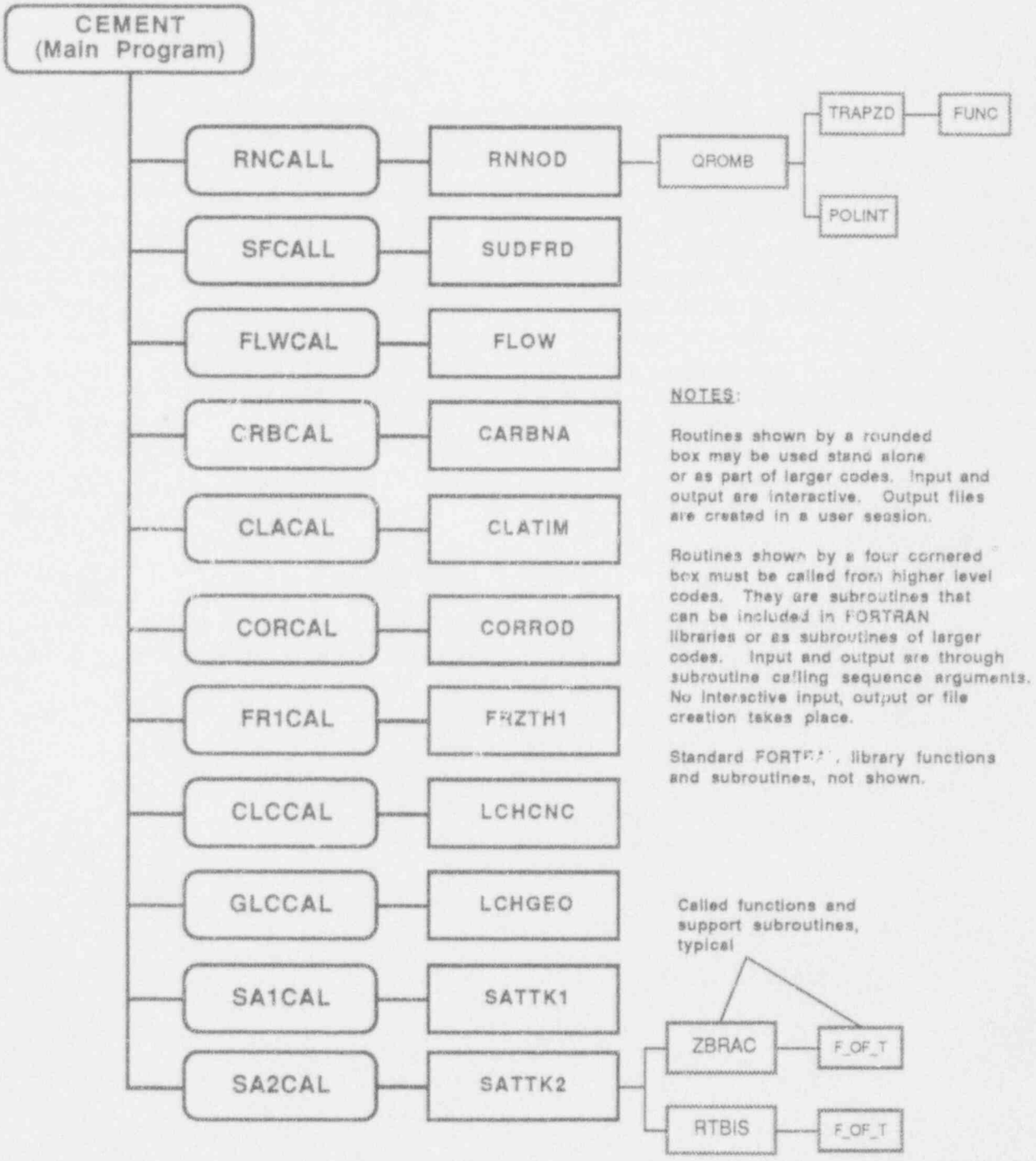

Figure 1. Cement Code Subroutine Flow.

2. THE COMPUTER CODES

Eleven computer codes, or models, were done to complement the mathematical-physical analyses completed for the U.S. NRC Concrete Barriers Program. A summary of these computer models is given below. User documentation for each model is given in Section 3.

2.1 The Subroutines

2.1.1 CARBNA. This subroutine implements the carbonation degradation model of Walton et al. (1990), Equation (5), page 23. As stated in Walton et al. (1990), the rate of carbonation is dependent on the water saturation or relative humidity of the environment; as relative humidity increases from 0 to 100%, the rate of carbonation passes through a maximum. This maximum occurs around a 50% relative humidity. Typical subsurface environments of concrete used as barriers have a 100% relative humidity and the concrete remains saturated. This is the assumption underlying the calculations in this routine. One-hundred values of deterioration depth x (depth of carbonation) versus time can be computed.

2.1.2 CLATIM. This subroutine implements the time to initiation of chloride attack model of Walton et al. (1990), Equation (1), page 15. An empirical model calculates the time to initiation of chloride attack as being proportional to the thickness of the concrete over the rebar divided by the product of the water to cement ratio and the chloride ion concentration. A single value of time to initiation of chloride attack is computed.

2.1.3 CORROD. This subroutine implements the oxygen diffusion degradation model of Walton et al. (1990), Equation (7), page 16, and the hydrogen evolution (passive) corrosion model of Seitz and Walton (1993), Equation (4), page 8. The corrosion rate is assumed to be controlled by the diffusion of oxygen in the concrete and background (passive) corrosion. Oxygen diffusion and hydrogen evolution corrode the rebar used in concrete structural reinforcement. The model computes the remaining percentage of rebar after the onset of corrosion versus time—up to 100 values versus time can be computed.

2.1.4 FRZTH1. This subroutine implements the freeze thaw degradation model of Walton et al., (1990), Equations (2), (3), (4) and (5), page 26. The model is based on laboratory tests and computes the annual rate of concrete loss, the reduction in dynamic modulus of elasticity based on the number of freeze thaw cycles, and the time to reach the level of reduction in dynamic modulus of elasticity.

2.1.5 LCHCNC. This subroutine implements the concrete controlled leaching degradation model of Walton et al. (1990), Equation (4), page 19. The rate of leaching is assumed to be controlled by diffusion in the concrete, that is, the rate of leaching from a concrete surface is rapid compared to diffusion through the concrete. One-hundred values of deterioration depth x (depth of leaching) versus time can be computed.

2.1.6 LCHGEO. This subroutine implements the geology controlled leaching degradation model of Walton et al. (1990), Equation (9), page 20. The rate of leaching is assumed to be controlled by diffusion in the local geologic environment (Walton et al. 1990). One-hundred values of deterioration depth x (depth of leaching) versus time can be computed.

2.1.7 SATTK1. This subroutine implements the empirical sulfate attack model of Walton et al. (1990), Equation (5), page 10. The model assumes that the rate of sulfate attack is proportional to the total sulfate concentration in solution and the amount of tricalcium-aluminum silicate in the cement. One-hundred values of deterioration depth x versus time can be computed in this routine.

2.1.8 SATTK2. This subroutine implements the sulfate attack degradation model of Walton et al. (1990), Equations (11–15), page 11. The model assumes sulfate ion diffusion into the concrete followed by reaction with aluminum phases and expansion, which subsequently causes stress

cracking and foliation of the concrete (Walton et al. 1990). A single value of deterioration rate in cm/yr is computed at each call to the subroutine.

2.1.9 FLOW. This subroutine implements the equation to calculate the nondimensional flow rate from Walton and Seitz (1991), Equations (5–9), pages 5–6. The nondimensional flow rate (ζ_3) is assumed to be controlled by the crack spacing and width, and the depth of perched water. The ratio of the flow rate through the cracks in the concrete to the permeability of the overlying porous media is represented by ζ_3. Thus, ζ_3 can be multiplied by the permeability of the overlying porous media to obtain an effective flow rate through the combined porous media/cracked concrete system.

2.1.10 RNNOD. This subroutine implements the equations to estimate the ratio of the contaminant concentration in the water at a specified location along a crack to the initial total contaminant concentration in the concrete matrix. The radionuclide is assumed to start in the matrix and move into the crack by diffusion. Transport through the crack is assumed to occur by advection. The model is discussed in Walton (1992), pages 10–15.

2.1.11 SUDFRD. This subroutine implements the equations to estimate the thickness of concrete required to attenuate the contaminant concentration in the leachate by a given fraction assuming specified crack width and spacing. The model is discussed in Walton (1992), pages 16–19.

3. INDIVIDUAL SUBROUTINE USER DOCUMENTATION

3.1 Subroutine User Instructions for CARBNA

This subroutine implements the carbonation degradation model of Walton et al. (1990), Equation (5), page 23. The rate of carbonation is dependent on the water saturation or relative humidity of the environment. As relative humidity increases from 0 to 100%, the rate of carbonation passes through a maximum (Walton et al. 1990). This maximum occurs around a 50% relative humidity. Typical subsurface environments of concrete used as barriers have a 100% relative humidity and the concrete remains saturated. This is the assumption underlying the calculations in this routine. One-hundred values of deterioration depth x (depth of carbonation) versus time can be computed.

3.1.1 Usage. The model is executed by a FORTRAN subroutine call:

CALL CARBNA (N, TIME, X, DI, CGWCA, CS, IERR, HELP)

The inputs and outputs to this subroutine are defined in the sections below.

3.1.2 Inputs. The subroutine has 5 inputs: N, TIME, DI, CGWCA, and CS. These inputs are described in Table 1. The variable TIME is an array N long. DI is the intrinsic diffusion coefficient and can be calculated from the fractional porosity, θ, and the effective diffusion coefficient, DE, using:

$$DI = \theta \bullet DE, \; 0 \leq \theta \leq 1.0 . \qquad (1)$$

3.1.3 Outputs. The subroutine has 2 outputs: X and IERR. X is an array containing N values of the depth of carbonation that correspond to the N values of time in the TIME array. IERR is an error flag and is described in Table 1.

3.1.4 Table of Inputs and Outputs. Table 1 describes the inputs and outputs for CARBNA.

3.1.5 Remarks. N must be input as a positive integer >0 but $\leq$100. N or CS must be >0. If any of

Table 1. Inputs and outputs for CARBNA.

FORTRAN name	NUREG symbol	Type	Units	Description	Remarks
DI	D_i	Real	cm^2/sec	Intrinsic diffusion coefficient	INPUT
CGWCA	Cgw	Real	$mols/cm^3$	Conc. of total inorganic Carbon in environment	INPUT
CS	Cs	Real	$mols/cm^3$	Bulk conc. of $Ca(OH)^2$ in solid concrete	INPUT
TIME	t	Real	Years	Array of time values	INPUT, N values
N	None	Integer	None	Number of time intervals desired	INPUT
IERR	None	Integer	None	Error flag	OUTPUT, set to "1" if input error
X	x	Real	cm	Depth of carbonation at time, t, an array	OUTPUT, N values

these conditions are not met, IERR will be set to 1 and no calculations will be made. Zero is a valid value for the remaining variables. HELP is a LOGICAL * 4 flag intended for future use to control diagnostic printout, and is presently not used by this routine. ND denotes "nondimensional."

3.2 Subroutine User Instructions for CLATIM

3.2.1 Description. This subroutine implements the time to initiation of chloride attack model of Walton et al. (1990), Equation (1), page 15. An empirical model calculates the time to initiation of chloride attack as being proportional to the thickness of the concrete over the rebar divided by the product of the water to cement ratio and the chloride ion concentration. A single value of time to initiation of chloride attack is computed.

3.2.2 Usage. The model is executed by a FORTRAN subroutine call:

```
CALL CLATIM (XC, WCR, CLCLION,
TC, IERR, HELP)
```

The inputs and outputs to this subroutine are defined in the sections below.

3.2.3 Inputs. The subroutine has 3 inputs: XC, WCR, and CLCLION as described in Table 2.

3.2.4 Outputs. The subroutine has 2 outputs: TC and IERR. TC is the time to initiation of chloride attack and IERR is an error flag; both are described in Table 2.

3.2.5 Table of Inputs and Outputs. Table 2 describes the inputs and outputs for this routine.

3.2.6 Remarks. WCR, CLCLION, and XC must be input as positive integers greater than zero. If any of these conditions are not met, IERR will be set to 1 and no calculations will be made. HELP is a LOGICAL * 4 flag intended for future use to control diagnostic printout, and is presently not used by this routine. ND denotes "nondimensional."

3.3 Subroutine User Instructions for CORROD

3.3.1 Description. This subroutine implements the oxygen diffusion degradation model of Walton et al. (1990), Equation (7), page 16, and the hydrogen evolution (passive) corrosion model of Seitz and Walton (1993), Equation (4), page 8. The corrosion rate is assumed to be controlled by the diffusion of oxygen in the concrete and background (passive) corrosion. Oxygen diffusion and hydrogen evolution corrode the rebar used in concrete structural reinforcement. The model computes the remaining percentage of rebar after the onset of corrosion versus time—up to 100 values versus time can be computed.

Table 2. Inputs and outputs for CLATIM.

FORTRAN name	NUREG symbol	Type	Units	Description	Remarks
XC	Xc	Real	in	Thickness of rebar over concrete	INPUT, > 0.0
WCR	WCR	Real	None	Water to cement ratio	INPUT, > 0.0
CLCLION	CL	Real	ppm	Chloride ion concentration in ground water	INPUT, > 0.0
IERR	None	Integer	None	Error flag	OUTPUT, set to "1" if input error
TC	tc	Real	yr	Time to start of corrosion	OUTPUT

3.3.2 Usage. The model is executed by a FORTRAN subroutine call

```
CALL CORROD (DX, S, DIAM, CGWO2,
DI, N, TIME, PRCNT1, PRCNT2,
PRCNT3, CORHER, CORTIM, IERR,
HELP)
```

The inputs and outputs to this subroutine are defined in the sections below.

3.3.3 Inputs. The subroutine has 9 inputs: DX, S, DIAM, CGWO2, DI, N, TIME, CORHER, and CORTIM. These inputs are described in Table 3. The variable TIME is an array N long. DX is the depth of the rebar below the concrete surface, S is the rebar spacing, and DIAM is the diameter of the rebar. CGWO2 is the ground water oxygen concentration. DI is the intrinsic diffusion coefficient and can be calculated from the fractional porosity, θ, and the effective diffusion coefficient, DE, using

$$DI = \theta * DE, \quad 0 \leq \theta \leq 1.0.$$

CORHER is the hydrogen evolution (passive) corrosion rate and CORTIM is the time until onset of oxygen corrosion.

3.3.4 Outputs. The subroutine has 4 outputs: PRCNT1, PRCNT2, PRCNT3, and IERR. PRCNT1, PRCNT2, and PRCNT3 are arrays containing N values of recent rebar remaining that correspond to the N values of time in the TIME array. IERR is an error flag and is described in Table 3.

3.3.5 Table of Inputs and Outputs. Table 3 describes the inputs and outputs for CORROD.

3.3.6 Remarks. N must be input as a positive integer >0 and <100. N, DX, or DIAM all must be greater than zero. If any of these conditions are not met, IERR will be set to 1 and no calculations will be made. Zero is a valid value for the remaining variables. HELP is a LOGICAL * 4 flag intended for future use to control diagnostic printout, it is presently not used by this routine. ND denotes "nondimensional."

3.4 Subroutine User Instructions for FRZTH1

3.4.1 Description. This subroutine implements the freeze thaw degradation model of Walton et al. (1990), Equations (2), (3), (4), and (5), page 26. The model is based on laboratory tests and computes the annual rate of concrete loss, the reduction in dynamic modulus of elasticity based on the number of freeze thaw cycles, and the time to reach the level of reduction in dynamic modulus of elasticity.

3.4.2 Usage. The model is executed by a FORTRAN subroutine call:

```
CALL FRZTH1 (NCYCLS, AIR, WCR,
PHI, TR, Y1, TC, RFT, IERR, HELP)
```

The inputs and outputs to this subroutine are defined in the sections below.

3.4.3 Inputs. The subroutine has 5 inputs: NCYCLS, AIR, WCR, PHI, and TR. These inputs are described in Table 4.

3.4.4 Outputs. The subroutine has 4 outputs: Y1, TC, RFT, and IERR. IERR is an error flag and is described in Table 4.

3.4.5 Table of Inputs and Outputs. Table 4 describes the inputs and outputs for FRZTH1.

3.4.6 Remarks. NCYCLS must be input as a positive integer greater than zero. NCYCLS, AIR, WCR, or PHI must be real numbers greater than zero. If any of these conditions are not met, IERR will be set to 1 and no calculations will be made. HELP is a LOGICAL * 4 flag intended for future use to control diagnostic printout and is presently not used by this routine. ND denotes "nondimensional."

Table 3. Inputs and outputs for CORROD.

FORTRAN name	NUREG symbol	Type	Units	Description	Remarks
DX	Delta X	Real	in	Rebar depth below concrete surface	INPUT, converted to cm
S	s	Real	in	Rebar spacing	INPUT, converted to cm
DIAM	d	Real	in	Rebar diameter	INPUT, converted to cm
CGWO2	Cgw	Real	mols/cm^3	Concentration of oxygen in ground water	INPUT
DI	Di	Real	cm^2/sec	Intrinsic diffusion coefficient	INPUT
TIME	t	Real	Years	Array of time values	INPUT, N values
CORHER	a	Real	cm/yr	Corrosion rate due to hydrogen evolution (passive)	INPUT
CORTIM	t_c	Real	Years	Time until onset of oxygen corrosion (chloride attack time from CLATIM can be used)	INPUT
N	None	Integer	None	Number of time intervals desired	INPUT
IERR	None	Integer	None	Error flag	OUTPUT, set to "1" if input error
PRCNT1	%	Real	None	Percent concrete remaining based on oxygen corrosion as a function of time	OUTPUT, N values
PRCNT2	%	Real	None	Percent concrete remaining based on hydrogen evolution (passive) corrosion as a function of time	OUTPUT, N values
PRCNT3	%	Real	None	Percent concrete remaining based on both corrosion mechanisms at time, t	OUTPUT, N values

Table 4. Inputs and outputs for FRZTH1.

FORTRAN name	NUREG symbol	Type	Units	Description	Remarks
NCYCLS	N or NumCycles	Integer	None	Number of freeze thaw cycles	INPUT, > 50
AIR	AIR	Real	None	% entrained air	INPUT, 1 ≤ AIR ≤ 20
WCR	WCR	Real	None	Water to cement ratio	INPUT, > 0.0
PHI	ϕ, θ	Real	None	Fractional % porosity	INPUT, > 0.0, ≤ 1.00
TR	Tr	Real	None	Residual water content	INPUT at 0.9 to match BARRIER code
IERR	None	Integer	None	Error flag	OUTPUT, set to "1" if input error
Y1	Y1	Real	None	Decrease in dynamic elasticity modulus	OUTPUT
TC	Tc	Real	yr	Time to reach the value of Y1	OUTPUT
RFT	Rft	Real	cm/yr	Annual rate of concrete loss	OUTPUT

3.5 Subroutine User Instructions for LCHCNC

3.5.1 Description. This subroutine implements the concrete controlled leaching degradation model of Walton et al. (1990), Equation (4), page 19. The rate of leaching is assumed to be controlled by diffusion in the concrete, that is, the rate of leaching from a concrete surface is rapid compared to diffusion through the concrete. One-hundred values of deterioration depth x (depth of leaching) versus time can be computed.

3.5.2 Usage. The model is executed by a FORTRAN subroutine call

```
CALL LCHCNC (N, TIME, X, DI,
CLCAION, CGWCA, CS, IERR, HELP)
```

The inputs and outputs to this subroutine are defined in the sections below.

3.5.3 Inputs. The subroutine has 6 inputs: N, TIME, DI, CLCAION, CGWCA and CS. These inputs are described in Table 5. The variable TIME is an array N long. DI is the intrinsic diffusion coefficient and can be calculated from the fractional porosity, θ, and the effective diffusion coefficient, DE, using

$DI = \theta \bullet DE, \quad 0 \leq \theta \leq 1.0.$

3.5.4 Outputs. The subroutine has 2 outputs: X and IERR. X is an array containing N values of the depth of leaching that correspond to the N values of time in the TIME array. IERR is an error flag and is described in Table 5.

3.5.5 Table of Inputs and Outputs. Table 5 describes the inputs and outputs for LCHCNC.

3.5.6 Remarks. N must be input as a positive integer >0 and ≤100 the inputs N or CS must be >0. If any of these conditions are not met, IERR will be set to 1 and no calculations will be made. Zero is a valid value for the remaining variables. HELP is a LOGICAL * 4 flag intended for future use to control diagnostic printout, and is presently not used by this routine. ND denotes "nondimensional."

Table 5. Inputs and outputs for LCHCNC.

FORTRAN name	NUREG symbol	Type	Units	Description	Remarks
DI	Di	Real	cm^2/sec	Intrinsic diffusion coefficient	INPUT
CLCAION	Cl	Real	$mols/cm^3$	Concentration of Ca++ ion in concrete pore water	INPUT
CGWCA	Cgw	Real	$mols/cm^3$	Concentration of total inorganic Carbon in environment	INPUT
CS	Cs	Real	$mols/cm^3$	Bulk concentration of $Ca(OH)^2$ in solid concrete	INPUT
TIME	t	Real	Years	Array of time values	INPUT, N values
N	None	Integer	None	Number of time intervals desired	INPUT
IERR	None	Integer	None	Error flag	OUTPUT, set to "1" if input error
X	x	Real	cm	Depth of leached concrete at time, t, an array	OUTPUT, N values

3.6 Subroutine User Instructions for LCHGEO

3.6.1 Description. This subroutine implements the geology-controlled leaching degradation model of Walton et al. (1990), Equation (9), page 20. The rate of leaching is assumed to be controlled by diffusion into the local geologic environment (Walton et al. 1990). One-hundred values of deterioration depth x (depth of leaching) versus time can be computed.

3.6.2 Usage. The model is executed by a FORTRAN subroutine call:

CALL LCHGEO (N, TIME, X, DE, RD, PHI, CLCAION, CGWCA, CB, IERR, HELP)

The inputs and outputs to this subroutine are defined in the sections below.

3.6.3 Inputs. The subroutine has 8 inputs: N, TIME, DE, RD, PHI, CLCAION, CGWCA, and CB. These inputs are described in Table 6. The variable TIME is an array N long. DE is the effective diffusion coefficient, RD is a retardation factor that is dependent on the local environment ($0 \leq RD$), and θ is the fractional porosity ($0 \leq \theta \leq 1.0$).

3.6.4 Outputs. The subroutine has 2 outputs: X and IERR. X is an array containing N values of the depth of leaching that correspond to the N values of time in the TIME array. IERR is an error flag and is described in Table 6.

3.6.5 Table of Inputs and Outputs. Table 6 describes the inputs and outputs for LCHGEO.

3.6.6 Remarks. N must be input as a positive integer >0 and $\leq$100. The inputs N or CB must be >0. If any of these conditions are not met, IERR will be set to 1 and no calculations will be made. HELP is a LOGICAL * 4 flag intended for future use to control diagnostic printout, it is presently not used by this routine. ND denotes "nondimensional."

Table 6. Inputs and outputs for LCHGEO.

FORTRAN name	NUREG symbol	Type	Units	Description	Remarks
DE	De	Real	cm^2/sec	Effective diffusion coefficient	INPUT
RD	Rd	Real	None	Retardation factor	INPUT, >0
Phi	ϕ	Real	None	Fractional percent porosity	INPUT, value between 0.0 and 1.0
CLCAION	Cl	Real	$mols/cm^3$	Concentration of Ca++ ion in concrete pore water	INPUT
CGWCA	Cgw	Real	$mols/cm^3$	Concentration of total inorganic carbon in environment	INPUT
CB	Cb	Real	$mols/cm^3$	Concentration of Ca++ in bulk concrete, solid + pores	INPUT
TIME	t	Real	Years	Array of time values	INPUT, N values
N	None	Integer	None	Number of time intervals desired	INPUT
IERR	None	Integer	None	Error flag	OUTPUT, set to "1" if input error
X	x	Real	cm	Depth of leached concrete at time, t, an array	OUTPUT, N values

3.7 Subroutine User Instructions for SATTK1

3.7.1 Description. This subroutine implements the empirical sulfate attack model of Walton et al. (1990), Equation (5), page 10. The model assumes that the rate of sulfate attack is proportional to the total sulfate concentration in solution and the amount of tricalcium-aluminum silicate in the cement. One-hundred values of deterioration depth x versus time can be computed in this routine.

3.7.2 Usage. The model is executed by a FORTRAN subroutine call:

CALL SATTK1 (N, TIME, X, C3A, S_CAMG, IERR, HELP).

The inputs and outputs to this subroutine are defined in the sections below.

3.7.3 Inputs. The subroutine has 4 inputs: N, TIME, C3A, and S_CAMG. These inputs are described in Table 7. The variable TIME is an array N long.

3.7.4 Outputs. The subroutine has 2 outputs: X and IERR. X is an array containing N values of the depth of deterioration that correspond to the N

values of time in the TIME array. IERR is an error flag and is described in Table 7.

3.7.5 Table of Inputs and Outputs. Table 7 describes the inputs and outputs for SATTK1.

3.7.6 Remarks. S_CAMG is the sum of the concentrations of SO_4 and Mg ions: $[SO_4^{--}] + [Mg^{--}]$. The input N must be a positive integer >0 and <100. The errors possible with this routine are that one or more of the inputs are zero. If any of these conditions are not met, IERR will be set to 1 and no calculations will be made. HELP is a LOGICAL * 4 flag intended for future use to control diagnostic printout, it is presently not used by this routine. ND denotes "nondimensional."

3.8 Subroutine User Instructions for SATTK2

3.8.1 Description. This subroutine implements the sulfate attack degradation model of Walton et al. (1990), Equations (11–15), page 11. The model assumes sulfate ion diffusion into the concrete followed by reaction with aluminum phases and expansion, which subsequently causes stress cracking and foliation of the concrete (Walton et al. 1990). A single value of deterioration rate in cm/yr is computed at each call to the subroutine.

3.8.2 Usage. The model is executed by a FORTRAN subroutine call

```
CALL SATTK2(DI, M0, MC, C0, CK, TR,
M_CMT, V_CNC, R, IERR, HELP)
```

The inputs and outputs to this subroutine are defined in the sections below.

3.8.3 Inputs. The subroutine has 13 inputs: DI, E, B, M0, MC, C0, CK, TR, ALPHA, GAMMA, NU, M_CMT, and V_CNC. The inputs: E, B, ALPHA, GAMMA, and NU are *stored* (fixed) *internally* as recommended by Atkinson and Hearne (1990). These inputs are described in Table 8. The variable TIME is an array N long. DI is the intrinsic diffusion coefficient and can be calculated from the fractional porosity, θ, and the effective diffusion coefficient, DE, using

$DI = \theta \bullet DE, 0 \leq \theta \leq 1.0.$

3.8.4 Outputs. The subroutine has 2 outputs: R and IERR. R is the deterioration rate in cm/yr. IERR is an error flag and is described in Table 8.

3.8.5 Table of Inputs and Outputs. Table 8 describes the inputs and outputs for SATTK2.

Table 7. Inputs and outputs for SATTK1.

FORTRAN name	NUREG symbol	Type	Units	Description	Remarks
C3A	C3A	Real	Percent	Weight percent C3A in unhydrated cement	INPUT, value between 0.0 and 100.0
S_CAMG	([Mg++] + [SO4--])	Real	mol/L	Total sulfate concentration	INPUT
TIME	t	Real	Years	Array of time values	INPUT, N values
N	None	Integer	None	Number of time intervals desired	INPUT
IERR	None	Integer	None	Error flag	OUTPUT, set to "1" if input error
X	x	Real	cm	Depth of deterioration at time, t, an array	OUTPUT, N values

Table 8. Inputs and outputs for SATTK2.

FORTRAN name	NUREG symbol	Type	Units	Description	Remarks
DI	Di	Real	m^2/sec	Intrinsic diffusion coefficient	INPUT
E	E	Real	Pa	Young's modulus	INPUT, fixed internally at $20 * 10^9$
B	B	Real	m^3/mol	Linear strain/mole of sulfate in 1 m^3 of volume	INPUT, fixed internally at $1.8 * 10^{-6}$
M0	Mo	Real	mols/kg	Kinetic constant	INPUT
MC	Mc	Real	mols/kg	Value of concrete mass for complete reaction	INPUT
C0	Co	Real	$mols/m^3$	Conc. of sulfate in external/bulk solution	INPUT
CK	Ck	Real	$mols/m^3$	Conc. of sulfate in kinetic experiments	INPUT, $\neq 0$
TR	tr	Real	seconds	Characteristic time for reaction	INPUT, $\neq 0$
ALPHA	α	Real	None	Roughness factor for fracture path	INPUT, $\neq 0$, fixed internally at 1.0
GAMMA	γ	Real	$Joules/m^2$	Fracture surface energy	INPUT, $\neq 0$, fixed internally at 10.0
NU	ν	Real	None	Poisson's ratio	INPUT, < 1.0, fixed internally at 0.3
M_CMT	Mass cement	Real	kg	Mass cement	INPUT
V_CNC	Volume of concrete	Real	m^3	Volume of concrete	INPUT, $\neq 0$
IERR	None	Integer	None	Error flag	OUTPUT, "1" if input error
R	R	Real	cm/yr	Depth of leached concrete at time, t, an array	OUTPUT, converted to cm/yr

3.8.6 Remarks. NU, ALPHA, GAMMA, CK, TR, and V_CNC must be >0. If any of these conditions are not met, IERR will be set to 1 and no calculations will be made. The user should carefully check the physical validity of the input variables when using this model.

For verification purposes, call to SATTK2 with the following arguments:

CALL SATTK2 (1.0E-12, 0.32, 1.24, 5.208333, 12.2, 3577.0, 300.0, 1.0, R, IERR, HELP)

This will give a value of R of .0566 cm/yr to three decimal places, where the conversion factor of 31557600.00 seconds per year is used to convert seconds to years. HELP is a LOGICAL * 4 flag intended for future use to control diagnostic printout and it is presently not used by this routine. The subroutines ZBRAC and RTBIS of Press et al. (1989) are used by this routine to bracket and determine the root of the physical equations. ND denotes "nondimensional."

3.9 Subroutine User Instructions for FLOW

3.9.1 Description. This subroutine implements the equation to calculate the nondimensional flow rate from Walton and Seitz (1991), Equations (5–9), pages 5–6. The nondimensional flow rate (ζ_3) is assumed to be controlled by the crack spacing and width, and the depth of perched water. This representation (ζ_3) is the ratio of the flow rate through the cracks in the concrete to the permeability of the overlying porous media. Thus, ζ_3 can be multiplied by the permeability of the overlying porous media to obtain an effective flow rate through the combined porous media/cracked concrete system.

3.9.2 Usage. The model is executed by a FORTRAN subroutine call:

CALL FLOW (CRACK, SPACE, PERCH, HELP, GAPFRC, SMALZ0, ZETA3, IERR).

The inputs and outputs to this subroutine are defined in the sections below.

3.9.3 Inputs. The subroutine has 4 inputs: CRACK, SPACE, PERCH, HELP. These inputs are described in the table of inputs and outputs. An arbitrary set of consistent units can be used for the inputs (L = length). HELP is a debug flag with an input value of .TRUE. or .FALSE. depending on whether a debug file is requested. The user must use care in specifying PERCH. This is used as a head loss across the porous media above the crack. If there is more than one layer above the concrete, the user must scale PERCH appropriately to represent the head loss in a homogeneous layer with the permeability of the material immediately above the concrete. The inputs are specified in Table 9.

3.9.4 Outputs. The subroutine has 4 outputs: GAPFRC, SMALZ0, ZETA3, and IERR. GAPFRC is the value used in Figure 5-8 of Walton and Seitz (1991) (CRACK/SPACE). SMALZ0 is z_0 from Equations (5–9), ZETA3 is $\zeta 3$ from Equations (5–9), and IERR is an error flag. The inputs are discussed in Table 9.

3.9.5 Table of Inputs and Outputs. Table 9 describes the inputs and outputs for FLOW.

3.9.6 Remarks. CRACK must be input as a value greater than 10^{-15}, SPACE must be greater than or equal to CRACK, and PERCH must be greater than or equal to 1. If any of these conditions are not met, IERR will be set to 1 and no calculations will be made. Run times are generally very short (seconds), but for large ratios (>10^5) of the depth of perched water (PERCH) to the crack half-width (CRACK/2) the run times can be longer depending on the type of computer being used. In general, run times on a 386 computer should be shorter than a minute.

3.10 Subroutine User Instructions for RNNOD

3.10.1 Description. This subroutine implements the equations to estimate the ratio of the concentration in the water at a specified location along a crack to the initial total contaminant concentration in the concrete matrix. The radionuclide is assumed to start in the matrix and move into the crack by diffusion. Transport through the crack is assumed to occur by advection. The model is discussed in Walton (1992), pages 10–15.

Table 9. Inputs and outputs for FLOW (L = length, M = mass, T = time).

FORTRAN name	NUREG symbol	Type	Units	Description	Remarks
CRACK	2Ψ	Real	L	Average crack width ($>10^{-15}$)	INPUT
SPACE	$2X_0$	Real	L	Average crack spacing (>CRACK)	INPUT
PERCH	Z_0	Real	L	Perched water depth ($\geq$1.)	INPUT
HELP	None	Logical	None	Debug flag	INPUT
GAPFRC	None	Real	None	Crack width/spacing, used in Figure 5-8 of Walton and Seitz (1991)	OUTPUT
SMALZ0	z_0	Real	None	Perched water depth/crack half-width	OUTPUT
ZETA3	$\zeta 3$	Real	None	Nondimensional flow rate	OUTPUT
IERR	None	Integer	None	Error flag, set to "1" if input error.	OUTPUT

3.10.2 Usage. The model is executed by a FORTRAN subroutine call:

CALL RNNOD (TIME, DARV, CRKFRC, DIST, SPACE, ALPHA, POR, DA, DECAY, EPS, HELP, CONC, IERR).

The subroutine calls 3 subroutines from Press et al. (1989) to complete a Romberg integration for the integral in Equation (26) of Walton (1992). The three subroutines from Numerical Recipes are: QROMB, TRAPZD, and POLINT. QROMB is a Romberg integration routine, TRAPZD is a routine to conduct trapezoidal rule calculations in support of the integration, and POLINT is a polynomial interpolation routine. Inputs and outputs are listed in the following sections.

3.10.3 Inputs. The subroutine has 11 inputs: TIME, DARV, CRKFRC, DIST, SPACE, ALPHA, POR, DA, DECAY, EPS, and HELP. An arbitrary set of consistent units can be used (L = length, M = mass, T = time). These inputs are described in Table 10. EPS is a parameter used in a Romberg integration routine to solve the integral in Equation (26) of Walton (1992). EPS should initially be set at 1–7. EPS is discussed further in Section 3.10.6. HELP is a debug flag with an input value of .TRUE. or .FALSE. depending on whether a debug file is requested.

3.10.4 Outputs. The subroutine has 2 outputs: CONC and IERR. CONC is the ratio of the contaminant concentration in the fracture to the initial total contaminant concentration in the matrix. To obtain a pore contaminant concentration in the fracture, CONC should be multiplied by the total initial contaminant concentration in the matrix (solids and pores/ALPHA). IERR is an error flag. Outputs are specified in Table 10.

3.10.5 Table of Inputs and Outputs. Table 10 describes the inputs and outputs for RNNOD.

Table 10. Inputs and outputs for RNNOD (L = length, M = mass, T = time).

FORTRAN name	NUREG symbol	Type	Units	Description	Remarks
DARV	V	Real	L/T	Average Darcy velocity through concrete ($>10^{-15}$)	INPUT
CRKFRC	None	Real	None	Used to calculate U_f in Walton et al. (1990) ($>10^{-15}$)	INPUT
TIME	t	Real	T	Time at which calculation is made ($>10^{-15}$)	INPUT
DIST	z	Real	L	Distance along crack for calculation ($>10^{-15}$)	INPUT
SPACE	S	Real	L	Spacing of cracks (> CRKFRC* SPACE)	INPUT
ALPHA	α	Real	None	Partitioning factor (αRd) ($>10^{-15}$)	INPUT
POR	ϕ_m	Real	None	Matrix porosity ($>10^{-15}$)	INPUT
DA	D_a	Real	L^2/T	Apparent diffusion coefficient ($>10^{-15}$)	INPUT
DECAY	λd	Real	L/T	Radioactive decay constant ($>10^{-15}$)	INPUT
EPS	None	Real	None	Convergence criteria (start with 10^{-7}) ($>10^{-15}$)	INPUT
HELP	None	Logical	None	Debug flag	INPUT
CONC	$\alpha C_f/C_{f0}$	Real	None	Nondimensional concentration	OUTPUT
IERR	None	Integer	None	Error flag, set to "1" if input error	OUTPUT

3.10.6 Remarks. TIME, CRKFRC, ALPHA, DECAY, DIST, DARV, POR, DA, and EPS must be input as a value greater than 10^{-15}, and SPACE must be greater than to CRKFRC*SPACE. If any of these criteria are not met, IERR will be set to 1 without any calculations being conducted. IERR will be set to 2 if too many iterations are required to complete the Romberg integration. In this case, the EPS must be increased to 10^{-6} and the user must check the results carefully to ensure that reasonable results are still obtained. This can be done by running simulations at times greater than and less than the requested time to check for oscillations in the predicted contaminant concentration.

Cases that require a large number of iterations also tend to take a long time to run. The run time is a function of the rate at which contaminant concentrations are changing in the system and the length of the simulation time period. For example, a 10,000-year simulation for a nonsorbing radionuclide and a relatively high flow velocity through the crack (DARV/CRKFRC > 1000) can require a large amount of computer time. For larger velocities in the crack, simulations can take

hours on a 386 computer. However, for simulations at lower velocities, shorter times, or including sorption the routine will run relatively quickly (seconds to minutes).

3.11 Subroutine User Instructions for SUDFRD

3.11.1 Description. This subroutine implements the equation to estimate the thickness of concrete required to attenuate the contaminant concentration in the leachate by a given fraction assuming a specified crack width and spacing. The model is discussed in Walton (1992), pages 16–19.

3.11.2 Usage. The model is executed by a FORTRAN subroutine call:

CALL SUDFRD (DARV, DECAY, DELTC, RD, DE, CRACK, SPACE, POR, HELP, THICK, IERR).

The inputs and outputs to this subroutine are defined in the sections below.

3.11.3 Inputs. The subroutine has 9 inputs: DARV, DECAY, DELTC, RD, DE, CRACK, SPACE, POR, HELP. An arbitrary set of consistent units can be used for the inputs (L=length, M=mass, T=time). These inputs are described in Table 11. HELP is a debug flag with an input

Table 11. Inputs and outputs for SUDFRD (L = length, M = mass, T = time).

FORTRAN name	NUREG symbol	Type	Units	Description	Remarks
DARV	None	Real	L/T	Average Darcy velocity through concrete used to calculate U_f ($>10^{-15}$)	INPUT
DECAY	λ	Real	L/T	Radioactive decay constant ($>10^{-15}$)	INPUT
DELTC	(c/c_0)	Real	None	Desired reduction in contaminant concentration	INPUT
RD	R_d	Real	None	Retardation coefficient ($\geq 1.$)	INPUT
DE	D_e	Real	L^2/T	Effective diffusion coefficient ($>10^{-15}$)	INPUT
CRACK	2b	Real	L	Average crack width ($>10^{-15}$)	INPUT
SPACE	2B	Real	L	Average crack spacing (>CRACK)	INPUT
POR	ϕ	Real	L	Concrete porosity	INPUT
HELP	None	Logical	None	Debug switch	INPUT
THICK	$d_{(c/c0)}$	Real	L	Thickness of concrete required to reduce by DELTC	OUTPUT
IERR	None	Integer	None	Error flag, set to "1" if input error.	OUTPUT

value of .TRUE. or .FALSE. depending on whether a debug file is requested.

3.11.4 Outputs. The subroutine has 2 outputs: THICK and IERR. THICK is the thickness of concrete required to reduce the contaminant concentration at the base of the cracked concrete by a specified fraction. IERR is an error flag.

3.11.5 Table of Inputs and Outputs. Table 11 describes the inputs and outputs for SUDFRD.

3.11.6 Remarks. CRACK, DARV, DECAY, and DE must be input as a value greater than 10^{-15}, SPACE must be >CRACK, and RD must be >1. If any of these criteria are not met, IERR will be set to 1 without any calculations being conducted.

4. REFERENCES

Atkinson, A. and J. A. Hearne, 1990, "Mechanistic Model for the Durability of Concrete Barriers Exposed to Sulfate-Bearing Groundwaters," *Proceedings of the Materials Research Society, 176.*

Press, W. H., B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, 1989, *Numerical Recipes (FORTRAN Version)*, Cambridge University Press.

Seitz, R. R. and J. C. Walton, 1993, Modeling Approaches for Concrete Barriers Used in Low-Level Waste Disposal, NUREG/CR-6070, EGG–2701, September.

Walton, J. C., 1992, *Performance of Intact and Partially Degraded Concrete Barriers in Limiting Mass Transport*, NUREG/CR-5445, EGG-2662, June.

Walton, J. C. and R. R. Seitz, 1991, *Performance of Intact and Partially Degraded Concrete Barriers in Limiting Fluid Flow*, NUREG/CR-5614, EGG-2614, June.

Walton J. C., L. E. Plansky, and R. W. Smith, 1990, *Computer Models for the Estimation of Service Life of Concrete Barriers*, NUREG/CR-5542, EGG-2597, August.

Appendix A

User Instructions for Driver Routines

Appendix A

User Instructions for Driver Routines

INTRODUCTION

The present version of CEMENT is version 1.01 as described in this user's guide. This appendix provides user instructions for the interactive driver modules to execute the subroutines described previously in this report. In most cases, the inputs and outputs from the driver modules are the same as those from the individual subroutines. Thus, input and output listings are not repeated for those routines. The user should refer to the appropriate section in the main body of this user's manual. However, in some cases, calculations are conducted in the driver routines and the inputs and outputs are slightly different from the subroutines. In these cases, descriptions of the input and output variables that are different are discussed in the text below.

Execution of each of the driver modules is addressed independently in the following sections. The first section addresses the main driver followed by discussions of the drivers for the Flow and Transport, and Degradation subroutines, respectively. The interactive screens for the main driver are illustrated in this section. Examples of the interactive input screens for each of the individual routines are provided in Appendix B to allow the user to see how the program will look on the screen when executed and also to provide some test problems for initial installation. The user is given the option to input values for the subroutine variables interactively or using a file. Examples using interactive and file inputs are provided for the first subroutine. Interactive input is assumed for each of the following subroutines. Example input files corresponding to the interactive test cases are provided for each routine.

USER INSTRUCTIONS FOR MAIN DRIVER

Description

The purpose of the main driver, CEMENT, is to allow the user to interactively select the subroutine to be run and then to execute the driver for that subroutine. After completion of the run for a given subroutine, the user is prompted to determine if further runs are desired with the same subroutine or different subroutines. All inputs are conducted interactively in accordance with instructions provided on the screen.

Usage

In order to execute the main driver simply type, CEMENT (or another name may be given to the executable, if the user desires). The first screen is the title screen that includes a brief title and the disclaimer as follows [note: if using a compiler other than LAHEY, the statement "Press ENTER to Continue" may not appear (See Appendix C)]:

```
*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

                        - CEMENT -  Version 1.01
 A MODULAR COLLECTION OF SIMPLIFIED CODES FOR USE IN ESTIMATING LONG-TERM PERFORMANCE OF CEMENT-BASED BARRIERS. THE CODES ARE BASED ON MODELS DESCRIBED IN NUREG/CR-5445, NUREG/CR-5542, NUREG/CR-5614, and NUREG/CR-6070.

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

NOTICE: This program was developed for the U.S. Nuclear Regulatory Commission. Neither the United States Government nor any agency thereof, or any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for any third party's use, or the results of such use, of any portion of this program or represents that its use by such third party would not infringe privately owned rights.

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
(Press ENTER to continue)
```

After pressing <ENTER>, the MAIN MENU will scroll onto the screen as follows:

```
**************************************************

                        MAIN MENU

**************************************************

1) Flow and Transport
2) Concrete Degradation
3) Quit

**************************************************
Enter the number for the type of model:
```

The user is then prompted to enter the number for the type of model to be run. If the user enters "1", the FLOW AND TRANSPORT MODELS MENU will scroll onto the screen as follows:

```
**************************************************

                FLOW AND TRANSPORT MODELS MENU

**************************************************

1) Release from Cracked Waste Form
2) Release through Cracked Vault
3) Flow through Cracked Roof
4) Return to Main Menu

**************************************************
Enter the number for the desired model:
```

If the user enters "2" at the MAIN MENU, the CONCRETE DEGRADATION MODELS MENU will scroll onto the screen as shown:

```
**************************************************

                CONCRETE DEGRADATION MODELS MENU

**************************************************

1) Carbonation
2) Chloride Attack (Time until Rebar Corrosion)
3) Rebar Corrosion
4) Freeze/Thaw
5) Concrete Controlled Leaching
6) Geology Controlled Leaching
7) Empirical Model for Sulfate Attack
8) Mechanistic Model for Sulfate Attack
9) Return to Main Menu

**************************************************
Enter the number for the desired model:
```

In both the FLOW AND TRANSPORT and CONCRETE DEGRADATION menus, the user is prompted to enter the number for the desired model. When a given model is selected, the subroutine for the driver for that model is called. The appropriate screen for the selected subroutine driver will scroll onto the screen as shown in Appendix B. After the subroutine is run, control is returned to the main driver and the user is prompted regarding additional runs as shown:

```
DO YOU WANT TO MAKE ADDITIONAL RUNS ?
  (1 = yes, with different subroutine)
  (2 = yes, with same subroutine)
  (0 = no)
```

If the user enters "1", the MAIN MENU screen will appear. If the user enters "2", the initial screen for the same subroutine driver run previously will appear. If "0" is entered, the program will stop.

Inputs/Outputs

All inputs are interactive. No variables are passed when each of the individual subroutine drivers are called. There are no input or output files.

Remarks

If the user inputs a value that is out of the range of values shown in the menu, the user is prompted to re-enter an appropriate value.

USER INSTRUCTIONS FOR CRBCAL—DRIVER FOR CARBONATION DEGRADATION MODEL

Description

CRBCAL is the driver routine for CARBNA, the carbonation model described in Section 3.1. CRBCAL provides an interface to allow the user to interactively execute the CARBNA routine.

Usage

CRBCAL can either be executed using the main driver, CEMENT, or can be compiled and executed separately due to the modular nature of the routines. Execution of CRBCAL using CEMENT is discussed in the instructions for use of CEMENT provided at the beginning of Appendix A. The FORTRAN subroutine call is:

CALL CRBCAL.

In order to compile and execute CRBCAL separately from the main driver the user must place a comment symbol in column 1 in front of the SUBROUTINE statement at the beginning of the routine and the user must change RETURN to STOP at the end of the subroutine. CRBCAL can then be compiled along with the CARBNA routine and executed as a separate interactive module. An example of the use of CRBCAL, which applies to use individually or when called by the CEMENT driver is provided in Appendix B.

Inputs/Outputs

All inputs and outputs for CRBCAL are the same as those for the CARBNA routine described in Section 3.1, except for the variable INPUT. INPUT is an integer to identify if the input will be interactive, using a file, or whether the user wants to leave the routine before conducting calculations.

Two files are opened by the CRBCAL subroutine for input and output. Unit 3 is opened with a file name of CARBNA.IN for input data and Unit 4 is opened with the file name CARBNA.OUT for the output file.

Remarks

If an inappropriate value is used for one of the input parameters, an error message will be printed and no results will be provided. The error conditions for CRBCAL are the same as those for CARBNA discussed in Section 3.1. In the case of an inappropriate input the following error message will appear on the screen and in the output file:

```
**********************
*** INPUT ERROR ***
**********************
```

USER INSTRUCTIONS FOR CLACAL—DRIVER FOR CHLORIDE ATTACK DEGRADATION MODEL

Description

CLACAL is the driver routine for CLATIM, the chloride attack model described in Section 3.2. CLACAL provides an interface to allow the user to interactively execute the CLATIM routine.

Usage

CLACAL can either be executed using the main driver, CEMENT, or can be compiled and executed separately due to the modular nature of the routines. Execution of CLACAL using CEMENT is discussed in the instructions for use of CEMENT provided at the beginning of Appendix A. The FORTRAN subroutine call is:

CALL CLACAL.

In order to compile and execute CLACAL separately from the main driver, the user must place a comment symbol in column 1 in front of the SUBROUTINE statement at the beginning of the routine and the user must change RETURN to STOP at the end of the subroutine. CLACAL can then be compiled along with the CLATIM routine and executed as a separate interactive module. An example of the use of CLACAL, which applies to use individually or when called by the CEMENT driver is provided in Appendix B.

Inputs/Outputs

All inputs and outputs for CLACAL are the same as those for the CLATIM routine described in Section 3.2, except for the variable INPUT. INPUT is an integer to identify if the input will be interactive, using a file, or whether the user wants to leave the routine before conducting calculations.

Two files are opened by the CLACAL subroutine for input and output. Unit 3 is opened with a file name of CLATIM.IN for input data and Unit 4 is opened with the file name CLATIM.OUT for the output file.

Remarks

If an inappropriate value is used for one of the input parameters, an error message will be printed and no results will be provided. The error conditions for CLACAL are the same as those for CLATIM discussed in Section 3.2. In the case of an inappropriate input, the following error message will appear on the screen and in the output file:

```
**********************
*** INPUT ERROR ***
**********************
```

USER INSTRUCTIONS FOR CORCAL- DRIVER FOR REBAR CORROSION DEGRADATIO MODEL

Description

CORCAL is the driver routine for CORROD, the oxygen diffusion and hydrogen evolution (passive) rebar corrosion model described in Section 3.3. CORCAL provides an interface to allow the user to interactively execute the CORROD routine.

Usage

CORCAL can either be executed using the main driver, CEMENT, or can be compiled and executed separately due to the modular nature of the routines. Execution of CORCAL using CEMENT is discussed in the instructions for use of CEMENT provided at the beginning of Appendix A. The FORTRAN subroutine call is:

```
CALL CORCAL.
```

In order to compile and execute CORCAL separately from the main driver the user must place a comment symbol in column 1 in front of the SUBROUTINE statement at the beginning of the routine and the user must change RETURN to STOP at the end of the subroutine. CORCAL can then be compiled along with the CORROD routine and executed as a separate interactive module. An example of the use of CORCAL, which applies to use individually or when called by the CEMENT driver is provided in Appendix B.

Inputs/Outputs

All inputs and outputs for CORCAL are the same as those for the CORROD routine described in Section 3.3, except for the variable INPUT. INPUT is an integer to identify if the input will be interactive, using a file, or whether the user wants to leave the routine before conducting calculations.

Two files are opened by the CORCAL subroutine for input and output. Unit 3 is opened with a file name of CORROD.IN for input data and Unit 4 is opened with the file name CORROD.OUT for the output file.

Remarks

If an inappropriate value is used for one of the input parameters, an error message will be printed and no results will be provided. The error conditions for CORCAL are the same as those for CORROD discussed in Section 3.3. In the case of an inappropriate input the following error message will appear on the screen and in the output file:

```
**********************
*** INPUT ERROR ***
**********************
```

USER INSTRUCTIONS FOR FR1CAL—DRIVER FOR FREEZE/THAW DEGRADATION MODEL

Description

FR1CAL is the driver routine for FRZTH1, the freeze/thaw model described in Section 3.4. FR1CAL provides an interface to allow the user to interactively execute the FRZTH1 routine.

Usage

FR1CAL can either be executed using the main driver, CEMENT, or can be compiled and executed separately due to the modular nature of the routines. Execution of FR1CAL using CEMENT is discussed in the instructions for use of CEMENT provided at the beginning of Appendix A. The FORTRAN subroutine call is:

CALL FR1CAL.

In order to compile and execute FR1CAL separately from the main driver the user must place a comment symbol in column 1 in front of the SUBROUTINE statement at the beginning of the routine and the user must change RETURN to STOP at the end of the subroutine. FR1CAL can then be compiled along with the FRZTH1 routine and executed as a separate interactive module. An example of the use of FR1CAL, which applies to use individually or when called by the CEMENT driver is provided in Appendix B.

Inputs/Outputs

All inputs and outputs for FR1CAL are the same as those for the FRZTH1 routine described in Section 3.4, except for the variable INPUT. INPUT is an integer to identify if the input will be interactive, using a file, or whether the user wants to leave the routine before conducting calculations.

Two files are opened by the FR1CAL subroutine for input and output. Unit 3 is opened with a file name of FRZTH1.IN for input data and Unit 4 is opened with the file name FRZTH1.OUT for the output file.

Remarks

If an inappropriate value is used for one of the input parameters, an error message will be printed and no results will be provided. The error conditions for FR1CAL are the same as those for FRZTH1 discussed in Section 3.4. In the case of an inappropriate input the following error message will appear on the screen and in the output file:

```
**********************
*** INPUT ERROR ! **
**********************
```

USER INSTRUCTIONS FOR CLCCAL—DRIVER FOR CONCRETE CONTROLLED LEACHING DEGRADATION MODEL

Description

CLCCAL is the driver routine for LCHCNC, the concrete controlled leaching model described in Section 3.5. CLCCAL provides an interface to allow the user to interactively execute the LCHCNC routine.

Usage

CLCCAL can either be executed using the main driver, CEMENT, or can be compiled and executed separately due to the modular nature of the routines. Execution of CLCCAL using CEMENT is discussed in the instructions for use of CEMENT provided at the beginning of Appendix A. The FORTRAN subroutine call is:

```
CALL CLCCAL.
```

In order to compile and execute CLCCAL separately from the main driver the user must place a comment symbol in column 1 in front of the SUBROUTINE statement at the beginning of the routine and the user must change RETURN to STOP at the end of the subroutine. CLCCAL can then be compiled along with the LCHCNC routine and executed as a separate interactive module. An example of the use of CLCCAL, which applies to use individually or when called by the CEMENT driver is provided in Appendix B.

Inputs/Outputs

All inputs and outputs for CLCCAL are the same as those for the LCHCNC routine described in Section 3.5, except for the variable INPUT. INPUT is an integer to identify if the input will be interactive, using a file, or whether the user wants to leave the routine before conducting calculations.

Two files are opened by the CLCCAL subroutine for input and output. Unit 3 is opened with a file name of LCHCNC.IN for input data and Unit 4 is opened with the file name LCHCNC.OUT for the output file.

Remarks

If an inappropriate value is used for one of the input parameters, an error message will be printed and no results will be provided. The error conditions for CLCCAL are the same as those for LCHCNC discussed in Section 3.5. In the case of an inappropriate input the following error message will appear on the screen and in the output file:

```
**********************
*** INPUT ERROR ***
**********************
```

USER INSTRUCTIONS FOR GLCCAL—DRIVER FOR GEOLOGY CONTROLLED LEACHING DEGRADATION MODEL

Description

GLCCAL is the driver routine for LCHGEO, the geology controlled leaching model described in Section 3.6. GLCCAL provides an interface to allow the user to interactively execute the LCHGEO routine.

Usage

GLCCAL can either be executed using the main driver, CEMENT, or can be compiled and executed separately due to the modular nature of the routines. Execution of GLCCAL using CEMENT is discussed in the instructions for use of CEMENT provided at the beginning of Appendix A. The FORTRAN subroutine call is:

CALL GLCCAL.

In order to compile and execute GLCCAL separately from the main driver the user must place a comment symbol in column 1 in front of the SUBROUTINE statement at the beginning of the routine and the user must change RETURN to STOP at the end of the subroutine. GLCCAL can then be compiled along with the LCHGEO routine and executed as a separate interactive module. An example of the use of GLCCAL, which applies to use individually or when called by the CEMENT driver is provided in Appendix B.

Inputs/Outputs.

All inputs and outputs for GLCCAL are the same as those for the LCHGEO routine described in Section 3.6, except for the variable INPUT. INPUT is an integer to identify if the input will be interactive, using a file, or whether the user wants to leave the routine before conducting calculations.

Two files are opened by the GLCCAL subroutine for input and output. Unit 3 is opened with a file name of LCHGEO.IN for input data and Unit 4 is opened with the file name LCHGEO.OUT for the output file.

Remarks

If an inappropriate value is used for one of the input parameters, an error message will be printed and no results will be provided. The error conditions for GLCCAL are the same as those for LCHGEO discussed in Section 3.6. In the case of an inappropriate input the following error message will appear on the screen and in the output file:

```
**********************
*** INPUT ERROR ***
**********************
```

USER INSTRUCTIONS FOR SA1CAL—DRIVER FOR EMPIRICAL SULFATE ATTACK DEGRADATION MODEL

Description

SA1CAL is the driver routine for SATTK1, the empirical sulfate attack model described in Section 3.7. SA1CAL provides an interface to allow the user to interactively execute the SATTK1 routine.

Usage

SA1CAL can either be executed using the main driver, CEMENT, or can be compiled and executed separately because of the modular nature of the routines. Execution of SA1CAL using CEMENT is discussed in the instructions for use of CEMENT provided at the beginning of Appendix A. The FORTRAN subroutine call is:

CALL SA1CAL.

In order to compile and execute SA1CAL separately from the main driver the user must place a comment symbol in column 1 in front of the SUBROUTINE statement at the beginning of the routine and the user must change RETURN to STOP at the end of the subroutine. SA1CAL can then be compiled along with the SATTK1 routine and executed as a separate interactive module. An example of the use of SA1CAL, which applies to use individually or when called by the CEMENT driver is provided in Appendix B.

Inputs/Outputs

All inputs and outputs for SA1CAL are the same as those for the SATTK1 routine described in Section 3.7, except for the variable INPUT. INPUT is an integer to identify if the input will be interactive, using a file, or whether the user wants to leave the routine before conducting calculations.

Two files are opened by the SA1CAL subroutine for input and output. Unit 3 is opened with a file name of SATTK1.IN for input data and Unit 4 is opened with the file name SATTK1.OUT for the output file.

Remarks

If an inappropriate value is used for one of the input parameters, an error message will be printed and no results will be provided. The error conditions for SA1CAL are the same as those for SATTK1 discussed in Section 3.7. In the case of an inappropriate input the following error message will appear on the screen and in the output file:

```
**********************
*** INPUT ERROR ***
**********************
```

USER INSTRUCTIONS FOR SA2CAL—DRIVER FOR MECHANISTIC SULFATE ATTACK DEGRADATION MODEL

Description

SA2CAL is the driver routine for SATTK2, the mechanistic sulfate attack model described in Section 3.8. SA2CAL provides an interface to allow the user to interactively execute the SATTK2 routine.

Usage

SA2CAL can either be executed using the main driver, CEMENT, or can be compiled and executed separately due to the modular nature of the routines. Execution of SA2CAL using CEMENT is discussed in the instructions for use of CEMENT provided at the beginning of Appendix A. The FORTRAN subroutine call is:

CALL SA2CAL.

In order to compile and execute SA2CAL separately from the main driver the user must place a comment symbol in column 1 in front of the SUBROUTINE statement at the beginning of the routine and the user must change RETURN to STOP at the end of the subroutine. SA2CAL can then be compiled along with the SATTK2 routine and executed as a separate interactive module. An example of the use of SA2CAL, which applies to use individually or when called by the CEMENT driver is provided in Appendix B.

Inputs/Outputs

All inputs and outputs for SA2CAL are the same as those for the SATTK2 routine described in Section 3.8, except for the variable INPUT. INPUT is an integer to identify if the input will be interactive, using a file, or whether the user wants to leave the routine before conducting calculations.

Two files are opened by the SA2CAL subroutine for input and output. Unit 3 is opened with a file name of SATTK2.IN for input data and Unit 4 is opened with the file name SATTK2.OUT for the output file.

Remarks

If an inappropriate value is used for one of the input parameters, an error message will be printed and no results will be provided. The error conditions for SA2CAL are the same as those for SATTK2 discussed in Section 3.8. In the case of an inappropriate input the following error message will appear on the screen and in the output file:

```
**********************
*** INPUT ERROR ***
**********************
```

USER INSTRUCTIONS FOR FLWCAL—DRIVER FOR FLOW THROUGH CRACKED ROOF MODEL

Description

FLWCAL is the driver routine for FLOW, the model for flow through a cracked vault roof described in Section 3.9. FLWCAL provides an interface to allow the user to interactively execute the FLOW routine.

Usage

FLWCAL can either be executed using the main driver, CEMENT, or can be compiled and executed separately because of the modular nature of the routines. Execution of FLWCAL using CEMENT is discussed in the instructions for use of CEMENT provided at the beginning of Appendix A. The FORTRAN subroutine call is:

CALL FLWCAL.

In order to compile and execute FLWCAL separately from the main driver the user must place a comment symbol in column 1 in front of the SUBROUTINE statement at the beginning of the routine and the user must change RETURN to STOP at the end of the subroutine. FLWCAL can then be compiled along with the FLOW routine and executed as a separate interactive module. An example of the use of FLWCAL, which applies to use individually or when called by the CEMENT driver is provided in Appendix B.

Inputs/Outputs

All inputs and outputs for FLWCAL are the same as those for the FLOW routine described in Section 3.9, except for the variables INPUT, PERM, and FLW. INPUT is an integer to identify if the input will be interactive, using a file, or whether the user wants to leave the routine before conducting calculations. PERM and FLW are discussed in Table A-1.

Two files are opened by the FLWCAL subroutine for input and output. Unit 3 is opened with a file name of FLOW.IN for input data and Unit 4 is opened with the file name FLOW.OUT for the output file.

Remarks

If an inappropriate value is used for one of the input parameters, an error message will be printed and no results will be provided. The error conditions for FLWCAL are the same as those for FLOW discussed in Section 3.9 with additional conditions as noted in Table A-1. In the case of an inappropriate input the following error message will appear on the screen and in the output file:

```
**********************
*** INPUT ERROR ***
**********************
```

Table A-1. Inputs and outputs for FLWCAL different from FLOW (L = length, M = mass, T = time).

FORTRAN name	NUREG symbol	Type	Units	Description	Remarks
PERM	N/A	Real	L/T	Hydraulic conductivity of homogeneous porous material above concrete used to calculate FLW = ZETA3 * PERM (>10^{-15})	INPUT
FLW	N/A	Real	L/T	Effective flow rate through combined porous media/concrete	OUTPUT

USER INSTRUCTIONS FOR RNCALL—DRIVER FOR RELEASE FROM CRACKED WASTE FORM MODEL

Description

RNCALL is the driver routine for RNNOD, the model for flow through a cracked vault roof described in Section 3.10. RNCALL provides an interface to allow the user to interactively execute the RNNOD routine.

Usage

RNCALL can either be executed using the main driver, CEMENT, or can be compiled and executed separately because of the modular nature of the routines. Execution of RNCALL using CEMENT is discussed in the instructions for use of CEMENT provided at the beginning of Appendix A. The FORTRAN subroutine call is:

CALL RNCALL.

In order to compile and execute RNCALL separately from the main driver the user must place a comment symbol in column 1 in front of the SUBROUTINE statement at the beginning of the routine and the user must change RETURN to STOP at the end of the subroutine. RNCALL can then be compiled along with the RNNOD routine and executed as a separate interactive module. An example of the use of RNCALL, which applies to use individually or when called by the CEMENT driver is provided in Appendix B.

Inputs/Outputs

All inputs and outputs for RNCALL are the same as those for the RNNOD routine described in Section 3.10, except for the variables INPUT, TC, CRACK, RKD, RHO, DE, THALF, CF, and RELS. INPUT is an integer to identify if the input will be interactive, using a file, or whether the user wants to leave the routine before conducting calculations. TC, CRACK, RKD, RHO, DE, THALF, CF, and RELS are discussed in Table A-2.

Two files are opened by the RNCALL subroutine for input and output. Unit 3 is opened with a file name of RNNOD.IN for input data and Unit 4 is opened with the file name RNNOD.OUT for the output file.

Remarks

If an inappropriate value is used for one of the input parameters, an error message will be printed and no results will be provided. The error conditions for RNCALL are the same as those for RNNOD discussed in Section 3.10 with additional conditions as noted in Table A-2. In the case of an inappropriate input the following error message will appear on the screen and in the output file:

```
**********
*** INPUT ERROR ***
**********
```

A warning will also be issued if too many iterations are required to obtain convergence in the numerical iteration. The user is requested to relax the EPS by increasing from 10^{-7} to 10^{-6}. The user must compare the results from the two simulations to note any change in the predicted concentration. Furthermore, when using any convergence criterion above 10^{-7} the user should check results for times on each side of the time being simulated to look for oscillations in the predicted concentration. The warning message that is printed is:

```
********** WARNING **********
Too many steps required to reach convergence. It is recommended that the user relax the convergence criteria (EPS) and rerun the program. An increase by a factor of 10 is recommended. If convergence criteria are relaxed, the user needs to use caution to ensure that the results are reasonable. The results for the case before and after relaxing EPS should be compared.
**********
```

Table A-2. Inputs and outputs for RNCALL different from RNNOD (L = length, M = mass, T = time).

FORTRAN name	NUREG symbol	Type	Units	Description	Remarks
TC	C_{t0}	Real	M/L^3	Total radionuclide concentration in unit volume of porous media (solid and liquid) used to calculate CF = CONC*TC/ALPHA	INPUT
CRACK	N/A	Real	L	Crack width used to calculate the RNNOD input CRKFRC prior to calling RNNOD. Where, CRKFRC = CRACK/SPACE ($>10^{-15}$)	INPUT
RKD	K_d	Real	L^3/M	Radionuclide distribution coefficient used to calculate the RNNOD input ALPHA prior to calling RNNOD. Where, ALPHA = POR * RD = POR * (1+RHO*RKD/POR) ($>10^{-15}$)	INPUT
RHO	ρ_b	Real	M/L^3	Dry bulk density of concrete used to calculate the RNNOD input ALPHA prior to calling RNNOD. Where, as noted above ($>10^{-15}$)	INPUT
DE	D_e	Real	L^2/T	Effective diffusion coefficient through concrete used to calculate the RNNOD input DA prior to calling RNNOD. Where, DA = DE/RD ($>10^{-15}$)	INPUT
THALF	$T_{1/2}$	Real	T	Radionuclide half-life used to calculate the RNNOD input DECAY prior to calling RNNOD. Where, DECAY = LN[2]/THALF ($>10^{-10}$)	INPUT
CF	C_f	Real	M/L^3	Radionuclide concentration in crack pore fluid used to calculate RELS = CF * DARV	OUTPUT
RELS	R	Real	M/L^2-T	Effective release rate from cracked concrete waste form	OUTPUT

USER INSTRUCTIONS FOR SFCALL—DRIVER FOR RELEASE THROUGH CRACKED VAULT MODEL

Description

SFCALL is the driver routine for SUDFRD, the model for releases through a cracked vault described in Section 3.11. SFCALL provides an interface to allow the user to interactively execute the SUDFRD routine.

Usage

SFCALL can either be executed using the main driver, CEMENT, or can be compiled and executed separately because of the modular nature of the routines. Execution of SFCALL using CEMENT is discussed in the instructions for use of CEMENT provided at the beginning of Appendix A. The FORTRAN subroutine call is:

CALL SFCALL.

In order to compile and execute SFCALL separately from the main driver the user must place a comment symbol in column 1 in front of the SUBROUTINE statement at the beginning of the routine and the user must change RETURN to STOP at the end of the subroutine. SFCALL can then be compiled along with the SUDFRD routine and executed as a separate interactive module. An example of the use of SFCALL, which applies to use individually or when called by the CEMENT driver is provided in Appendix B.

Inputs/Outputs

All inputs and outputs for SFCALL are the same as those for the SUDFRD routine described in Section 3.11, except for the variables INPUT, THALF, RKD, and RHO. INPUT is an integer to identify if the input will be interactive, using a file, or whether the user wants to leave the routine before conducting calculations. THALF, RKD, and RHO are discussed in Table A-3.

Two files are opened by the SFCALL subroutine for input and output. Unit 3 is opened with a file name of SUDFRD.IN for input data and Unit 4 is opened with the file name SUDFRD.OUT for the output file.

Remarks

If an inappropriate value is used for one of the input parameters, an error message will be printed and no results will be provided. The error conditions for SFCALL are the same as those for SUDFRD discussed in Section 3.11 with additional conditions as noted in Table A-3. In the case of an inappropriate input the following error message will appear on the screen and in the output file:

```
**********************
*** INPUT ERROR ***
**********************
```

Table A-3. Inputs and outputs for SFCALL different from SUDFRD (L = length, M = mass, T = time).

FORTRAN name	NUREG symbol	Type	Units	Description	Remarks
THALF	N/A	Real	T	Radionuclide half-life used to calculate the SUDFRD input DECAY prior to calling SUDFRD. Where, DECAY = LN[2]/THALF ($>10^{-15}$)	INPUT
RKD	K_d	Real	L^3/M	Radionuclide distribution coefficient used to calculate the SUDFRD input RD prior to calling SUDFRD. Where, RD = 1+RHO*RKD/POR ($>10^{-15}$)	INPUT
RHO	ρ_b	Real	M/L^3	Dry bulk density of concrete used to calculate RD	INPUT

Appendix B

Examples of Subroutine Use and Test Data

Appendix B

Examples of Subroutine Use and Test Data

INTRODUCTION

Example interactive screens are provided for each of the routines discussed in Appendix A. The examples all use option "0" for INPUT, which is to use interactive input. Following each example interactive session, an example input file using the same input parameters is provided. Each file can be used if option "1" is selected for inputting data using a file. After selecting option "1", the routine will proceed to solving the problem with no further input necessary. If option "1" is selected and the data file is not complete, the following message will appear on the screen and the user will be returned to the menu in the main driver requesting if the user would like to conduct another simulation:

```
**ATTEMPTED TO READ PAST END OF INPUT FILE **
**All inputs have not been included in file**
**or a blank line needs to be added to end **
**of input file (extra line is required) **
LEAVING SUBROUTINE
```

Note that an extra line is required in the input file after the last input parameter. In the example input files, the bottom half of the file is a list of the input parameters in the order they are read. Including this information at the bottom of the file satisfies the requirement for an extra line.

EXAMPLE SESSION FOR CRBCAL

```
*****************************************************

        -- DRIVER FOR CARBNA SUBROUTINE --

        Estimates depth of carbonation in
        concrete as a function of time.

*******************************          **************

HOW DO YOU WANT TO INPUT THE DATA ?
0 = interactive on the screen
1 = input file, CARBNA.IN must be in current dir.
2 = leave routine
0
*** ENTER VARIABLES ***

ENTER NUMBER OF TIMES TO CONSIDER (0<N<=
11
ENTER START TIME (YR AFTER START OF CARBONATION) =>
0
ENTER TIME INCREMENT (yr) (TIMINC >= 1) =>
100
ENTER CO2 INTRINSIC DIFF. COEFF. (cm ^ 2/s) =>
9.E-7
ENTER TOTAL INORGANIC CARBON CONC.
IN GROUNDWATER (moles/cm ^ 3) =>
5.E-8
ENTER Ca CONC. IN CONCRETE (moles/cm ^ 3) =>
2.E-2

********** Input Parameters ***********
Number of Times                    = 11
Starting Time (yr)                 = 0.0E-01
Time increment (yr)                = 1.0E+02
Intrinsic Diff. Coeff. (cm ^ 2/s)  = 9.0E-07
TIC in Groundwater (moles/cm ^ 3)  = 5.0E-08
Ca in Concrete (moles/cm ^ 3)      = 2.0E-02

******* RESULTS *********
Time(yr)      Depth(cm)
0.0E-01       0.0E-01
1.0E+02       1.2E-01
2.0E+02       1.7E-01
3.0E+02       2.1E-01
4.0E+02       2.4E-01
5.0E+02       2.7E-01
6.0E+02       2.9E-01
7.0E+02       3.2E-01
8.0E+02       3.4E-01
9.0E+02       3.6E-01
1.0E+03       3.8E-01

**************************

OUTPUT FILE => 'CARBNA.OUT'
```

EXAMPLE CARBNA.IN AND CARBNA.OUT FILES

```
11
0.
100.
9.E-7
5.E-8
2.E-2
*** Inputs in Order for test driver for CARBNA
Number of Times
Initial Time (yr)
Time Increment (yr)
Intrinsic diffusion coefficient (cm ^ 2/s)
TIC concentration in groundwater (moles/cm ^ 3)
Ca concentration in concrete (moles/cm ^ 3)
```

CARBNA.OUT Version 1.0

```
********** Input Parameters ************
Number of Times                          = 11
Starting Time (yr)                       = 0.0E-01
Time increment (yr)                      = 1.0E+02
intrinsic Diff. Coeff. (cm ^ 2/s)        = 9.0E-07
Ca in Groundwater (moles/cm ^ 3)         = 5.0E-08
Ca in Concrete (moles/cm ^ 3)            = 2.0E-02

******* RESULTS *********
Time(yr)        Depth(cm)
0.0E-01         0.0E-01
1.0E+02         1.2E-01
2.0E+02         1.7E-01
3.0E+02         2.1E-01
4.0E+02         2.4E-01
5.0E+02         2.7E-01
6.0E+02         2.9E-01
7.0E+02         3.2E-01
8.0E+02         3.4E-01
9.0E+02         3.6E-01
1.0E+03         3.8E-01
**************************
```

EXAMPLE SESSION FOR CLACAL

```
*****************************************************

        - DRIVER FOR CLATIM SUBROUTINE -

     Estimates time until onset of rebar corrosion.

*****************************************************

HOW DO YOU WANT TO INPUT THE DATA ?
0 = interactive on the screen
1 = input file, CLATIM.IN must be in current dir.
2 = leave routine

0
*** ENTER VARIABLES ***
ENTER THICKNESS OF CONCRETE ABOVE REBAR (in) =>
2
ENTER WATER TO CEMENT RATIO =>
.5
ENTER Cl- ION CONC. IN GROUNDWATER (ppm) =>
1

********** Input Parameters ***********
Depth of rebar in concrete (in)  =  2.0E+00
Water to Cement ratio            =  5.0E-01
Cl- in Groundwater (ppm)         =  1.0E+00

*********** RESULTS ***********
Estimated time to onset of rebar
corrosion is 6.0E+02 years.
*********************************

OUTPUT FILE => 'CLATIM.OUT'
```

EXAMPLE CLATIM.IN AND CLATIM.OUT FILES

```
2
.5
1.0
*** Inputs in Order for test driver for CLATIM
Thickness of Concrete above rebar (in)
Water-to-Cement ratio
Cl- ion concentration in groundwater (ppm)
```

```
CLATIM.OUT, Version 1.0

********** Input Parameters **********
Depth of rebar in concrete (in) =  2.0E+00
Water to Cement ratio           =  5.0E-01
Cl- in Groundwater (ppm)        =  1.0E+00

********** RESULTS **********
Estimated time to onset of rebar
corrosion is 6.0E+02 years.
**********
```

EXAMPLE SESSION FOR CORCAL

```
*****************************************************

 - DRIVER FOR CORROD SUBROUTINE -

Estimates percent of rebar remaining
as a function of time as a result of
oxygen diffusion and hydrogen evolution (passive) corrosion.

*****************************************************

HOW DO YOU WANT TO INPUT THE DATA ?
0 = interactive on the screen
1 = input file, CORROD.IN must be in current dir.
2 = leave routine

0
*** ENTER VARIABLES ***

ENTER NUMBER OF TIMES TO CONSIDER (0<N<=100)
=>
11
ENTER START TIME (YR AFTER START OF CORROSION)
=>
0
ENTER TIME INCREMENT (yr) (TIMINC >= 1) =>
100
ENTER THICKNESS OF CONCRETE ABOVE REBAR (in) =>
2
ENTER REBAR SPACING (in) =>
12
ENTER REBAR DIAMETER (in) =>
.5
ENTER O2 CONC. IN GROUNDWATER (moles/cm ^ 3) =>
3.125E-7
ENTER O2 INTRINSIC DIFF. COEFF. (cm ^ 2/s) =>
9.E-7
ENTER CONSTANT CORROSION RATE DUE TO
HYDROGEN EVOLUTION REACTION (cm/yr) =>
0.0003
ENTER TIME UNTIL ONSET OF O2 CORROSION (YR) = >
100.0

********** Input Parameters ***********
Number of Times                    =  11
Starting Time (yr)                 =  0.0E-01
Time Increment (yr)                =  1.0E+02
Depth of rebar in concrete (in)    =  2.0E+00
Rebar diameter (in)                =  5.0E-01
Rebar spacing (in)                 =  1.2E+01
Intrinsic Diff. Coeff. (cm ^ 2/s)  =  9.0E-07
O2 in Groundwater (moles/cm ^ 3)   =  3.1E-07
HER corrosion rate (cm/yr)         =  3.0E-04
Time until O2 corrosion (yr)       =  1.0E+02

********** RESULTS **********
Time(yr)          Percent Remaining
          O2 corrosion    HER corrosion    Total
0.0E-01   1.0E-02         1.0E+02          1.0E+02
1.0E+2    1.0E+02         9.1E+01          9.1E+01
2.0E+02   9.6E+01         8.2E+01          7.8E+01
3.0E+02   9.2E+01         7.4E+01          6.6E+01
4.0E+02   8.8E+01         6.6E+01          5.4E+01
5.0E+02   8.4E+01         5.8E+01          4.3E+01
6.0E+02   8.0E+01         5.1E+01          3.2E+01
7.0E+02   7.6E+01         4.5E+01          2.1E+01
8.0E+02   7.2E+01         3.9E+01          1.1E+01
9.0E+02   6.8E+01         3.3E+01          1.4E+00
1.0E+03   6.4E+01         2.8E+01          0.0E-01
*****************************************************

OUTPUT FILE => 'CORROD.OUT'
```

EXAMPLE CORROD.IN AND CORROD.OUT FILES

```
11
0.
100.
2.
12.
.5
3.125E-7
9.E-7
0.0003
100.0
*** Inputs in Order for test driver for CORROD
Number of times to consider
Starting Time (yr)
Time increment (yr)
Thickness of concrete above rebar (in)
Rebar spacing (in)
Rebar diameter (in)
O2 concentration in groundwater (moles/cm ^ 3)
O2 intrinsic diffusion coefficient (cm ^ 2/s)
HER (passive) corrosion rate (cm/yr)
Time until onset of O2 corrosion (yr)

  CORROD.OUT, Version 1.1

********** Input Parameters **********
Number of Times                    = 11
Starting Time (yr)                 = 0.0E+00
Time Increment (yr)                = 1.0E+02
Depth of rebar in concrete (in)    = 2.0E+00
Rebar diameter (in.)               = 5.0E-01
Rebar spacing (in.)                = 1.2E+01
Intrinsic Diff. Coeff. (cm ^ 2/s)  = 9.0E-07
O2 in Groundwater (moles/cm ^ 3)   = 3.1E-07
HER corrosion rate (cm/yr)         = 3.0E-04
Time until O2 corrosion (yr)       = 1.0E+02

********** RESULTS **********
Time(yr)          Percent Remaining
          O2 corrosion   HER corrosion   Total
0.0E+00   1.0E+02        1.0E+02         1.0E+02
1.0E+2    1.0E+02        9.1E+01         9.1E+01
2.0E+02   9.6E+01        8.2E+01         7.8E+01
3.0E+02   9.2E+01        7.4E+01         6.6E+01
4.0E+02   8.8E+01        6.6E+01         5.4E+01
5.0E+02   8.4E+01        5.8E+01         4.3E+01
6.0E+02   8.0E+01        5.1E+01         3.2E+01
7.0E+02   7.6E+01        4.5E+01         2.1E+01
8.0E+02   7.2E+01        3.9E+01         1.1E+01
9.0E+02   6.8E+01        3.3E+01         1.4E+00
1.0E+03   6.4E+01        2.8E+01         0.0E+00
**********
```

EXAMPLE SESSION FOR FR1CAL

```
*****************************************************

        -- DRIVER FOR FRZTH1 SUBROUTINE --

Estimates fractional decrease in dynamic
modulus of elasticity, time to reach 50%
reduction in dyn. mod. of elast., and
annual rate of degradation.

*****************************************************

HOW DO YOU WANT TO INPUT THE DATA ?
0  =  interactive on the screen
1  =  input file, FRZTH1.IN must be in current dir.
2  =  leave routine

0
*** ENTER VARIABLES ***

ENTER # OF FREEZE/THAW CYCLES YEARLY (>50) =>
350
ENTER % ENTRAINED AIR IN CONCRETE (1<AIR<20) =>
7
ENTER WATER TO CEMENT RATIO =>
.5
ENTER CONCRETE POROSITY (0<PHI<1) =>
.3
ENTER RESIDUAL MOISTURE CONTENT (0<TR<1) =>
8.94E-2

********** Input Parameters ***********
# of Frz/Thaw cycles per year  =  350
Percent entrained air         =  7.0E+00
Water to Cement ratio         =  5.0E-01
Porosity of concrete          =  3.0E-01
Residual moisture content     =  8.9E-02

**************** RESULTS ****************
Estimated annual fractional decrease in
dynamic modulus of elasticity is 3.9E-02
and estimated time to reach 50% reduction
is 1.1E+01 years.

Estimated annual degradation rate due to
freeze/thaw is 2.3E+00 cm/yr.
*****************************************

OUTPUT FILE => 'FRZTH1.OUT'
```

EXAMPLE FRZTH1.IN AND FRZTH1.OUT FILES

```
350
7.
.5
.3
8.94E-2
*** Inputs in Order for test driver for FRZTH1
Number of freeze/thaw cycles per year
% entrained air in concrete
Water-to-Cement ratio
Concrete porosity
Residual moisture content

   FRZTH1.OUT, Version 1.0
```

```
********** Input Parameters ***********
# of Frz/Thaw cycles per year  =  350
Percent entrained air          =  7.0E+00
Water to Cement ratio          =  5.0E-01
Porosity of concrete           =  3.0E-01
Residual moisture content      =  8.9E-02

**************** RESULTS ****************
Estimated annual fractional decrease in
dynamic modulus of elasticity is 3.9E-02
and estimated time to reach 50% reduction
is 1.1E+01 years.
Estimated annual degradation rate due to
freeze/thaw is 2.3E+00 cm/yr.
*****************************************
```

EXAMPLE SESSION FOR CLCCAL

```
*****************************************************

        - DRIVER FOR LCHCNC SUBROUTINE -

Estimates depth of concrete controlled
leaching as a function of time.

*****************************************************

HOW DO YOU WANT TO INPUT THE DATA ?
0 = interactive on the screen
1 = input file, LCHCNC.IN must be in current dir.
2 = leave routine

0
*** ENTER VARIABLES ***

ENTER NUMBER OF TIMES TO CONSIDER (0<N<=100)
=>
11
ENTER START TIME (YR AFTER START OF LEACHING) =>
0
ENTER TIME INCREMENT (yr) (TIMINC >= 1) =>
100
ENTER Ca INTRIN. DIFF. COEF. IN CONCRETE (cm ^ 2/s)
=>
9.E-7
ENTER Ca CONC. IN PORE SOLUTION (moles/cm ^ 3) =>
2.7E-6
ENTER Ca CONC. IN GROUNDWATER (moles/cm ^ 3) =>
5.E-8
ENTER Ca CONC. IN CONCRETE (moles/cm ^ 3) =>
2.E-2

********** Input Parameters ***********
Number of Times                    = 11
Starting Time (yr)                 = 0.0E-01
Time Increment (yr)                = 1.0E+02
Intrinsic Diff. Coeff. (cm ^ 2/s)  = 9.0E-07
Ca in Pore Solution (moles/cm ^ 3) = 2.7E-06
Ca in Groundwater (moles/cm ^ 3)   = 5.0E-08
Ca in Concrete (moles/cm ^ 3)      = 2.0E-02

******* RESULTS *********
Time(yr)      Depth(cm)
0.0E-01       0.0E-01
1.0E+02       8.7E-01
2.0E+02       1.2E+00
3.0E+02       1.5E+00
4.0E+02       1.7E+00
5.0E+02       1.9E+00
6.0E+02       2.1E+00
7.0E+02       2.3E+00
8.0E+02       2.5E+00
9.0E+02       2.6E+00
1.0E+03       2.7E+00

*************************

OUTPUT FILE => 'LCHCNC.OUT'
```

EXAMPLE LCHCNC.IN AND LCHCNC.OUT FILES

```
11
0.
100.
9.E-7
2.7E-6
5.E-8
2.E-2
*** Inputs in Order for test driver for LCHCNC
Number of times to consider
Starting Time (yr)
Time increment (yr)
Ca Intrinsic diffusion coeff. in concrete (cm ^ 2/s)
Ca concentration in pore solution (moles/cm ^ 3)
Ca concentration in groundwater (moles/cm ^ 3)
Ca concentration in concrete (moles/cm ^ 3)

  LCHCNC.OUT, Version 1.0

********** Input Parameters **********
Number of Times                    = 11
Starting Time (yr)                 = 0.0E-01
Time Increment (yr)                = 1.0E+02
Intrinsic Diff. Coeff. (cm ^ 2/s)  = 9.0E-07
Ca in Pore Solution (moles/cm ^ 3) = 2.7E-06
Ca in Groundwater (moles/cm ^ 3)   = 5.0E-08
Ca in Concrete (moles/cm ^ 3)      = 2.0E-02

******** RESULTS *******
Time(yr)      Depth(cm)
0.0E-01       0.0E-01
1.0E+02       8.7E-01
2.0E+02       1.2E+00
3.0E+02       1.5E+00
4.0E+02       1.7E+00
5.0E+02       1.9E+00
6.0E+02       2.1E+00
7.0E+02       2.3E+00
8.0E+02       2.5E+00
9.0E+02       2.6E+00
1.0E+03       2.7E+00

**************************
```

EXAMPLE SESSION FOR GLCCAL

```
*****************************************************

        - DRIVER FOR LCHGEO SUBROUTINE -

Estimates depth of geology controlled
leaching as a function of time.

*****************************************************


HOW DO YOU WANT TO INPUT THE DATA ?
0 = interactive on the screen
1 = input file, LCHGEO.IN must be in current dir.
2 = Inave routine

0
*** ENTER VARIABLES ***

ENTER NUMBER OF TIMES TO CONSIDER (0<N<=100) =>
11
ENTER START TIME (YR AFTER START OF LEACHING) =>
0
ENTER TIME INCREMENT (yr) (TIMINC >= 1) =>
100
ENTER Ca EFF. DIFF. COEFF. IN SOIL (cm ^ 2/s) =>
3.E-6
ENTER RETARDATION COEFF. IN SOIL (1<=RD) =>
5
ENTER SOIL POROSITY (0<PHI<=1) =>
.3
ENTER Ca CONC. IN PORE SOLUTION (moles/cm ^ 3) =>
2.7E-6
ENTER Ca CONC. IN GROUNDWATER (moles/cm ^ 3) =>
5.E-8
ENTER Ca CONC. IN CONCRETE (moles/cm ^ 3) =>
2.E-2

********** Input Parameters ***********
Number of Times                    = 11
Starting Time (yr)                 = 0.0E-01
Time Increment (yr)                = 1.0E+02
Soil Eff. Diff. Coeff.(cm ^ 2/s)   = 3.0E-06
Soil Retardation Factor            = 5.0E+00
Soil Porosity                      = 3.0E-01
Ca in Pore Solution (moles/cm ^ 3) = 2.7E-06
Ca in Groundwater (moles/cm ^ 3)   = 5.0E-08
Ca in Concrete (moles/cm ^ 3)      = 2.0E-02

******* RESULTS *********
Time(yr)      Depth(cm)
0.0E-01       0.0E-01
1.0E+02       9.8E-03
2.0E+02       1.4E-02
3.0E+02       1.7E-02
4.0E+02       2.0E-02
5.0E+02       2.2E-02
6.0E+02       2.4E-02
7.0E+02       2.6E-02
8.0E+02       2.8E-02
9.0E+02       2.9E-02
1.0E+03       3.1E-02

*************************

OUTPUT FILE => 'LCHGEO.OUT'
```

EXAMPLE LCHGEO.IN AND LCHGEO.OUT FILES

```
11
0.
100.
3.E-6
5.
.3
2.7E-6
5.E-8
2.E-2
*** Inputs in Order for test driver for LCHGEO
Number of times to consider
Starting Time (yr)
Time increment (yr)
Ca effective diffusion coeff. in soil (cm ^ 2/s)
Ca retardation coefficient in soil
Soil porosity
Ca concentration in pore solution (moles/cm ^ 3)
Ca concentration in groundwater (moles/cm ^ 3)
Ca concentration in concrete (moles/cm ^ 3)

  LCHGEO.OUT, Version 1.0

********** Input Parameters ***********
Number of Times                          = 11
Starting Time (yr)                       = 0.0E-01
Time Increment (yr)                      = 1.0E+02
Soil Eff. Diff. Coeff.(cm ^ 2/s)         = 3.0E-06
Soil Retardation Factor                  = 5.0E+00
Soil Porosity                            = 3.0E-01
Ca in Pore Solution (moles/cm ^ 3)       = 2.7E-06
Ca in Groundwater (moles/cm ^ 3)         = 5.0E-08
Ca in Concrete (moles/cm ^ 3)            = 2.0E-02

******* RESULTS *********
Time(yr)      Depth(cm)
0.0E-01       0.0E-01
1.0E+02       9.8E-03
2.0E+02       1.4E-02
3.0E+02       1.7E-02
4.0E+02       2.0E-02
5.0E+02       2.2E-02
6.0E+02       2.4E-02
7.0E+02       2.6E-02
8.0E+02       2.8E-02
9.0E+02       2.9E-02
1.0E+03       3.1E-02

**************************
```

EXAMPLE SESSION FOR SA1CAL

```
*****************************************************

        - DRIVER FOR SATTK1 SUBROUTINE -

Estimates depth of sulfate attack as a
function of time using an empirical model.

*****************************************************

HOW DO YOU WANT TO INPUT THE DATA ?
0  =  interactive on the screen
1  =  input file, SATTK1.IN must be in current dir.
2  =  leave routine

0
*** ENTER VARIABLES ***

ENTER NUMBER OF TIMES TO CONSIDER (0<N<=100) =>
11
ENTER START TIME (YR AFTER START OF ATTACK) =>
0
ENTER TIME INCREMENT (yr) (TIMINC >= 1) =>
100
ENTER Wt.% of C3A IN UNHYDRATED CEMENT =>
5
ENTER TOT. [SO4--]+[Mg++] CONC. IN SOLUTION[M] =>
8.1E-5

********** Input Parameters ***********
Number of Times                    =  11
Starting Time (yr)                 =  0.0E-01
Time Increment (yr)                =  1.0E+02
Wt.% C3A in Unhydrated Cement      =  5.0E+00
[SO4--]+[Mg++] in Solution (M)     =  8.1E-05

******** RESULTS ********
Time(yr)        Depth(cm)
0.0E-01         0.0E-01
1.0E+02         2.2E-02
2.0E+02         4.5E-02
3.0E+02         6.7E-02
4.0E+02         8.9E-02
5.0E+02         1.1E-01
6.0E+02         1.3E-01
7.0E+02         1.6E-01
8.0E+02         1.8E-01
9.0E+02         2.0E-01
1.0E+03         2.2E-01

**************************

OUTPUT FILE => 'SATTK1.OUT'
```

EXAMPLE SATTK1.IN AND SATTK1.OUT FILES

```
11
0.
100.
5
8.1E-5
*** Inputs in Order for test driver for SATTK1
Number of times to consider
Starting Time (yr)
Time increment (yr)
Wt% C3A in unhydrated cement
Total SO4-- + Mg++ conc. in soil water (M)

  SATTK1.OUT, Version 1.0

********** Input Parameters **********
Number of Times                    = 11
Starting Time (yr)                 = 0.0E-01
Time Increment (yr)                = 1.0E+02
Wt.% C3A in Unhydrated Cement      = 5.0E+00
[SO4--]+[Mg++] in Solution (M)     = 8.1E-05

******* RESULTS *********
Time(yr)      Depth(cm)
0.0E-01       0.0E-01
1.0E+02       2.2E-02
2.0E+02       4.5E-02
3.0E+02       6.7E-02
4.0E+02       8.9E-02
5.0E+02       1.1E-01
6.0E+02       1.3E-01
7.0E+02       1.6E-01
8.0E+02       1.8E-01
9.0E+02       2.0E-01
1.0E+03       2.2E-01

**************************
```

EXAMPLE SESSION FOR SA2CAL

```
*****************************************************

        - DRIVER FOR SATTK2 SUBROUTINE -

Estimates depth of sulfate attack as a
function of time using a mechanistic model.

*****************************************************

HOW DO YOU WANT TO INPUT THE DATA ?
0  =  interactive on the screen
1  =  input file, SATTK2.IN must be in current dir.
2  =  leave routine
0
*** ENTER VARIABLES ***
(OPC = Ordinary Portland Cement)
(SRPC = Sulfate Resistant Portland Cement)

ENTER INTRINSIC DIFF. COEFF. IN CEMENT (m ^ 2/s) =>
1.E-12
ENTER KINETIC CONSTANT FOR QUANTITY OF
   SULFATE REACTED WITH CEMENT
   (moles/kg of anhydrous cement)
   (0.32 for OPC, 0.16 for SRPC) =>
.32
ENTER QUANTITY OF SULFATE REQUIRED TO
   COMPLETELY REACT WITH CEMENT
   (moles/kg of anhydrous cement)
   (1.24 for OPC, 1.07 for SRPC) =>
1.24
ENTER SULFATE CONCENTRATION IN BULK SOLUTION
   (moles/m ^ 3) =>
5.20833
ENTER SULFATE CONC. IN KINETIC EXPERIMENTS
   (moles/m ^ 3) =>
12.2
ENTER CHARACTERISTIC TIME FOR REACTION (s)
   (3577 for OPC, 1555 for SRPC) =>
3577
ENTER MASS OF CEMENT (kg) =>
300
ENTER VOLUME OF CONCRETE (m ^ 3) =>
1

********** Input Parameters ***********
Intrinsic Diff. Coeff. (m ^ 2/s)        =  1.0E-12
m0 (moles/kg anhydrous cement)          =  3.2E-01
mc (moles/kg anhydrous cement)          =  1.2E+00
SO4 conc. in Sol. (moles/m ^ 3)         =  5.2E+00
SO4 conc. in Exp. (moles/m ^ 3)         =  1.2E+01
Characteristic reaction time (s)        =  3.6E+03
Mass of cement (kg)                     =  3.0E+02
Volume of concrete (m ^ 3)              =  1.0E+00

********** RESULTS ***********
Degradation Rate = 5.7E-02 cm/yr
**********************************

OUTPUT FILE => 'SATTK2.OUT'
```

EXAMPLE SATTK2.IN AND SATTK2.OUT FILES

1.E-12
.32
1.24
5.20833
12.2
3577.
300.
1.
*** Inputs in Order for test driver for SATTK2
Intrinsic diffusion coefficient in concrete (m ^ 2/s)
Kinetic const. for quan. of sulfate reacted w/cement (moles/kg)
Quantity of sulfate required to compl. react w/cement (moles/kg)
Sulfate concentration in bulk solution (moles/m ^ 3)
Sulfate concentration in kinetic experiments (moles/m ^ 3)
Characteristic time for reaction (s)
Mass of cement (kg)
Volume of concrete (m ^ 3)

SATTK2.OUT, Version 1.0

********** Input Parameters **********

Intrinsic Diff. Coeff. (m ^ 2/s)	= 1.0E-12
m0 (moles/kg anhydrous cement)	= 3.2E-01
mc (moles/kg anhydrous cement)	= 1.2E+00
SO4 conc. in Sol. (moles/m ^ 3)	= 5.2E+00
$SO\bar{4}$ conc. in Exp. (moles/m ^ 3)	= 1.2E+01
Characteristic reaction time (s)	= 3.6E+03
Mass of cement (kg)	= 3.0E+02
Volume of concrete (m ^ 3)	= 1.0E+00

********** RESULTS **********
Degradation Rate = 5.7E-02 cm/yr
****

EXAMPLE SESSION FOR FLWCAL

```
*****************************************************

          --DRIVER FOR FLOW SUBROUTINE --

Estimates flow rate through porous media above
vault and cracks in vault roof.

*****************************************************

HOW DO YOU WANT TO INPUT THE DATA ?
0  =  interactive on the screen
1  =  input file, FLOW.IN must be in current dir.
2  =  leave routine

0
*** USE CONSISTENT UNITS ***

ENTER CRACK WIDTH =>
.1
ENTER CRACK SPACING =>
100
ENTER DEPTH OF PERCHED WATER =>
50
ENTER HYD. COND. OF POROUS MEDIA =>
1.5
** Running ... **

*** Input Parameters ***
Crack Width (crack)                 =  1.0E-01
Crack Spacing (space)               =  1.0E+02
Depth of Perched Water (perch)      =  5.0E+01
Hyd. Cond. of Porous Media (perm)   =  1.5E+00

*** Dimensionless Variables ***
Gap Fraction (crkfrc)               =  1.0E-03
z0 (smalz0)                         =  1.0E+03

************ RESULTS ************
ZETA3 is 1.9E-01, which yields an
effective flow rate of 2.8E-01
**********************************

OUTPUT FILE => 'FLOW.OUT'
```

EXAMPLE FLOW.IN AND FLOW.OUT FILES

```
0.1
100.
50.
1.5
*** Inputs in Order for test program for FLOW subroutine ***
(use consistent units -> L=length,T=time)
Crack Width (L),
Crack Spacing (L),
Depth of Perched Water (L),
Hyd. Cond. of Porous Media (L/T),
```

FLOW.OUT, Version 1.0

```
*** Input Parameters ***
Crack Width (crack)                = 1.0E-01
Crack Spacing (space)              = 1.0E+02
Depth of Perched Water (perch)     = 5.0E+01
Hyd. Cond. of Porous Media (perm) = 1.5E+00

*** Dimensionless Variables ***
Gap Fraction (crkfrc)              = 1.0E-03
z0 (smalz0)                        = 1.0E+03

********** RESULTS **********
ZETA3 is 1.9E-01, which yields an
effective flow rate of 2.8E-01
**********
```

EXAMPLE SESSION FOR RNCALL

```
*******************************************************

        - DRIVER FOR RNNOD SUBROUTINE -

Estimates concentration and release
rate of radionuclide from fracture in
a concrete waste form.

*******************************************************

HOW DO YOU WANT TO INPUT THE DATA ?
0 = interactive on the screen
1 = input file, RNNOD.IN must be in current dir.
2 = leave routine
0
*** USE CONSISTENT UNITS ***

ENTER TOTAL CONC. IN CONCRETE MATRIX AND PORES =>
2.
ENTER SIMULATION TIME =>
10.
ENTER DARCY VELOCITY THROUGH CRACKED
  CONCRETE =>
.001
ENTER CRACK WIDTH =>
.01
ENTER CRACK LENGTH =>
100.
ENTER CRACK SPACING =>
10
ENTER CONCRETE KD =>
100.
ENTER CONCRETE DRY BULK DENSITY =>
1.3
ENTER CONCRETE POROSITY =>
.5
ENTER CONCRETE EFFECTIVE DIFFUSION COEFFICIENT
(* USE CONSISTENT UNITS *) =>
0.01
ENTER RADIONUCLIDE HALF-LIFE =>
12.
ENTER EPS (TYPICALLY 1.E-7) =>
1.E-7
** Running ... **

********** Input Parameters ***********
Total Concentration (totalc)      = 2.0E+00
Simulation Time (time)            = 1.0E+01
Average Darcy Vel.(darv)          = 1.0E-03
Crack Width (crack)               = 1.0E-02
Crack Length (dist)               = 1.0E+02
Crack Spacing (space)             = 1.0E+01
Distribution Coeff. (rkd)         = 1.0E+02
Dry Bulk Density (rho)            = 1.3E+00
Porosity (por)                    = 5.0E-01
Eff. Diffusion Coeff. (de)        = 1.0E-02
Half-Life (thalf)                 = 1.2E+01
EPS                               = 1.0E-07
*********************************"*******

********** RESULTS ***********************
(M=mass,L=length,T=time)
Fractional Concentration computed by RNNOD = 5.6E-01
Concentration in Crack Water is 8.6E-03 (M/L^3)
The corresponding average release rate is 8.6E-06
(M/L^2-T)
*******************************************************

OUTPUT FILE => 'RNNOD.OUT'
```

EXAMPLE RNNOD.IN AND RNNOD.OUT FILES

```
2.
10.
0.001
0.01
100.
10.
100.
1.3
0.5
0.01
12.
1.E-7
*** Inputs in Order for test driver for RNNOD
(use consistent units->L=length,T=time,M=mass)
TotalC (M/L^3),
Time (T),
Darv (L/T),
Crack Width (L),
Crack Length (L),
Crack Spacing (L),
Radionuclide Kd in Concrete (L^3/M),
Concrete Dry Bulk Density (M/L^3),
Concrete Porosity (-),
Concrete Effective Diffusion Coefficient (L^2/T),
Radionuclide Half-Life (T),
Convergence Criteria (typically 1.E-7)
```

```
  RNNOD.OUT, Version 1.0

********** Input Parameters ***********
Total Concentration (totalc)      = 2.0E+00
Simulation Time (time)            = 1.0E+01
Average Darcy Vel.(darv)          = 1.0E-03
Crack Width (crack)               = 1.0E-02
Crack Length (dist)               = 1.0E+02
Crack Spacing (space)             = 1.0E+01
Distribution Coeff. (rkd)         = 1.0E+02
Dry Bulk Density (rho)            = 1.3E+00
Porosity (por)                    = 5.0E-01
Eff. Diffusion Coeff. (de)        = 1.0E-02
Half-Life (thalf)                 = 1.0E+02
EPS                               = 1.0E-07
**************************************

********** RESULTS **********
(M=mass,L=length,T=time)
Fractional Concentration computed by RNNOD = 5.6E-01
Concentration in Crack Water is 8.6E-03 (M/L^3)
The corresponding average release rate is 8.6E-06
(M/L^2-T)
**************************************************
```

EXAMPLE SESSION FOR SFCALL

```
*******************************************************

        - DRIVER FOR SUDFRD SUBROUTINE -

Estimates thickness of cracked concrete floor
that will result in a specified reduction in
concentration due to decay and/or sorption
as a radionuclide passes through the floor.

*******************************************************

HOW DO YOU WANT TO INPUT THE DATA ?
0 = interactive on the screen
1 = input file, SUDFRD.IN must be in current dir.
2 = leave routine

0
*** USE CONSISTENT UNITS ***

ENTER DARCY VELOCITY THROUGH CRACKED
  CONCRETE =>
.05
ENTER RADIONUCLIDE HALF-LIFE =>
5730
ENTER DESIRED REDUCTION IN CONCENTRATION =>
.1
ENTER CONCRETE KD =>
5000
ENTER CONCRETE DRY BULK DENSITY =>
1.3
ENTER CONCRETE EFFECTIVE DIFFUSION COEFFICIENT
  (* USE CONSISTENT UNITS *) =>
0.01
ENTER CRACK WIDTH =>
.5
ENTER CRACK SPACING =>
100
ENTER CONCRETE POROSITY =>
.4
** Running ... **

*** Input Parameters ***
Average Darcy Vel.(darv)        = 5.0E-02
Half-Life (thalf)               = 5.7E+03
Attenuation Factor (deltc)      = 1.0E-01
Distribution Coeff. (rkd)       = 5.0E+03
Dry Bulk Density (rho)          = 1.3E+00
Eff. Diffusion Coeff. (de)      = 1.0E-02
Crack Width (crack)             = 5.0E-01
Crack Spacing (space)           = 1.0E+02
Porosity (por)                  = 4.0E-01

********** RESULTS **********
(L=length)
Thickness of Concrete required to attenuate the concentra-
tion by a factor of 1.0E-01 is 1.0E+02 (L)
*******************************************************

OUTPUT FILE => 'SUDFRD.OUT'
```

EXAMPLE SUDFRD.IN AND SUDFRD.OUT FILES

```
0.05
5730.
0.1
5000.
1.3
0.01
0.5
100.
0.4
*** Inputs in Order for test driver to SUDFRD ***
(use consistent units -> L=length, T=time, M=mass)
Darv (L/T),
Radionuclide Half-Life (T),
Concentration Reduction (-),
Radionuclide Kd in Concrete (L^3/M),
Concrete Dry Bulk Density (M/L^3),
Concrete Effective Diffusion Coefficient (L^2/T),
Crack Width (L),
Crack Spacing (L),
Concrete Porosity (-)

  SUDFRD.OUT, Version 1.0

*** Input Parameters ***
Average Darcy Vel.(darv)        = 5.0E-02
Half-Life (thalf)               = 5.7E+03
Attenuation Factor (deltc)      = 1.0E-01
Distribution Coeff. (rkd)       = 5.0E+03
Dry Bulk Density (rho)          = 1.3E+00
Eff. Diffusion Coeff. (de)      = 1.0E-02
Crack Width (crack)             = 5.0E-01
Crack Spacing (space)           = 1.0E+02
Porosity (por)                  = 4.0E-01

********** RESULTS **********
(L=length)
Thickness of Concrete required to attenuate the concentration by a factor of 1.0E-01 is 1.0E+02 (L)
**********
```

Appendix C

Programmer Notes

Appendix C

Programmer Notes

This section summarizes the compiler options and portability information for CEMENT. The primary use of this code is for running on an IBM PC or compatible using the LAHEY FORTRAN compiler. Some compiler specific concerns are identified. In general, the code has been written to be easily ported to different platforms. It has been run on MacIntosh and NeXT computers using the ABSOFT FORTRAN compiler with minimal changes. These changes are discussed below.

The LAHEY FORTRAN compiler, version 5.00 was used for this program. The default options for LAHEY FORTRAN were used. The following options were used for the ABSOFT compiler: -O, -C, -e, -g, -s, -f, and -c.

In the main driver, it may be necessary to use the line that writes "Press ENTER to Continue" after the disclaimer is written to the screen. This is Line 103 in Version 1.01 of CEMENT. LAHEY automatically writes this when a PAUSE command is used. However, other compilers (e.g., ABSOFT) may not automatically write this statement to the screen. The line with the write statement is a comment in the LAHEY version of the computer code. To execute the line, simply delete the "C" in column 1 in front of the write statement.

In each of the individual driver routines, a logical variable called HELP is provided. The variable is currently inactive except for being initialized in each of the driver routines. The variable is passed in the call to the subroutine associated with each driver, but all references to the variable within the subroutines are commented. If the user wishes to activate HELP for any subroutine, the following steps are necessary. The user must first add HELP to the end of the READ statement for the input file in the driver and must also remove the comments at the start of the lines for interactive input of the HELP variable in the driver. Most of the subroutines do not have any coding to take advantage of HELP, thus, the user will need to add any required coding to the routines. It may also be necessary to add an OPEN statement at the start of the subroutine for a debug output file if the user desires.

Appendix D

Configuration Control

Appendix D

Configuration Control

The VCS3 system was used to maintain all files associated with the development and release of the concrete analysis modules. The configuration management of the software product during development is essential to the delivery of error-free and reliable software, and to ensure compliance with the requirements of the applicable Software Engineering Standard Practice.

The SCMP provides information regarding the requirements and procedures necessary for the configuration management activities to be used in development of the software product. The purpose of the plan is to: (a) prevent unauthorized or undocumented changes to software or documentation under configuration control, (b) establish a method of controlling software such that revision status can always be ascertained, and (c) establish a method of controlling software such that a complete change history can be produced for each controlled software item.

Policies, directives, and procedures applicable to the SCMP include the following:

- Prevent unauthorized or undocumented changes to software or documentation under configuration control
- Control software to ensure that revision status can always be ascertained
- Control software to ensure that a complete change history can be produced for each controlled software item.

The implementing procedure and principal tool is used generally to establish a directory structure for all CIs within a UNIX file system, apply file and directory permissions to restrict access, and use VCS as a (Quality Level B) tool to provide revision status and change history on a file-by-file basis.

Appendix E

U.S. Nuclear Regulatory Commission
Software Transmittal Form

Appendix E

U.S. Nuclear Regulatory Commission
Software Transmittal Form

NRC Scientific Software Submittal Package Description Form

Program Name: CEMENT

Programming Language(s) Used: FORTRAN 77

Machine: IBM PC

Core Storage Requirements for Sample Problem: < 200 kb

Approximate Execution Time for Sample Problem:

Described in #4-3, Page E–4.

Package Contents (Items 1–8 are mandatory):

Yes 1. Abstract

Yes 2. Documentation consisting of the items described in the space below:

a. Plansky, L. E. and R. R. Seitz, *User's Guide for Simplified Computer Models for the Estimation of Long-Term Performance of Cement-Based Materials*, EGG-WM-10793, October 1993

Yes 3. Source program on diskette.

Yes 4. Sample problem input on diskette.

Yes 5. Compilation of the source program (Item 3).

Given on the disk, CEMENT.EXE.

Yes 6. Listing of the sample problem input (Item 4).

Given on the disk and in Appendix B of Item 2a, above.

Yes 7. Output from an execution of the sample problem input, Item 4, using the source code provided in Item 3, including plots, if any.

Given in Item 2a and on the disk. Sample inputs are provide for all 11 subroutines.

No 8. For tape submittals, a copy of the output for the job that created the tape.

Not Applicable.

Yes 9. Other (describe other materials such as data libraries, control information, etc., in the space below).

N/A

Code Portability Information:

1. Provide name and description of the required system library routines not included in the submittal, and indicate where these routines are called (subroutines name, line).

   N/A

2. Provide name and description of required FORTRAN library routines (internal or external) which may differ between mainframes, and indicate where these routines are called (subroutine name, line).

   N/A

3. Describe any special compiler or loader options used, such as:

   1. Core present to zero or other values (CDC PRESET=ZERO)

   2. Compiler optimization level

   3. Rounding or truncation options.

      All default LAHEY FORTRAN, Version 4.00 compiler options.

      Math coprocessor is required.

4. Provide any special instructions regarding execution time or core storage requirements, such as:

   1. Special core storage requirements for loading or executing

   2. Instructions for adjusting required core storage by increasing or decreasing array dimensions

   3. Estimates of relation of execution time to critical input parameters.

   Run times for all routines except FLOW and RNNOD are generally very short (seconds). All test problem run times are on the order of seconds on a 486 PC.

   For the routine FLOW, run times are generally very short (seconds), but for large ratios ($>10^5$) of the depth of perched water (PERCH) to the crack half-width (CRACK/2) the run times can be longer depending on the type of computer being used. In general, run times on a 386 computer should be shorter than a minute.

   For the routine RNNOD, cases that require a large number of iterations also tend to take a long time to run. The run time is a function of the rate at which concentrations are changing in the

system and the length of the simulation time period. For example, a 10,000 year simulation for a non-sorbing radionuclide and a relatively high flow velocity through the crack (DARV/CRKFRC > 1000) can require a large amount of computer time. For larger velocities in the crack, simulations can take hours on a 386 computer. However, for simulations at lower velocities, shorter times, or including sorption, the routine will run relatively quickly (seconds to minutes).

5. Identify all input and output unit numbers and their purpose. List all locations of end-of-file tests.

   Input on FORTRAN 77 "*", interactive

   Input on FORTRAN 77 "3", local directory input required

   Output on FORTRAN 77 "4", local directory output files created

   If tapes are used for submittal, please use the attached tape description form. Be sure to include a copy of the job that created the tape. Where possible, we would prefer to receive tapes with the following format: 9 track, 1600 bpi, unlabeled, in EBCDIC character format, with fixed length block or unblocked records.

   Not applicable.

   If diskettes are used for submittal, they should be 3.5" and MS-DOS compatible.

   Delivered compatible with MS-DOS on 3.5".

NRC FORM 335 (2-89) NRCM 1102, 3201, 3202

U.S. NUCLEAR REGULATORY COMMISSION

BIBLIOGRAPHIC DATA SHEET

(See instructions on the reverse)

1. REPORT NUMBER (Assigned by NRC, Add Vol., Supp., Rev., and Addendum Numbers, If any.)

NUREG/CR-6138
EGG-2719

2. TITLE AND SUBTITLE

User's Guide for Simplified Computer Models for the Estimation of Long-Term Performance of Cement-Based Materials.

3. DATE REPORT PUBLISHED

MONTH	YEAR
February	1994

4. FIN OR GRANT NUMBER

A6858

5. AUTHOR(S)

L.E. Plansky, R.R. Seitz

6. TYPE OF REPORT

Technical

7. PERIOD COVERED *(Inclusive Dates)*

8. PERFORMING ORGANIZATION – NAME AND ADDRESS *(If NRC, provide Division, Office or Region, U.S. Nuclear Regulatory Commission, and mailing address; if contractor, provide name and mailing address.)*

Idaho National Engineering Laboratory
EG&G Idaho, Inc.
Idaho Falls, Idaho 83415

9. SPONSORING ORGANIZATION – NAME AND ADDRESS *(If NRC, type "Same as above"; if contractor, provide NRC Division, Office or Region, U.S. Nuclear Regulatory Commission, and mailing address.)*

Division of Regulatory Applications
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, D.C. 20555-0001

10. SUPPLEMENTARY NOTES

11. ABSTRACT *(200 words or less)*

This report documents user instructions for several simplified subroutines and driver programs that can be used to estimate various aspects of the long-term performance of cement-based barriers used in low-level radioactive waste disposal facilities. The subroutines are prepared in a modular fashion to allow flexibility for a variety of applications. Three levels of codes are provided: the individual subroutines, interactive drivers for each of the subroutines, and an interactive main driver, CEMENT, that calls each of the individual drivers. The individual subroutines for the different models may be taken independently and used in larger programs, or the driver modules can be used to execute the subroutines separately or as a part of the main driver routine. A brief program description is included and user-interface instructions for the individual subroutines are documented in the main report. These are intended to be used when the subroutines are used as subroutines in a larger computer code. User instructions for the drivers and example interactive screens for the main drivers are provided in Appendix A. Examples showing the use of the individual driver routines to execute the different subroutines and test data are included in Appendix B. Programmer notes are provided in Appendix C and the configuration control system is discussed in Appendix D.

12. KEY WORDS/DESCRIPTORS *(List words or phrases that will assist researchers in locating the report.)*

concrete barriers
codes
subroutine
low-level waste
cement

13. AVAILABILITY STATEMENT

Unlimited

14. SECURITY CLASSIFICATION

(This Page)

Unclassified

(This Report)

Unclassified

15. NUMBER OF PAGES

16. PRICE

Federal Recycling Program