LBNL-60134

TOUGH
Training Course

IAEA Technical Co-operation Project C7-INT 9.173.009
Numerical Simulation of Subsurface Processes

TOUGH Training Course

August 14–17, 2006

Lawrence Berkeley National Laboratory
Berkeley, California

Instructors:

Stefan Finsterle and Michael Kowalsky

with contributions from:

Jonny Rutqvist and Tianfu Xu

Lawrence Berkeley National Laboratory
Earth Sciences Division
1 Cyclotron Rd., MS 90-1116
Berkeley, CA 94720

Telephone: (510) 486-5205
E-mail: SAFinsterle@lbl.gov

This work was supported, in part, by the U.S. Dept. of Energy under Contract No. DE-AC02-05CH11231.

DISCLAIMER

IAEA Technical Co-operation Project C7-INT 9.173.009
Numerical Simulation of Subsurface Processes

TOUGH Training Course

August 14–17, 2006

Lawrence Berkeley National Laboratory
Berkeley, California

Monday, August 14, 2006

Morning Session

9:00 am *Welcome and Introduction (Bo Bodvarsson)*

9:15 am *Overview and Background of TOUGH*

- Introduction to TOUGH family of codes

10:00 am *TOUGH-FLAC: Coupled Thermal-Hydrological-Mechanical Simulator (Jonny Rutqvist)*

- Overview of TOUGH-FLAC capabilities
- TOUGH-FLAC applications

Afternoon Session

1:00 pm *Mathematical Basis*

- Physical process description
  - Multiphase flow
  - Radionuclide transport
  - Heat flow
  - Flow through fractured porous media
- Mathematical model and governing equations

3:00 pm *Numerical Solution Techniques*

- Numerical solution procedure
  - Spatial discretization
  - Temporal discretization
  - Handling nonlinearities
  - Linear equation solvers
- Code architecture
- Code installation and execution

Tuesday, August 15, 2006

Morning Session

8:30 am *Building a TOUGH Model*

- Computer exercise: Column experiment
  - Mesh generation
  - Boundary elements
  - Defining material properties

10:30 am *Initial and Boundary Conditions*

- Computer exercise: Column experiment (cont.)
  - Concept of primary variables
  - Initial conditions
  - Boundary conditions

Afternoon Session

1:00 pm *Computational Parameters*

- Computer exercise: Column experiment (cont.)
  - Time stepping
  - Convergence criteria
  - Weighting schemes

3:30 pm *Program Options*

- Computer exercise: Column experiment (cont.)
  - Output options
  - Other options and summary

Wednesday, August 16, 2006

Morning Session

8:30 am *Discussion: Modeling Nuclear Waste Repositories*

- Discussion of flow and radionuclide transport problems by participants
- Discussion of conceptual model development and data requirements

10:00 am *Radionuclide Transport*

- EOS7R module
- Computer exercise: Nuclear waste repository
  - Problem description
  - Mesh generation
  - Initial and boundary conditions
  - Properties

Afternoon Session

1:00 am *Radionuclide Transport*

- Computer exercise: Nuclear waste repository (cont.)
  - Radionuclide transport and decay

3:00 am *Gas and Heat Generation*

- Computer exercise: Nuclear waste repository (cont.)
  - Gas generation
  - Heat generation
  - Discussion

Thursday, August 17, 2006

Morning Session

8:30 am *TOUGHREACT: Reactive Geochemical Transport Simulator (Tianfu Xu)*

- Overview of TOUGHREACT capabilities
- TOUGHREACT applications

10:00 am *iTOUGH2 Introduction*

- Discussion of inverse modeling concept
- Inverse modeling theory
  - Objective function
  - Minimization algorithms
  - Residual and error analyses
  - Uncertainty propagation analysis

Afternoon Session

1:00 pm *iTOUGH2 Application*

- Overview of iTOUGH2 features and usage
- Computer exercise: Two-phase column experiment
  - Sensitivity analysis
  - Parameter estimation
  - Monte Carlo simulations

3:30 pm *Discussion and Concluding Remarks*

1. Introduction

The TOUGH family of codes is a suite of computer programs for the simulation of multiphase fluid and heat flows in porous and fractured media with applications to geothermal reservoir engineering, nuclear waste disposal in geologic formations, geologic carbon sequestration, gas hydrate research, vadose zone hydrology, environmental remediation, oil and gas reservoir engineering, and other mass transport and energy transfer problems in complex geologic settings. TOUGH has been developed in the Earth Sciences Division of the Lawrence Berkeley National Laboratory (LBNL). Many modifications and enhancements have been made to TOUGH (at LBNL and elsewhere) from the time it was first released in 1987. TOUGH and its various descendants (such as iTOUGH2, T2VOC, TMVOC, EWASG, TOUGHREACT, TOUGH+ and many more) are currently in use in approximately 300 research laboratories, private companies, and universities in 33 countries. The LBNL group, headed by Karsten Pruess, serves as custodian of the code.

The TOUGH simulators were developed for problems involving strongly heat-driven flow. To describe these phenomena a multi-phase approach to fluid and heat flow is used, which fully accounts for the movement of gaseous and liquid phases, their transport of latent and sensible heat, and phase transitions between liquid and vapor. TOUGH takes account of fluid flow in both liquid and gaseous phases—and, in certain modules, a non-aqueous phase liquid (NAPL)—occurring under pressure, viscous, and gravity forces according to Darcy's law. Interference between the phases is represented by means of relative permeability functions. The code includes Klinkenberg effects and binary diffusion in the gas phase, and capillary and phase adsorption effects for the liquid phase. Heat transport occurs by means of conduction (with thermal conductivity dependent on water saturation), convection, and binary diffusion, which includes both sensible and latent heat.

The goal of this training course is to teach participants with limited numerical modeling experience the fundamental concepts of modeling with the TOUGH family of codes. The material to be covered includes the following:

- Introduction to the TOUGH family of codes and applications
- Underlying physics, mathematical models, and numerical approaches
- Program structure and code installation
- Explanation of input and output files

The course will revolve around sample problems that are meant to familiarize users with TOUGH modeling concepts, such as grid generation, specification of material properties, initial and boundary conditions, and program control

The most common equation of state (EOS) modules will be considered for a variety of applications and levels of complexity (ranging from isothermal problems with a single component and phase, to non-isothermal problems with multiple components and phases) Examples of advanced applications from the TOUGH family of codes, will be presented.

Meeting the following requirements will be helpful for the course participants:

- Knowledge of the physics of fluid and heat flow and transport in porous and fractured media
- Basic understanding of numerical methods for solving partial differential equations
- Experience with computer programming (preferably using FORTRAN)
- Familiarity with basic text editors in the PC environment (Notepad or WordPad) and with the DOS Command Prompt will be helpful (the course will be conducted on PC computers)

An executable version of the code will be pre-installed on the computers used for the training. Users are requested to obtain a valid license of the code. Additional information can be found at http://www-esd.lbl.gov/TOUGH2.

2. TOUGH2 Summary Description

TOUGH2 is a general-purpose numerical simulation program for multi-dimensional fluid and heat flows of multiphase, multicomponent fluid mixtures in porous and fractured media. Chief application areas are in geothermal reservoir engineering, nuclear waste isolation studies, environmental assessment and remediation, and flow and transport in variably saturated media and aquifers.

TOUGH2 solves mass and energy balance equations that describe fluid and heat flow in general multiphase, multicomponent systems. Fluid advection is described with a multiphase extension of Darcy's law; in addition there is diffusive mass transport in all phases. Heat flow occurs by conduction and convection, the latter including sensible as well as latent heat effects. The description of thermodynamic conditions is based on the assumption of local equilibrium of all phases. Fluid and formation parameters can be arbitrary nonlinear functions of the primary thermodynamic variables. Thermophysical properties of water are represented, within experimental accuracy, by steam table equations provided by the International Formulation Committee. The program provides options for specifying injection or withdrawal of heat and fluids.

For numerical simulation, the continuous space and time variables must be discretized. In all members of the TOUGH family of codes, space discretization is made directly from the integral form of the basic conservation equations, without converting them into partial differential equations. This "integral finite difference" method avoids any reference to a global system of coordinates, and thus offers the advantage of being applicable to regular or irregular discretizations in one, two, and three dimensions. The IFDM also makes it possible, by means of simple preprocessing of geometric data, to implement double- and multiple-porosity methods for fractured media. No particular price needs to be paid for this flexibility; indeed, for systems of regular grid blocks referred to a global coordinate system the IFDM is completely equivalent to conventional finite differences. Time is discretized fully implicitly as a first-order backward finite difference. This together with 100 % upstream weighting of flux terms at interfaces is

necessary to avoid impractical time step limitations in flow problems involving phase (dis-)appearances, and to achieve unconditional stability.

The discretization results in a set of strongly coupled nonlinear algebraic equations, with the time-dependent primary thermodynamic variables of all grid blocks as unknowns. These equations are cast in residual form, i.e., {residual} ∫ {left-hand side} - {right-hand side} = 0, and are solved simultaneously, using Newton-Raphson iteration. Time steps can be automatically adjusted (increased or reduced) during a simulation run, depending on the convergence rate of the iteration process. Automatic time step adjustment is essential for an efficient solution of multiphase flow problems, where intrinsic time scales for significant changes in the flow system may vary by many orders of magnitude during a simulation run. Different methods are available to solve the linear equations arising at each iteration step, including preconditioned conjugate gradient solvers, as well as sparse direct matrix methods. While direct methods are more predictable and less problem-dependent in their performance, it is only through the application of iterative conjugate gradient methods that solutions for large grid systems (10,000 blocks or more) and three-dimensional problems can be accomplished.

The accuracy of TOUGH2 has been tested by comparison with many different analytical and numerical solutions, with results from laboratory experiments, and with field observations. However, it should be emphasized that the integration of many different modules into a single program structure is a difficult and potentially “hazardous” task. Many different options can be selected in different program modules. It is not practically possible to exhaustively cross-check the mutual compatibility and proper performance of all options. Users are cautioned that there is no finite process by which all program bugs that may be present can be identified and corrected. Fixing a bug may cause unanticipated problems elsewhere. Continuing vigilance and application testing are needed.

3. Main Governing Equations

β	Phase:	g = gas, l = liquid, n = NAPL, …
κ	Component:	w = water; a = air; h = heat; …

Mass- and energy-balance equation:

$$\frac{d}{dt}\int_{V_n} M^\kappa dV_n = \int_{\Gamma_n} \mathbf{F}^\kappa \bullet \mathbf{n} d\Gamma_n + \int_{V_n} q^\kappa dV_n$$

Mass accumulation:

$$M^\kappa = \phi \sum_\beta S_\beta \rho_\beta X_\beta^\kappa$$

Advective mass flux:

$$\mathbf{F}^\kappa \Big|_{adv} = \sum_\beta X_\beta^\kappa \mathbf{F}_\beta$$

$$\mathbf{F}_\beta = \rho_\beta \mathbf{u}_\beta = -k \frac{k_{r\beta}\rho_\beta}{\mu_\beta} (\nabla P_\beta - \rho_\beta \mathbf{g})$$

$$P_\beta = P + P_{c\beta}$$

Heat accumulation:

$$M^{NK+1} = (1-\phi)\rho_R C_R T + \phi \sum_\beta S_\beta \rho_\beta u_\beta$$

Heat flux:

$$\mathbf{F}^{NK+1} = -\lambda \nabla T + \sum_\beta h_\beta \mathbf{F}_\beta$$

Radionuclide decay:

$$\frac{dM^{\kappa}}{dt} = -\lambda^{\kappa} M^{\kappa}$$

Half life:

$$T_{1/2} = \frac{\ln 2}{\lambda^{\kappa}}$$

Linear equilibrium adsorption of radionuclides:

$$M^{\kappa} = \phi \sum_{\beta} S_{\beta} \rho_{\beta} X_{\beta}^{\kappa} + (1-\phi) \rho_R \rho_{aq} X_{aq}^{\kappa} K_d^{\kappa}$$

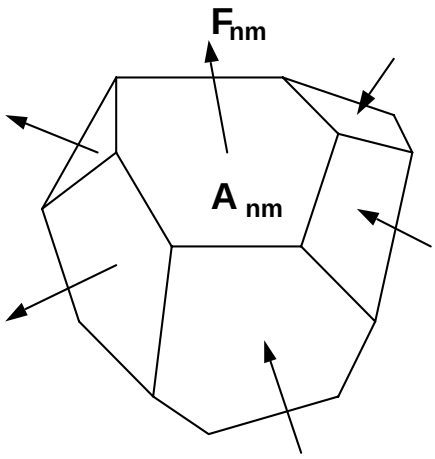
Henry's Law:

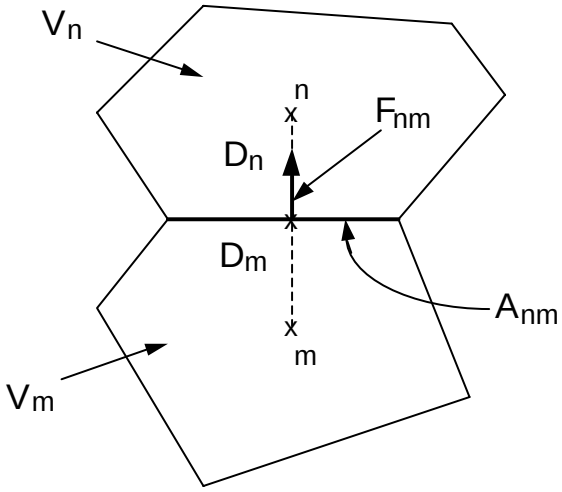
$$P_{NCG} = K_h x_{aq}^{NCG}$$

Klinkenberg effect:

$$k_g = k_l \left(1 + \frac{b}{P_g} \right)$$

4. Space and Time Discretization





Space discretization:

$$\int_{V_n} M\,dV = V_n M_n \qquad\qquad \int_{\Gamma_n} \mathbf{F}^{\kappa} \bullet \mathbf{n}\,d\Gamma = \sum_m A_{nm} F_{nm}$$

Discretized mass- and energy balance equation:

$$\frac{dM_n^{\kappa}}{dt} = \frac{1}{V_n}\sum_m A_{nm} F_{nm}^{\kappa} + q_n^{\kappa}$$

Discretized flow terms:

$$F_{\beta,nm} = -k_{nm}\left[\frac{k_{r\beta}\rho_{\beta}}{\mu_{\beta}}\right]_{nm}\left[\frac{P_{\beta,n}-P_{\beta,m}}{D_{nm}} - \rho_{\beta,nm} g_{nm}\right]$$

Implicit time discretization:

$$\frac{(M_n^{\kappa,k+1} - M_n^{\kappa,k})}{\Delta t} = \frac{1}{V_n}\cdot\left(\sum_m A_{nm}\cdot F_{nm}^{\kappa,k+1} + V_n \cdot q_n^{\kappa,k+1}\right)$$

Residual equations:

$$R_n^{\kappa,k+1} = \left(M_n^{\kappa.k+1} - M_n^{\kappa.k}\right) - \frac{\Delta t}{V_n}\left(\sum_m A_{nm} F_{nm}^{\kappa,k+1} + V_n \cdot q_n^{\kappa,k+1}\right) \equiv 0$$

Linearization and Newton-Raphson iterations:

$$-\sum_i \left.\frac{\partial R_n^{\kappa,k+1}}{\partial x_i}\right|_p \left(x_{i,p+1} - x_{i,p}\right) = R_n^{\kappa,k+1}(x_{i,p})$$

5. Fluid Property Modules

Module	Capabilities
EOS1	water, water with tracer
EOS2	water, CO_2
EOS3	water, air
EOS4	water, air, with vapor pressure lowering
EOS5	water, hydrogen
EOS7	water, brine, air
EOS7R	water, brine, air, parent-daughter radionuclides
EOS8	water, “dead” oil, non-condensible gas
EOS9	variably-saturated isothermal flow according to Richards’ equation
T2VOC	three-phase flow of ater, air, volatile organic compound (VOC)
EWASG	water, salt (NaCl), non-condensible gas (includes precipitation and dissolution, with porosity and permeability change; optional treatment of vapor pressure lowering effects)

6. Primary Variables

The thermophysical properties of fluid mixtures needed for assembling the governing mass- and energy-balance equations are provided by "equation-of-state" (EOS) modules. The flow module of TOUGH2 is coded in general fashion to set up and solve mass balances for an arbitrary number of NK components that are distributed among NPH phases. As explained in more detail in the description of individual EOS modules, below, fluid phase conditions are recognized from the numerical values of the primary variables. Besides providing values for all secondary (thermophysical) parameters as functions of the primary thermodynamic variables, i.e., assigning all data in the PAR-array, the EOS module fulfills three additional important functions:

(i) the phase conditions pertaining to a given set of primary variables are identified for all volume elements (grid blocks);

(ii) the appearance or disappearance of phases is recognized as primary variables change during the Newton-Raphson iteration process;

(iii) primary variables are switched and properly re-initialized in response to a change of phase.

The following tables summarize the primary variables for the most widely used equation-of-state (EOS) modules.

EOS3

Components	# 1: water # 2: air
Parameter choices	
(NK, NEQ, NPH, NB) =	(2, 3, 2, 6) water and air, nonisothermal (default) (2, 2, 2, 6) water and air, isothermal molecular diffusion can be modeled by setting NB = 8
Primary Variables *	
single-phase conditions	
(P, X, T) - (pressure, air mass fraction, temperature)	
two-phase conditions	
(P_g, S_g + 10, T) - (gas phase pressure, gas saturation plus 10, temperature)	

* By setting MOP(19) = 1, initialization can be made with TOUGH-style variables (P, T, X) for single-phase, (P_g, S_g, T) for two-phase.

EOS9

Components	# 1: water
Parameter choices (NK, NEQ, NPH, NB) =	(1, 1, 1, 6) water, isothermal
Primary Variables *† saturated conditions (P_{liq}) - (water pressure: $P_{liq} \geq P_{gas}$) unsaturated conditions (S_{liq}) - (water saturation: $0 < S_{liq} < 1$)	

EOS7R

Components	# 1: water # 2: brine # 3: Rn1 (radionuclide 1; "parent") # 4: Rn2 (radionuclide 2; "daughter") # 5: air (optional)†
Parameter choices (NK, NEQ, NPH, NB)§ =	(5, 5, 2, 8) water, brine, Rn1, Rn2, air, isothermal (default) (5, 6, 2, 8) water, brine, Rn1, Rn2, air, nonisothermal (4, 4, 2, 8) water, brine, Rn1, Rn2, no air, isothermal† (4, 5, 2, 8) water, brine, Rn1, Rn2, no air, nonisothermal†
molecular diffusion can be suppressed by setting NB = 6	
Primary Variables§ single-phase conditions (P, X_b, X_{Rn1}, X_{Rn2}, X_{air}, T) - (pressure, brine mass fraction, mass fraction of Rn1, mass fraction of Rn2, air mass fraction, temperature) two-phase conditions (P, X_b, X_{Rn1}, X_{Rn2}, S+10, T) - (gas phase pressure, brine mass fraction, mass fraction of Rn1, mass fraction of Rn2, gas saturation plus ten, temperature)*	

† the no air option (NK = 4) may only be used for problems with single-phase liquid conditions throughout

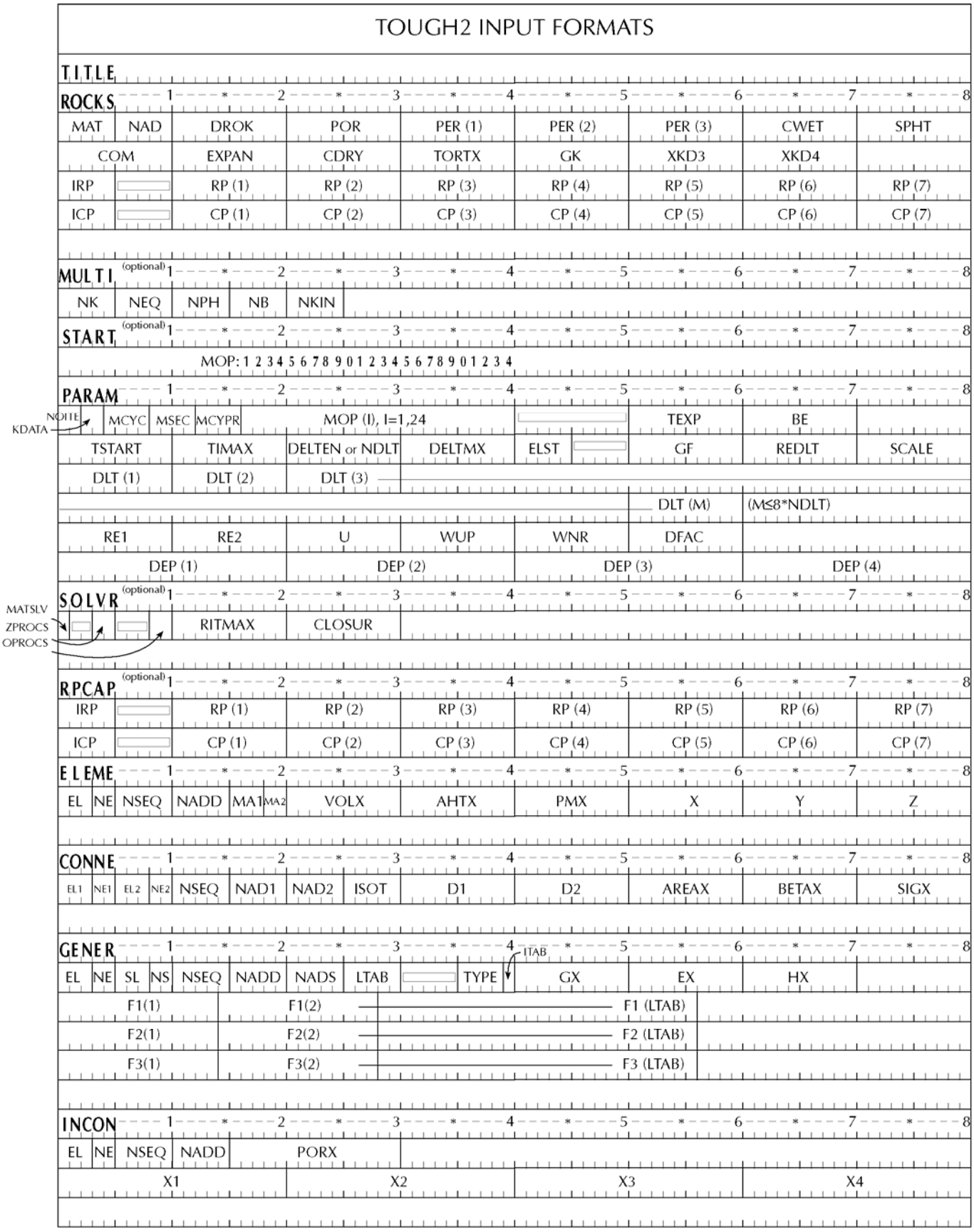
§ parameter NKIN following NB may optionally be set to NKIN = NK-2, in which case radionuclide mass fractions will be omitted, and initialization will be made from only four EOS7-style variables; radionuclide mass fractions will be initialized as zero;

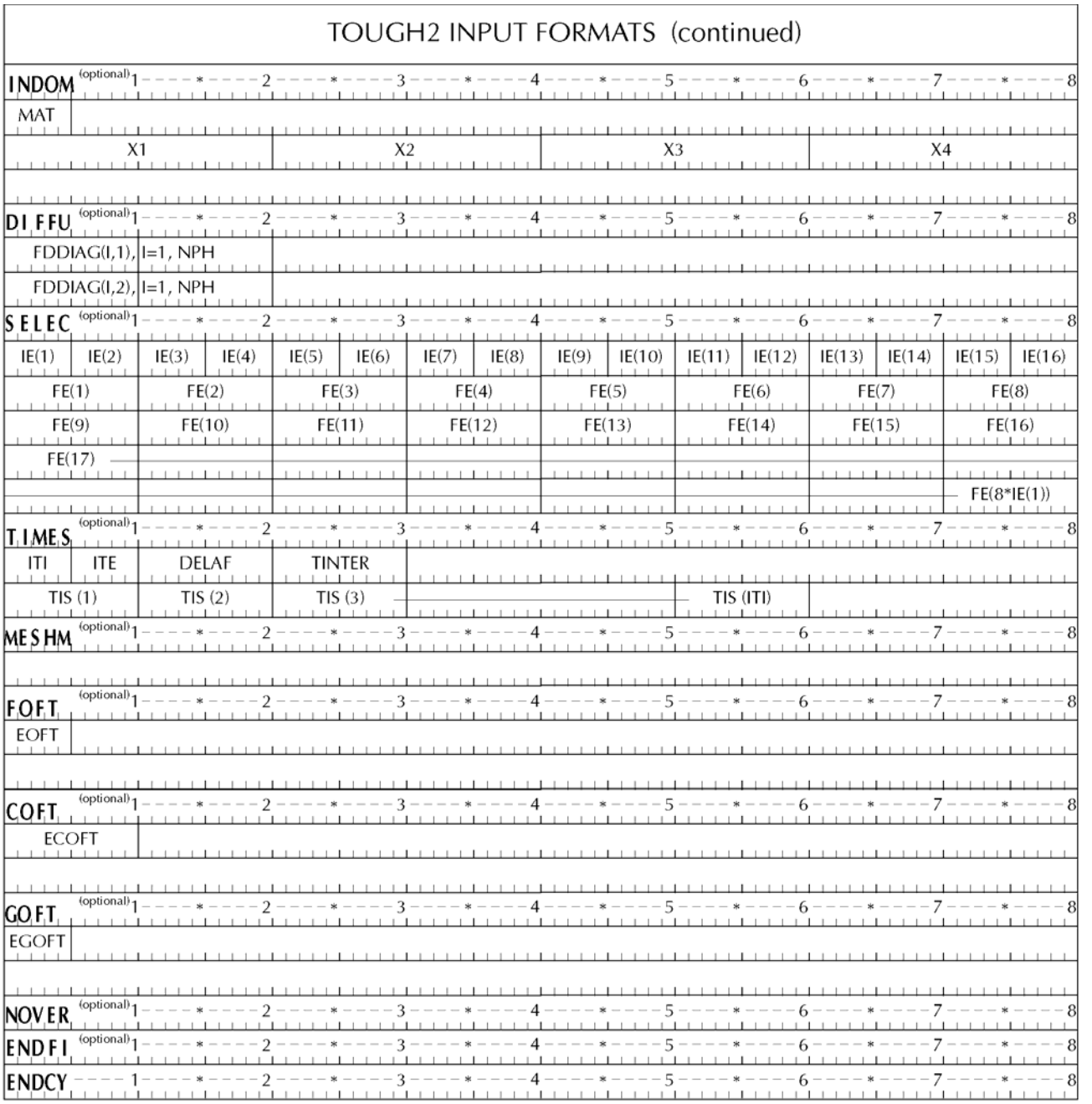
* in two-phase conditions, X_{Rn1} and X_{Rn2} are mass fractions in the aqueous phase

7. Data Blocks and Input Formats

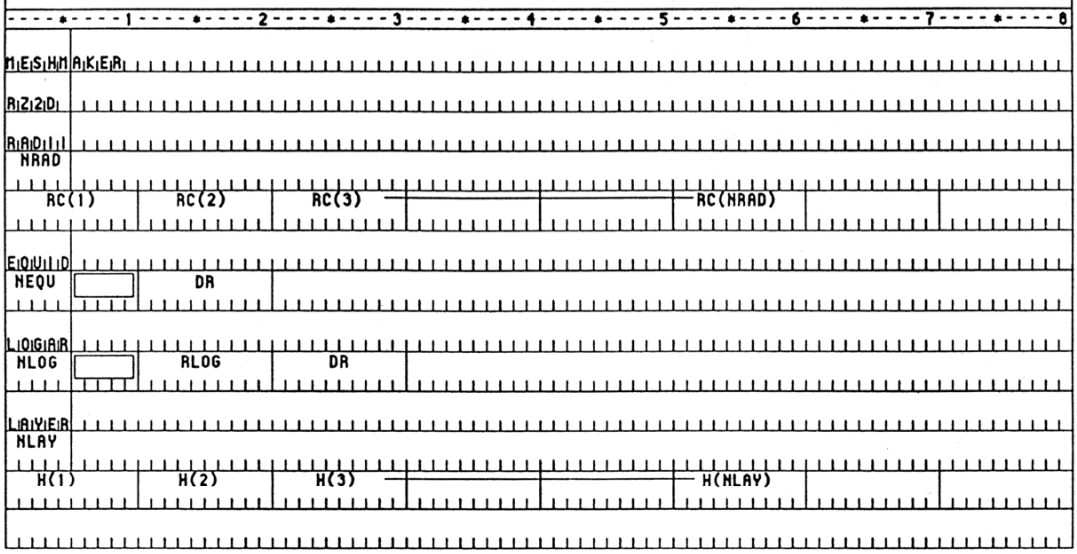
Keyword	Function
TITLE (first record)	one data record (single line) with a title for the simulation problem
MESHM	optional; parameters for internal grid generation through MESHMaker
ROCKS	hydrogeologic parameters for various reservoir domains
MULTI	optional; specifies number of fluid components and balance equations per grid block; applicable only for certain fluid property (EOS) modules
SELEC	used with certain EOS-modules to supply thermophysical property data
START	optional; one data record for more flexible initialization
PARAM	computational parameters; time stepping and convergence parameters; program options
DIFFU	diffusivities of mass components
FOFT	optional; specifies grid blocks for which time series data are desired
COFT	optional; specifies connections for which time series data are desired
GOFT	optional; specifies sinks/sources for which time series data are desired
RPCAP	optional; parameters for relative permeability and capillary pressure functions
TIMES	optional; specification of times for generating printout
*ELEME	list of grid blocks (volume elements)
*CONNE	list of flow connections between grid blocks
*GENER	optional; list of mass or heat sinks and sources
INDOM	optional; list of initial conditions for specific reservoir domains
*INCON	optional; list of initial conditions for specific grid blocks
NOVER (optional)	optional; if present, suppresses printout of version numbers and dates of the program units executed in a TOUGH2 run
ENDCY (last record)	one record to close the TOUGH2 input file and initiate the simulation
ENDFI	alternative to “ENDCY” for closing a TOUGH2 input file; will cause flow simulation to be skipped; useful if only mesh generation is desired

§ Blocks labeled with a star * can be provided as separate disk files, in which case they would be omitted from the INPUT file.

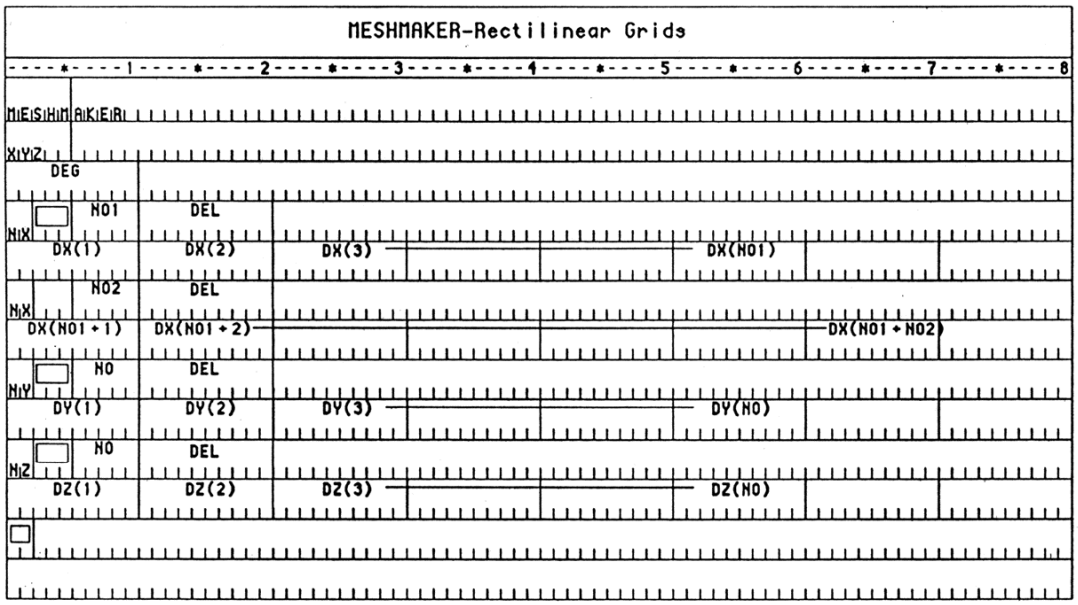

TOUGH2 INPUT FORMATS
TITLE
ROCKS
MAT NAD DROK POR PER (1) PER (2) PER (3) CWET SPHT
COM EXPAN CDRY TORTX GK XKD3 XKD4
IRP RP (1) RP (2) RP (3) RP (4) RP (5) RP (6) RP (7)
ICP CP (1) CP (2) CP (3) CP (4) CP (5) CP (6) CP (7)
MULTI (optional)
NK NEQ NPH NB NKIN
START (optional)
MOP: 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4
PARAM
NOITE
KDATA
MCYC MSEC MCYPR MOP (I), I=1,24 TEXP BE
TSTART TIMAX DELTEN or NDLT DELTMX ELST GF REDLT SCALE
DLT (1) DLT (2) DLT (3)
DLT (M) (M≤8*NDLT)
RE1 RE2 U WUP WNR DFAC
DEP (1) DEP (2) DEP (3) DEP (4)
SOLVR (optional)
MATSLV
ZPROCS
OPROCS
RITMAX CLOSUR
RPCAP (optional)
IRP RP (1) RP (2) RP (3) RP (4) RP (5) RP (6) RP (7)
ICP CP (1) CP (2) CP (3) CP (4) CP (5) CP (6) CP (7)
ELEME
EL NE NSEQ NADD MA1 MA2 VOLX AHTX PMX X Y Z
CONNE
EL1 NE1 EL2 NE2 NSEQ NAD1 NAD2 ISOT D1 D2 AREAX BETAX SIGX
GENER
LTAB
EL NE SL NS NSEQ NADD NADS LTAB TYPE GX EX HX
F1(1) F1(2) F1 (LTAB)
F2(1) F2(2) F2 (LTAB)
F3(1) F3(2) F3 (LTAB)
INCON
EL NE NSEQ NADD PORX
X1 X2 X3 X4


TOUGH2 INPUT FORMATS (continued)
INDOM (optional)
MAT
X1
X2
X3
X4
DIFFU (optional)
FDDIAG(I,1), I=1, NPH
FDDIAG(I,2), I=1, NPH
SELEC (optional)
IE(1) IE(2) IE(3) IE(4) IE(5) IE(6) IE(7) IE(8) IE(9) IE(10) IE(11) IE(12) IE(13) IE(14) IE(15) IE(16)
FE(1) FE(2) FE(3) FE(4) FE(5) FE(6) FE(7) FE(8)
FE(9) FE(10) FE(11) FE(12) FE(13) FE(14) FE(15) FE(16)
FE(17)
FE(8*IE(1))
TIMES (optional)
ITI ITE DELAF TINTER
TIS (1) TIS (2) TIS (3) TIS (ITI)
MESHM (optional)
FOFT (optional)
EOFT
COFT (optional)
ECOFT
GOFT (optional)
EGOFT
NOVER (optional)
ENDFI (optional)
ENDCY

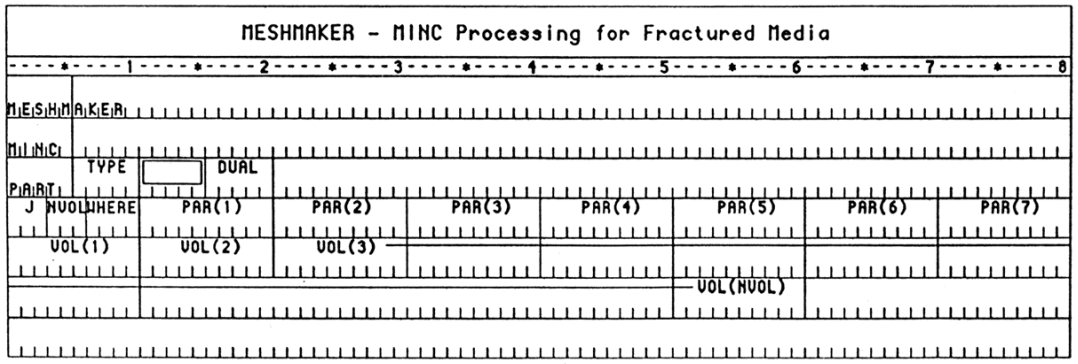
MESHMAKER - Two-dimensional R-Z Grids



MESHMAKER-Rectilinear Grids



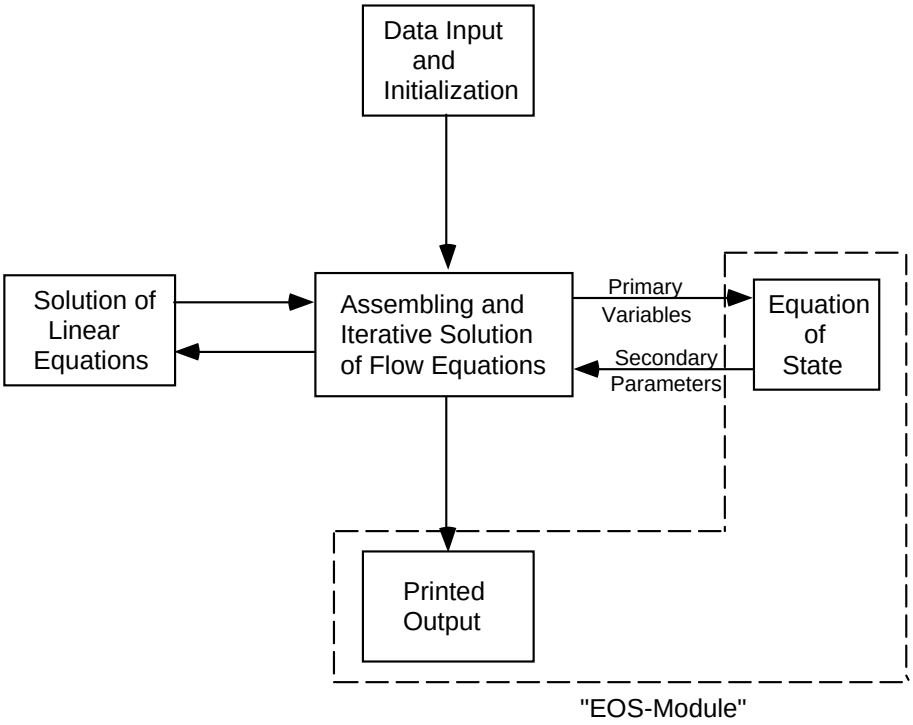
MESHMAKER - MINC Processing for Fractured Media



8. Disk Files

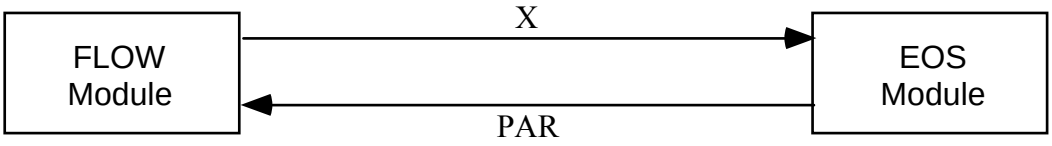
File	Use
MESH	written in subroutine INPUT from ELEME and CONNE data, or in module MESHMAKER from mesh specification data; read in RFILE to initialize all geometry data arrays used to define the discretized flow problem
GENER	written in subroutine INPUT from GENER data; read in RFILE to define nature, strength, and time-dependence of sinks and sources
INCON	written in subroutine INPUT from INCON data; read in RFILE to provide a complete specification of thermodynamic conditions
SAVE	written in subroutine WRIFI to record thermodynamic conditions at the end of a TOUGH2 simulation run; compatible with formats of file or data block INCON for initializing a continuation run
MINC	written in module MESHMAKER with MESH-compatible specifications, to provide all geometry data for a fractured-porous medium mesh (double porosity, dual permeability, etc.); read (optionally) in subroutine RFILE to initialize geometry data for a fractured-porous system
LINEQ	written during linear equation solution, to provide informative messages on linear equation solution
TABLE	written in CYCIT to record coefficients of semi-analytical heat exchange at the end of a TOUGH2 simulation run read (optionally) in subroutine QLOSS to initialize heat exchange coefficients in a continuation run
FOFT	written in FGTAB to provide time series data for elements for plotting
COFT	written in FGTAB to provide time series data for connections for plotting
GOFT	written in FGTAB to provide time series data for sinks/sources for plotting
VERS	written in all TOUGH2 program units with informational message on version number, date, and function; read in main program and printed to default OUTPUT at the conclusion of a TOUGH2 simulation run; printing of version information is suppressed when keyword NOVER is present in INPUT file

9. TOUGH2 Program Structure



subroutine(s)	function
TOUGH2 (main program)	executive routine; define problem-size dependent arrays, generate internal documentation
INPUT, RFILE (and satellite routines, including MESHMaker)	problem initialization
CYCIT	executive routine for time stepping
EOS (and satellites)	thermophysical properties and phase diagnostics
MULTI (and satellites)	assembly of mass and energy balance equations
QU (and satellites)	sink and source terms
LINEQ (and satellites)	solution of linear equations, provisional updating of thermodynamic variables
CONVER	conclusion of converged time steps, updating of thermodynamic variables and iteration counters
WRIFI, OUT, BALLA	output of results

10. TOUGH2 Program Structure



Volume Element	Primary Variables
# 1	X(1) • • • X(NK1)
# 2	X(NK1+1) • • • X(2*NK1)
	• • • • •
# N	X(NLOC+1) • • • X(NLOC+NK1)
	• • • • •
# NEL	• • • X(NEL*NK1)

Volume Element	SECONDARY PARAMETERS		gas phase	aqueous phase
# 1	phase saturation	S	PAR(1)	PAR(NBK+1)
	relative permeability	k_r	PAR(2)	•
	viscosity	μ	PAR(3)	•
	density	ρ	PAR(4)	•
	specific enthalpy	h	PAR(5)	•
	capillary pressure	P_c	PAR(6)	•
	diffusion factor 1(§)	a	PAR(7)	•
	diffusion factor 2(&)	b	PAR(8)	•
	mass fractions			
	component 1	X^1	PAR(NB+1)	•
	•	•	•	•
	component NK	X^{NK}	PAR(NB+NK)	PAR(2*NBK)
	temperature	T	PAR(NSEC-1)	
	(miscellaneous)		PAR(NSEC)	
	(second set of secondary parameters: X(1) incremented)			
			PAR(NSEC+1)	
			•	
			•	
			PAR(2*NSEC)	
			•	
			•	
			•	
			PAR((NEQ+1)*NSEC)	
# 2 • • • # NEL	• • • • •			

(§) $a=\phi\tau_0\tau_\beta\rho_\beta$ (&) $b=d_\beta^\kappa(T,P)/d_\beta^\kappa(T_0,P_0)$

11. SELEC Block in EOS7R

SELEC keyword to introduce a data block with reference brine, geometry, dispersion, and radioactive decay data.

Record SELEC.1

Format(16I5)
IE(1)

IE(1) set equal to 6 to read six additional records with data for brine and radionuclides, and for hydrodynamic dispersion.

Record SELEC.2

Format(3E10.4)
P_0, T_0, ρ_b

P_0 reference pressure, Pa.

T_0 reference temperature, ˚C.

ρ_b brine density at (P_0, T_0), kg/m^3.

If any of these parameters is entered as zero, default values of P_0 = 1 bar, T_0 = 25 ˚C, ρ_b = 1185.1 kg/m^3 will be used. For $P_0 < 0$, brine properties will be assumed identical to water.

Record SELEC.3

Format(3E10.4)
v_1, v_2, v_3

coefficients for salinity correction of aqueous phase viscosity, see Eq. (9) in TOUGH2 User's Guide.

Record SELEC.4

Format(2E10.4)
ALPHAT, ALPHAL

Used in T2DM only; leave blank.

Record SELEC.5

Format(6E10.4)
FDDIAG(NP,NK), NK=1,2,5; NP=1,2

FDDIAG(NP,NK) molecular diffusivities in units of m^2/s; first the three gas phase diffusivities for water, brine, and air; then the three aqueous phase diffusivities for water, brine, and air. See Section D.4 in Appendix D of TOUGH2 User's Guide for user options. If a data block DIFFU is present, it will override the diffusivity specifications made in SELEC.

Record SELEC.6

Format(7E10.4)
XHALF(3), XMW(3), (FDDIAG(NP,NK), NK=3; NP=1,2), blank, blank, HCRN1

XHALF(3) half life of parent radionuclide (Rn1, component 3), seconds.

XMW(3) molecular weight of Rn1, kg/mol.

FDDIAG(NP,3) molecular diffusivity of Rn1 in the gas phase in m^2/s; followed by molecular diffusivity of Rn1 in the aqueous phase. See Section D.4 in Appendix D of TOUGH2 User's Guide for user options. If a data block DIFFU is present, it will override the diffusivity specifications made in SELEC.

HCRN1 inverse Henry's constant $(K_h)^{-1}$ (see Eq. 6) for parent radionuclide Rn1, Pa^{-1}. (The inverse Henry's constant can be thought of as an aqueous phase solubility.)

Record SELEC.7

Format(7E10.4)
XHALF(4), XMW(4), (FDDIAG(NP,NK), NK=4; NP=1,2), blank, blank, HCRN2

XHALF(4) half life of daughter radionuclide (Rn2, component 4), seconds.

XMW(4) molecular weight of Rn2, kg/mol.

FDDIAG(NP,4) molecular diffusivity of Rn2 in the gas phase in m^2/s; followed by molecular diffusivity of Rn2 in the aqueous phase. See Section D.4 in Appendix D of TOUGH 2 User's Guide for user options. If a data block DIFFU is present, it will override the diffusivity specifications made in SELEC.

HCRN2 inverse Henry's constant $(K_h)^{-1}$ (see Eq. 6) for daughter radionuclide Rn2, Pa^{-1}. (The inverse Henry's constant can be thought of as an aqueous phase solubility.)

12. Code Installation and Execution

Compiling TOUGH2

- To create TOUGH2 executables, the following files must be compiled and linked:

  t2cg22.f, eosX.f, t2f.f, meshm.f, t2solv.f, ma28.f

  where X in eosX.f represents the desired equation of state (EOS) module number. The files t2cg22.f and T2 (which must be present for compiling) should be downloaded from: http://www-esd.lbl.gov/TOUGH2/T2V2bf.html

- When the EOS7R, EOS9 or T2VOC module is desired, the following change(s) to the file t2f.f must be made before compiling: rename subroutine MULTI to MULTIx (for T2VOC, also rename subroutines RELP and PCAP to RELPx and PCAPx).

- The machine-dependent timing function in subroutine SECOND of file t2cg22.f may also need to be replaced for some compilers. See stubs provided at: http://www-esd.lbl.gov/TOUGH2/PROGRAMS/FREEPROGRAMS.html

- TOUGH2 is written in single precision, but requires 64-bit arithmetic. Therefore, an appropriate compiler switch must be engaged (e.g., -dbl for Lahey compiler, /real_size:64 for Compaq Visual Fortran compiler; -r8 for most UNIX Fortran compilers). Also, specify that integers be stored as 8 byte words (e.g., using -i8) to avoid alignment problems that may arise in some COMMON blocks where floating point and integer variables are mixed.

- In addition, make sure to specify that the source code is in fixed format (72 columns) and that array bounds are NOT to be checked during run-time. See file read.me for additional information.

- The directory containing the executable files (e.g., C:\TOUGH2) should be added to the command search path. To accomplish this on a PC running Windows XP, click on Start → Settings → Control Panel → Advanced → Environmental Variables, and then edit the PATH listed in the System Variables window, so that the directory of the executables is listed at the end (e.g., add ";C:\TOUGH2").

- The TOUGH2 executables needed for this course are already provided (named t2_X.exe, with X again referring to the number of the EOS module).

Running TOUGH2 on a PC

- Open a command prompt (MS-DOS window) by clicking on:

  Start → Programs → Accessories→ Command Prompt

- Go to the directory where the input file is located by typing "cd" followed by a space and the directory name, for example:

  cd c:\tough2\problem1

- Run the code by typing:

  t2_X < *input_file_name* > *output_file_name*

  Additional output file are created by TOUGH2 creates as well, such as MESH, INCON, GENER, LINEQ, TABLE, VERS and FOFT.

- For example, an input file named sam1.txt could be run with the EOS9 module by typing:

  t2_9 < sam1.txt > sam1.out

- Visualization of TOUGH2 results requires post-processing of the TOUGH2 output files and the use of an external visualization package (such as EXCEL, TECPLOT, or MATLAB).

Tips for Editing TOUGH input/output files on a PC (using Notepad)

- Many of the output files do not have file extensions (e.g., .txt), which would be useful on a PC since they indicate which program needs to be used for opening the file. When opening TOUGH-related files not containing a file extension (like SAVE or MESH), double click on the file and an "Open with" window comes up. Choose Notepad for text editing. If you type "n" after the "Open with" window appears, the Notepad program should be immediately be highlighted and ready to open (by typing the enter key).

- Never use the TAB key within input file (it causes illegal characters to be inserted)

- Text editing of TOUGH input files commonly involves the copying and pasting blocks of data within files or from old to new files.

    - One way (the slow way) to copy all the contents of a file (such as from a SAVE file) and paste them somewhere else (such as in a new input file) is as follows: from drop-down menu choose Edit→ Select All, followed by Edit → Copy. Then go to location where you wish to insert the text, and from the drop-down menu choose Edit → Paste).

    - A faster way to accomplish the same is to simply type Ctrl + a, and then Ctrl + c to select and copy everything in a file; then go to the desired new file location and type Ctrl + v to paste what was just copied.

    - For highlighting only portions of text within a file to be copied, type Shift + Up-arrow or Shift + Down-arrow (or use Shift + Page Up or Page Down keys to highlight whole pages at a time). After highlighting desired portion of text, type Ctrl + c to copy it, etc.