FGC-18-02
LBNL-7016E

Enhancements to the TOUGH2 Simulator Integrated in iTOUGH2

Stefan Finsterle

Finsterle GeoConsulting, LLC
Kensington, CA 94708

Energy Geosciences Division
Lawrence Berkeley National Laboratory
University of California
Berkeley, CA 94720

revised and updated

April 21, 2024
iTOUGH2 V7.2.4

This work was supported, in part, by the Laboratory of Hydraulics, Hydrology, and Glaciology (VAW) of the Swiss Federal Institute of Technology (ETH), Zürich, Switzerland; the U.S. Dept. of Energy under Contract No. DE-AC02-05CH11231; the Swiss National Cooperative for the Disposal of Radioactive Waste (Nagra), Wettingen, Switzerland; Warm Arctic ehf, Kópavogur, Ísland; Vatnaskil, Reykjavík, Ísland; Deep Isolation, Berkeley, California; and Geologica, San Francisco, California.

DISCLAIMER
FINSTERLE GEOCONSULTING, LLC

This report may concern work in progress that may have had limited review. While this document is believed to contain correct information, neither Finsterle GeoConsulting nor any Sponsor of this work make any warranty of any kind, either expressed, implied, or statutory, or assumes any legal responsibility for the accuracy, completeness, or usefulness of this information, or represents that its use would not infringe privately owned rights. Any use which a third party makes of the information presented in this work, or any reliance on or decisions to be made based on it, are the responsibility of such third parties.

DISCLAIMER
LAWRENCE BERKELEY
NATIONAL LABORATORY

This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor The Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or The Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof, or The Regents of the University of California.

Ernest Orlando Lawrence Berkeley National Laboratory
is an equal opportunity employer.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 Introduction

iTOUGH2 is a program for parameter estimation, sensitivity analysis, and uncertainty propagation analysis. It is based on the TOUGH2 simulator for non-isothermal multiphase, multicomponent flow and transport in fractured and porous media. The manuals describing iTOUGH2's capabilities are listed in Table 1.

The core of iTOUGH2 contains slightly modified versions of TOUGH2 modules as documented in *Pruess et al.* [2012] and other reports (see Table 2 below).

Most code modifications are editorial and do not affect the simulation results. Thus, standard TOUGH2 input files can be used in iTOUGH2, and consistent results are obtained if iTOUGH2 is run in forward mode. However, a number of modifications have been made to the version of TOUGH2 that is implemented in iTOUGH2. They enhance the functionality, flexibility, and ease-of-use of the forward simulator. Some of these enhancements are described in this report; more significant developments are documented in separate reports. Table 2 lists manuals describing general aspects of TOUGH2 modules or specific features of the forward simulator. (Note that not all capabilities mentioned in Table 2 are publicly released.)

The key to a successful application of iTOUGH2 is (i) a good understanding of multiphase flow processes, (ii) the ability to conceptualize the given flow and transport problem and to develop a corresponding TOUGH2 model, (iii) detailed knowledge about the data used for calibration, (iv) an understanding of parameter estimation theory and the correct interpretation of inverse modeling results, (v) proficiency in using iTOUGH2 options. This report is mainly concerned with issues (i) and (ii).

Table 1. Documentation of iTOUGH2

Module / Feature / Capability	**Reference**	**Source Files**
iTOUGH2 user's guide; theoretical background	*Finsterle*, 2015a	-
iTOUGH2 command reference	*Finsterle*, 2015b	-
iTOUGH2 sample problems	*Finsterle*, 2015c	-
Parallel execution of TOUGH2 runs	*Finsterle*, 1998	it2pvm.f
Geostatistical simulations	*Finsterle and Kowalsky*, 2007	it2gslib.f
Link to external models using PEST protocol	*Finsterle*, 2010	it2pest.f mio.f90
Global sensitivity and data-worth analysis	*Wainwright and Finsterle*, 2015	it2sa.f

Table 2. Documentation of TOUGH2 and Related Modules

Module / Feature / Capability	**Reference**	**Source Files**
Main TOUGH2 reference; general description of code capabilities, mathematical model, numerical scheme, input formats, and sample problems	*Pruess et al.*, 2012	eos1.f eos2.f eos3.f eos4.f eos5.f eos7.f eos7r.f eos8.f eos9.f eosewasg.f
Three-phase (gas, aqueous, NAPL), three-component (water, air, VOC) module T2VOC	*Falta et al.*, 1995	eost2voc.f
Three-phase (gas, aqueous, NAPL), multi-component (water, NCGs, VOCs) module TMVOC	*Pruess and Battistelli*, 2002	eostmvoc.f
Three-phase (CO_2-rich phase, aqueous, solid), three-component (water, CO_2, NaCl) module ECO2N	*Pruess*, 2005	eco2n.f
Four-phase (gaseous and liquid CO_2, aqueous, solid), three-component (water, CO_2, NaCl) module ECO2M	*Pruess*, 2011	eco2m.f
Two-phase (gas, aqueous), five-component (water, brine, CO_2 or N_2 (or air), tracer, CH_4) module EOS7C	*Oldenburg et al.*, 2004	eos7c.f eos7ca.f
Two-phase (steam, liquid), two-component ($water_1$, $water_2$) supercritical module EOS1sc	*Magnúsdóttir and Finsterle*, 2015	eos1sc.f
Two-phase (gas, liquid), three-component (water, air, miscible, solidifying gel) module EOS11	*Finsterle et al.*, 1994	eosgel.f
Non-Darcy flow based on Forchheimer equation	*Finsterle and Witherspoon*, 2001	t2f.f
Iterative linear equation solvers	*Moridis and Pruess*, 1997	t2cg22.f t2solv.f
Hysteretic relative permeability and capillary pressure functions	*Doughty*, 2013	it2hyster.f
Geostatistical simulations	*Finsterle and Kowalsky*, 2007	it2gslib.f
Two-phase (steam, liquid) flow of water and multiple tracers, module EOS1nT	*Finsterle*, 2021	eos1nt.f
Two-phase (gas, aqueous), five-component (water, brine, NCG, tracer, base gas) module EOS7CH	*Oldenburg and Finsterle, 2023*	eos7ch.f

2 Program Options and Control Parameters

2.1 More Options (Block MOMOP)

Table 3 describes additional options invoked by integer flags on a line following keyword MOMOP.

Table 3. More Options

MOP2	Value	Description
1		*Minimum number of Newton-Raphson iterations*
	0, 1	Allow convergence in a single Newton-Raphson iteration
	2	Perform at least two iterations; primary variables are always updated.
	3	Allow convergence in a single Newton-Raphson iteration for negative simulation times but require at least two for positive times; useful for steady-state followed by transient simulations.
	4	Allow convergence in a single Newton-Raphson iteration for positive simulation times but require at least two for negative times.
2		*Length of element names (default: 5 characters)*
	5–9	Format of blocks ELEME, CONNE, INCON, and GENER change depending on element-name length as follows:
		ELEME
	5	(A3,I2,I5,I5,A2,A3,6E10.4)
	6	(A3,I3,I5,I4,A2,A3,6E10.4)
	7	(A3,I4,I4,I4,A2,A3,6E10.4)
	8	(A3,I5,I4,I3,A2,A3,6E10.4)
	9	(A3,I6,I3,I3,A2,A3,6E10.4)
		CONNE
	5	(2(A3,I2),I5,2I5,I5,4E10.4)
	6	(2(A3,I3),I5,2I4,I5,4E10.4)
	7	(2(A3,I4),I5,2I3,I5,4E10.4)
	8	(2(A3,I5),I3,2I3,I5,4E10.4)
	9	(2(A3,I6),I3,2I2,I5,4E10.4)
		INCON
	5	(A3,I2,I5,I5,E15.8,4E12.4)
	6	(A3,I3,I5,I4,E15.8,4E12.4)
	7	(A3,I4,I4,I4,E15.8,4E12.4)
	8	(A3,I5,I4,I3,E15.8,4E12.4)
	9	(A3,I6,I3,I3,E15.8,4E12.4)
		GENER
	5	(A3,I2,A3,I2,I5,2I5,I5,5X,A4,A1,3E10.4)
	6	(A3,I3,A3,I2,I6,2I4,I5,5X,A4,A1,3E10.4)
	7	(A3,I4,A3,I2,I5,2I4,I5,5X,A4,A1,3E10.4)
	8	(A3,I5,A3,I2,I4,2I4,I5,5X,A4,A1,3E10.4)
	9	(A3,I6,A3,I2,I5,2I3,I5,5X,A4,A1,3E10.4)

3		*Honoring generation times in block* GENER
	0	Generation times ignored
	≥1	Time steps adjusted to match generation times
4		*Vapor pressure lowering*
	0	No vapor pressure lowering
	1	Vapor pressure lowering according to Kelvin's equation, $P_v = P_{sat}$ exp $[P_c M_w/(\rho_l RT)]$
	2	Reduces vapor pressure for $S_l < 0.02$ to zero at $S_l = 0.0$ to prevent liquid disappearance by evaporation
	3	Combines options 1 and 2
5		*Active Fracture Model (see Section 8)*
	0	Active Fracture Model applied to liquid phase only
	≥1	Active Fracture Model applied to all phases
6		*Leverett scaling of capillary pressure if element-specific permeabilities and/or porosities are specified, or if the respective parameter is adjustable and updated*
	0	No Leverett scaling
	1	Rescale capillary pressure: $P_c = P_{c,ref}\sqrt{k_{ref}/k}$
	2	Rescale capillary pressure: $P_c = P_{c,ref}\sqrt{(k_{ref} \cdot \phi_i)/(k \cdot \phi_{ref})}$
7		*Zero nodal distance*
	0	Take absolute permeability from other element
	≥1	Take absolute and relative permeability from other element
8		*Version of sink/source subroutine QU*
	1	Take QU V1
	2	Take QU V2 (default)
9		*Time stepping after time-step reduction to honor printout time*
	0	Continue with time step used before forced time-step reduction
	≥1	Continue with time step imposed by forced time-step reduction
10		*Writing SAVE file*
	0	Write SAVE file only at the end of a forward run
	≥1	Write SAVE file after each printout time
11		*Water properties*
	0	International Formulation Committee (1967)
	1	IAPWS-IF97
	2	EOS1sc only: IAPWS-IF97 for T<800 °C IAPWS-95 for T≥800 °C
12		*Enthalpy of liquid water*
	0	Potential energy not included in enthalpy of liquid water
	≥1	Potential energy included in enthalpy of all phases [*Stauffer et al.*, 2014]

13		*Adjustment of Newton-Raphson increment weighting*
	0	No adjustment
	≥1	Reduce *WNR* by *MOP2*(13) percent if Newton-Raphson iterations oscillate and time step is reduced because *ITER* = *NOITE*
14		*Print input file to the end of output file*
	0	Do not reprint input files
	1	Print TOUGH2 input file to the end of TOUGH2 output file
	2	Print iTOUGH2 input file to the end of iTOUGH2 output file
	3	Print TOUGH2 and iTOUGH2 input files to the end of respective output files
15		*Porosity used for calculation of rock energy content*
	0	Use porosity of block ROCKS; this assumes that the porosities provided in block INCON were the result of a pore compressibility/expansivity calculation; the "original" porosity from block ROCKS is used to compensate for equivalent rock-grain density changes.
	≥1	Use porosity from block INCON; this assumes that these porosities were not the result from a pore compressibility/expansivity calculation; changes in rock-grain density due to pore compressibility/expansivity are not compensated.
16		*Porosity-permeability relationships for heterogeneous media*
	0	No deterministic correlation
	1	Material-specific empirical correlations (see subroutine PER2POR)
17		*Invokes wellbore simulator FloWell (Guðmundsdóttir et al., 2018)*
	0	Do not run FloWell simulator
	1	Run FloWell for each calibration and printout time
	2	Run FloWell for each time step
18		*Porosity update by ROCMECH*
	0	Do not update porosity
	1	Apply porosity correction
19		*Primary variable update (J. O'Sullivan)*
	0	Update all primary variables
	≥1	Do not update primary variable if scaled increment is smaller than $10^{(-MOP2(19)-5)}$
20		*Reading anisotropic permeability modifiers in block* ELEME (*see also block* MAPPI *(Section 5.2) for reading and mapping permeabilities provided on external files)*
	0	Read isotropic permeability modifiers from columns 41–50
	1	Read anisotropic permeability modifiers from columns 81–110
	2	Read anisotropic permeability modifiers for *ISOT* = 1 from columns 41–50 and for *ISOT* = 2 and 3 from columns 91–110
21		*Honoring generation times in blocks* TIMBC *(see Section 6.1) and* LOAD *(see Section 11)*
	0	Time stepping ignores times specified in blocks TIMBC and LOAD
	≥1	Time steps adjusted to match times TIMBC and LOAD

22		*Format for reading coordinates in block* `ELEME`
	0	Read coordinates in format 3E10.4 from columns 51–80
	≥1	Read coordinates in format 3E20.14 from columns 51–110
23		*Makes choice of 5- or 9-point mesh when using* `MESHM`
	0	5-point mesh
	1	9-point mesh; use only with rotation angle `DEG` = 0 or 90
	2	9-point mesh; symmetry at `I` = 1
24		*Format (text or binary) of* MESH *input and output files*
	0	Input: text Output: text
	1	Input: binary Output: binary
	2	Input: text Output: binary
	3	Input: binary Output: text
25		*Check mesh for duplicate element names*
	0	Do not check the mesh
	1	Check the mesh
26		*Printout format (see block* `OUTPU`*, Section 3.1)*
	0	Traditional output format (add 3 to the following options to print in both traditional and CSV formats)
	1	CSV/TECPLOT/GNU format (types separated; times combined)
	2	CSV/TECPLOT/GNU format (types separated; times separated)
	3	CSV/TECPLOT/GNU format (types combined; times combined)
27		*Element naming in* `MESHM`
	0	Traditional element naming scheme
	1	Element name is equal to consecutive element number
28		*Scaling of capillary pressure by temperature-dependent surface tension*
	0	No scaling
	1	$P_c(T) = P_c(20) \cdot \left[\frac{\sigma(T)}{\sigma(20)}\right]$
29		*Phase transition window (for certain EOSs only)*
	0	Hair-trigger phase transition
	≥1	Finite-window phase transition; window size: $10^{-MOP2(29)}$
30		*Switch from hair-trigger to finite-window phase transition (only if* `MOP2(29) > 0`*)*
	0	Always use finite-window phase transition
	≥1	Switch from hair-trigger to finite-window phase transition if `ITER` ≥ `MOP2(30)`
31		*Reduce convergence criterion* `RE1` *for positive simulation times*
	0	Keep `RE1` constant
	≥1	Reduce `RE1` by `MOP2(31)` orders of magnitude when simulation time transitions from negative to positive (useful when running a steady-state simulation for negative times)

32		*Turn off production/injection wells if pressure goes below/above critical pressure specified in variable* HG *(A. Arnaldsson, J.-C. Berthet)*
	0	No pressure limits
	1	For production: Turn well off if pressure falls below *HG* (*HG* is in Columns 61–70) turn well on again if pressure raises above (*HG* + *GMOD*) (*GMOD* is in Columns 71–80) For injection: Turn well off if pressure rises above *HG*; turn well on again if pressure falls below (*HG* - *GMOD*)
	2	Turn on/off make-up wells (see Section 6.10)
	3	Permanently turn well off if pressure exceeds limit once; at the same time, turn on make-up wells (see Section 6.10)
	9	Switch to Dirichlet boundary condition after pressure limit is reached Stop simulation if pressure limit is reached
33		*Upstream weighting in the presence of small gradients (J. O'Sullivan)*
	0	No adjustment of WUP
	≥1	Set WUP to 0.5 if relative pressure gradient is less than $10^{(-MOP2(33))}$
34		*Convergence criterion based on maximum primary variable increment (see also MOP2(45) and MOP2(57)) (J. O'Sullivan)*
	0	Convergence criterion *RE3* not engaged
	≥1	Newton-Raphson iterations converge if maximum scaled primary variable increment is less than *RE3* = $10^{(-MOP2(34)-5)}$
35		*Apply pathway dilation and clay swelling/shrinking models*
	0	Do not apply dilation and clay swelling/shrinking models
	1	Apply tensile fracturing model [*Nagra*, 2004]; see Section 16
	2	Change permeability due to clay swelling/shrinking after *Xie et al.* [2007]; see Section 18
	3	Apply tensile fracturing model [*Nagra*, 2004] and permeability change due to clay swelling/shrinking [*Xie et al.*, 2007]; see Sections 16 and 18
	4	Apply gas pathway dilation model [*Horseman et al.*, 1996] using clay swelling/shrinking model by *Xie et al*. [2007]; see Sections 17 and 18
	5	Apply gas pathway dilation model [*Horseman et al.*, 1996] using power-law gas permeability factor, Eq. (17-2); see Section 17
36		*Strategies to recover from NaNs in linear solve*
	0	Do not apply NaN strategy
	1	Write and read SAVE file when a NaN appears during linear solve
	≥2	Write and read SAVE file when a NaN appears during linear solve, and perturb primary variables with random Gaussian noise with a standard deviation of $10^{(-MOP2(36)-5)}$
37		*Control of full printout (saves CPU time and disk space if running in parallel)*
	0	Print out according to option in block PARAM
	1	Do not print out full output if evaluating Jacobian
	2	Do not print out full output if running on a child process

38		*Initial temperature, T_0, and thermal diffusivity, D_t, of confining unit for semi-analytical heat exchange (for 1 ≤ MOP(15) ≤2)*
	0	T_0 and D_t uniform, taken from last element
	1	T_0 non-uniform, taken from adjacent element; and D_t uniform, taken from last element
	2	T_0 and D_t non-uniform, taken from adjacent element
40		*Set variable ZERO used to initialize primary variables after phase change*
	0	ZERO = 10^{-6}
	≥1	ZERO = $10^{-\text{MOP2(40)}}$
41		*Convergence criteria for overland flow (see Section 9)*
	0	Same as for subsurface flow
	≥1	Relaxed by factor $10^{\text{MOP2(41)}}$
42		*Initialize hydrostatic pressure profile (FE(5) = surface pressure [Pa]; FE(6) = fluid density [kg/m^3])*
	0	No initialization of pressure profile
	1	Initialize pressure profile except for elements in block INDOM
	2	Initialize pressure profile for all elements
43		*Initialize temperature profile (FE(7) = surface temperature [°C]; FE(8) = geothermal gradient [°C/m])*
	0	No initialization of temperature profile
	1	Initialize temperature profile except for elements in block INDOM
	2	Initialize temperature profile for all elements
44		*Prediction uncertainty calculation*
	0	Calculate prediction uncertainties (matrix $\mathbf{C}_{zz}$)
	1	Do not calculate prediction uncertainties (e.g., if *m* is very large)
45		*Exclude equation from evaluation of RE3 convergence criterion (see also MOP2(34) and MOP2(57))*
	0	All residuals included in RE3 criterion evaluation
	>0	Only include residuals from equations ≤MOP2(45) and heat equation in RE3 criterion evaluation
46		*Printout for semi-analytical heat exchange calculations (MOP(15) > 0)*
	0	No printout
	>0	Increasing amount of printout
47		*Temperature dependence of diffusion coefficient in liquid phase*
	0	No temperature dependence of diffusion coefficient
	1	Stokes-Einstein equation for temperature dependence, assuming T_{ref} = 25°C
48		*Minimum component mass fractions X_{min} in EOS1nT*
	0–8	$X_{min} = 10^{-(15+\text{MOP2(48)})}$
	9	X_{min} = 0.0

49		*Effective phase density for gravity term between element i and j*
	0	$\rho_e = (\rho_i + \rho_j)/2$
	1	$\rho_e = (S_i \cdot \rho_i + S_j \cdot \rho_j)/(S_i + S_j)$ (Croucher et al., 2020)
50		*Printing of all elements or connections in user-defined regions*
	0	No printing of elements or connections in regions
	≥1	Print elements or connections in regions
51		*Writing of generation regions in standard TOUGH2 format*
	0	No writing of file GENER_T2
	≥1	Convert generation block to TOUGH2 format
52		*Freezing capabilities (for certain EOS modules only)*
	0	No freezing capabilities
	1	Freezing capabilities
53		*Replace blanks with zeroes in element and sink/source names*
	0	No black substitution
	≥1	Replace blanks with zeroes
54		*Relative permeability functions in EOS8, T2VOC, and TMVOC*
	0	Use three-phase relative permeability functions
	≥1	Use two-phase relative permeability functions; set k_{ro} to zero
55		*Capillary pressure functions in EOS8, T2VOC, and TMVOC*
	0	Use three-phase capillary pressure functions
	≥1	Use two-phase capillary pressure functions; set p_{co} to zero
56		*Writing MESH file after simulation*
	0	Do not write MESH file
	≥1	Write MESH file with permeability modifiers
57		*Exclude equation from evaluation of `RE3` convergence criterion (see also `MOP2(34)` and `MOP2(45)`)*
	0	No residuals excluded in `RE3` criterion evaluation
	>0	Residuals from equation `MOP2(57)` excluded from `RE3` criterion evaluation
58		*Pressure dependence of liquid density*
	0	Based on reference (gas) pressure
	1	Based on liquid pressure ($p_{liq} = p_{gas} + p_{cap}$)
59		*Time shift or multiplication factors for time-dependent generation rates*
	0	No time shift for `F1` or multiplication factors for `F2` or `F3`
	1	Use `GX` as multiplication factor for `F2`
	2	Use `EX` as multiplication factor for `F3`
	3	Use `GX` as time shift for `F1`
	4	Use `EX` as time shift for `F1`
	5	Use `GX` and `EX` as multiplication factors for `F2` and `F3`, respectively
	6	Use `GX` as multiplication factor for `F2` and `EX` as time shift for `F1`
	7	Use `EX` as multiplication factor for `F3` and `GX` as time shift for `F1`
60		*Parent-daughter decay in EOS7R*
	0	RN1 decays into RN2
	1	RN1 does not decay into RN2

2.2 Continue Newton Raphson Iterations (Variable *NOITE*)

If variable *NOITE* in block PARAM.1 is set to a negative number, Newton Raphson iterations will continue beyond |*NOITE*| as long as the maximum residuals are decreasing. The maximum number of iterations is 2|*NOITE*|.

2.3 Maximum Number of Time Steps (Variable *MCYC*)

Variable *MCYC* in block PARAM.1 sets the maximum number of time steps. If *MCYC* = 9999, the maximum number of time steps is set to 10^6. If *MCYC* or *MCYPR* is negative, the maximum number of time steps or printout time frequency is set to 2^{-MCYC} or 2^{-MCYPR}, respectively.

2.4 Free Format Reading of Block TIMES

Printout times in block TIMES.2 can be specified in free format, indicated by an asterisk ("*") in Column 6, immediately after keyword TIMES. (Note that block TIMES.1 is still read in standard TOUGH2 format.) A time conversion factor can be provided in Columns 11–20 (e.g., 86400.0 if printout times are given in days).

If Column 7 contains the letter "D", printout times are given as dates, one per line, as shown in Figure 1. The first date is used as a reference date (unless a reference date is specified in the iTOUGH2 input file).

```
TIMES*       86400.0 ---*----3----*----4----*----5----*----6----*----7----*----8
    5   51       0.0       7.0
0.25 1.00 2.00 4.00 7.00

TIMESD---1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
    4
 1-Jan-2017 00:00:00
14-Sep-2017 02:00:00
15-Oct-2017 21:30:00
15-Nov-2017 07:45:29
```

Figure 1. Two examples of block TIMES, reading printout times in free and date format.

2.5 Array Dimensions (Block DIMEN)

Memory for (almost) all arrays is dynamically allocated, with the required array dimensions automatically determined by iTOUGH2. If special options are invoked (e.g., region definitions in blocks ROCKS and GENER, see Sections 5.3 and 6.13, respectively), the required array sizes cannot be automatically determined. In these cases, the user must provide the maximum number of materials and/or sinks/sources in a data block DIMEN. User-provided array dimensions may also be required for arrays related to blocks TIMES, TIMBC, MESHM, and some other, rarely used options. Should an array be dimensioned insufficiently an unambiguous error message will be issued, instructing the user to provide array dimensions as needed. Only dimensions specifically requested by iTOUGH2 need to be

specified; others may be left blank, in which case iTOUGH2 determines the maximum dimension internally. Figure 2 shows an example of block DIMEN. Block DIMEN has to come after the title and before any other TOUGH2 input block. (Dimensions for iTOUGH2-related arrays are provided in the iTOUGH2 input file using first-level command > DIMENSIONS.)

```
DIMEN----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
        20     10000       500       100      1000      1000      1000     10000
        10       100        50        20     10000       101        20      1200
         ^         ^         ^         ^         ^         ^         ^         ^
    MAXMAT   MAXTIME    MAXSS MAXGENTAB    MAXDX     MAXDY     MAXDZ    MAXNEL
MAXPROPVST MAXTIMVST MAXREGION MAXREGPAR   MAXLTAB     MXTBC    MXTBPT  MAXCOORD
MAXRPCPTAB

MAXMAT        : Maximum number of materials in block ROCKS
MAXTIME       : Maximum number of times in TOUGH2 block TIMES and iTOUGH2 block TIME)
MAXSS         : Maximum number of sinks/sources in GENER
MAXGENTAB     : Maximum number of time-dependent rates in GENER
MAXDX         : Maximum number of grid increments in X direction (for XYZ MESHM)
MAXDY         : Maximum number of grid increments in Y direction (for XYZ MESHM)
MAXDZ         : Maximum number of grid increments in Z direction (for XYZ MESHM)
MAXNEL        : Maximum number of elements of grid increments in radial direction
MAXPROPVST    : Maximum number of time-dependent properties
MAXTIMPROPVST : Maximum number of times when properties change
MAXREGION     : Maximum number of regions
MAXREGPAR     : Maximum number of parameters defining region
MAXLTAB       : Maximum number of time-dependent generation rates (if LTABCOUNT")
MXTBC         : Maximum number of elements with time vs. boundary condition data
MXTBPT        : Maximum number of time vs. boundary condition data
MAXCOORD      : Maximum coordination number (number of connections for a given element)
MAXRPCPTAB    : Maximum number of look-up table entries for rel. perm. and cap. Pres.
```

Figure 2. Block DIMEN for specifying maximum array dimensions.

2.6 Reading Blocks from External Files

In standard TOUGH, block ELEME and CONNE can be provided on an external file named *MESH*, and blocks INCON and GENER can be provided on external files *INCON* and *GENER*, respectively. In iTOUGH2, block ROCKS may also be given on an external file of arbitrary name, where the filename is specified after keyword "ROCKS FILE: ", as indicated in Figure 3. Note that the external file (here *rocks_block.dat*) must contain a full block, including the line containing the key word ROCKS as well as the empty line that closes the block. This option is useful to share material properties among multiple simulation files, or if properties are generated or modified using a pre-processor.

```
ROCKS FILE: rocks_block.dat    ----*----4----*----5----*----6----*----7----*----8
```

Figure 3. Specifying block ROCKS on an external file.

3 Printout Options

3.1 Printout Format (Block OUTPU)

The standard TOUGH2 output consists of blocks of predefined element-, connection-, and generation-related output variables. The amount and frequency of printout depends on variables *KDATA* (see Section 3.2) and *MCYPR* as well as the times specified in block TIMES (in iTOUGH2, printout frequency also depends on the calibration times specified in the iTOUGH2 input file).

Visualization of simulation results generally requires a reformatting program (such as EXT), which parses the standard output file along with spatial information provided in the MESH file and then generates a plot file in a format suitable for visualization (e.g., specifically TECPLOT and PetraSim).

The purpose of the input block OUTPU is (1) to allow the user to select the output variables to be printed, and (2) to provide output in formats that can either directly be used by standard visualization software, or that is more easily parsed by post-processing programs. Components of flow vectors are calculated at the centers of grid-blocks.

The line following the data block identifier OUTPU contains a keyword indicating the desired output format, currently either CSV, TECPLOT (or VISIT), PETRASIM or GNU. The number of output variables to be printed, MOUTVAR, is given on the next line, followed by MOUTVAR lines, each containing a keyword and either none, one, or two integer values (read in free format anywhere *after* Column 20) for further specification of the output variable. Table 4 summarizes the available keywords and options. The output variables will be written to separate files (one for each type and/or time). Alternatively, they can be combined according to the option selected by variable MOP2(26) (see Table 3). An example OUTPU block is shown in Figure 4; excerpts of resulting output files for element-, connection- and generation-related output variables are shown in Figure 5, Figure 6, and Figure 7, respectively.

The variables identified in block OUTPU are also used for printing out time series data (at each time step) at elements, connections, and sinks/sources according to TOUGH blocks FOFT, COFT, and GOFT. If no block OUTPU is specified, the default variable sets ISET = 1, ISET = 2, and ISET = 3 (see Table 4) will be printed for element-, connection-, and generation-related time series, respectively.
Block OUTPU must be specified *after* block MULTI.

Table 4. Keywords of Block OUTPU and Related Output Variables

Keyword	*ID1*	*ID2*	Output Variable
SET	ISET	–	Prints predefined sets of output variables ISET = 1: Set of main element-related output variables ISET = 2: Set of main connection-related output variables ISET = 3: Set of main generation-related output variables
NO COMMA	–	–	Omit commas between values
NO QUOTES	–	–	Omit quotes around values
NO NAME	–	–	Omit element names
COORDINATE COORD	NXYZ	–	Grid-block coordinates; mesh dimension and orientation are automatically determined, or can be specified through variable NXYZ: NXYZ = 1: Mesh is 1D "X " NXYZ = 2: Mesh is 1D " Y " NXYZ = 3: Mesh is 1D " Z" NXYZ = 4: Mesh is 2D "XY " NXYZ = 5: Mesh is 2D "X Z" NXYZ = 6: Mesh is 2D " YZ" NXYZ = 7: Mesh is 3D "XYZ"
INDEX	–	–	Index (running number) of elements, connections, or sinks/sources
TIME	–	–	Simulation time
KCYC, ITERATION	–	–	Time step index
DOMAIN	X_{min} X_{max} Y_{min} Y_{max} Z_{min} Z_{max}	–	Only print out if element is within bounding box (provide vertices of box on one line, starting after Column 18)
Element-related output variables			
MATERIAL ROCK TYPE	–	–	Material number
MATERIAL NAME ROCK NAME ROCK TYPE NAME	–	–	Material name
ABSOLUTE PERMEABILITY	ISOT	–	Absolute permeability in direction ISOT; if ISOT = 0, permeabilities related to all three directions are printed
POROSITY	–	–	Porosity
TEMPERATURE	–	–	Temperature
PRESSURE	IPH#	–	Pressure of phase IPH
SATURATION	IPH#	–	Saturation of phase IPH

RELATIVE	IPH#	–	Relative permeability of phase IPH
VISCOSITY	IPH#	–	Viscosity of phase IPH
DENSITY	IPH#	–	Density of phase IPH
ENTHALPY	IPH#	–	Enthalpy of phase IPH
MASS FRACTION	IPH	IC	Mass fraction of component IC in phase IPH
DIFFUSION1	IPH#	–	Diffusion parameter group 1 ($\phi\tau_0\tau_\beta\rho_\beta$) of phase β= IPH
DIFFUSION2	IPH#	–	Diffusion parameter group 2 ($\frac{P_0}{P}\left[\frac{T+273.15}{273.15}\right]^\theta$) of phase β= IPH
PSAT	–	–	Saturated vapor pressure
PRIMARY	IPV	–	Primary variable No. IPV; if IPV = 0, all NK+1 primary variables are printed
SECONDARY	IPH#	ISP	Secondary parameter No. ISP related to phase IPH, where ISP = 0 : All secondary parameters ISP = 1 : Phase saturation ISP = 2 : Relative permeability ISP = 3 : Viscosity ISP = 4 : Density ISP = 5 : Enthalpy ISP = 6 : Capillary pressure ISP = 7 : Diffusion parameter group 1 ISP = 8 : Diffusion parameter group 2
SUBSIDANCE	–	–	One-dimensional deformation, see Section 11
Connection-related output variables			
TOTAL FLOW TOTAL FLOW RATE	–	–	Total flow rate
FLOW RATE FLOW RATE CUMULATIVE FLOW	IPH	IC	Advective flow rate or cumulative flow. IPH=0; IC=0: Total flow IPH>0; IC=0: Flow of phase IPH IPH<0; IC=0: Flow of each phase IPH>0; IC>0: Flow of component IC in phase IPV IPH>0; IC<0: The flow of each component in phase IPH
MAX FLOW MAX FLOW TIME	IPH	–	Maximum advective flow rate or time of maximum advective flow rate of a given phase

			IPH > 0: Flow of phase IPH IPH < 0: Flow of each phase
DIFFUSIVE FLOW	IPH	IC	Diffusive flow rate. IPH=0; IC=0: Total flow IPH>0; IC=0: Flow of phase IPH IPH<0; IC=0: Flow of each phase IPH>0; IC>0: Flow of component IC in phase IPV IPH>0; IC<0: The flow of each component in phase IPH
HEAT FLOW	–	–	Heat flow rate
VELOCITY	IPH#	–	Flow velocity of phase IPH
Generation-related output variables			
GENERATION GENEARTION RATE	–	–	Production or injection rate
FLOWING ENTHALPY	–	–	Flowing enthalpy
FRACTIONAL FLOW	IPH#	–	Fractional flow of phase IPH (production only)
WELLBORE PRESSURE	–	–	Wellbore pressure (wells on deliverability only)
# If IPH = 0, the output variables of all phases are printed			

```
OUTPU----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
CSV
9
PRESSURE
SECONDARY          1 3
SECONDARY          1 4
MASS FRACTION      2 2
FLOW               1
VELOCITY           1
GENERATION
FRACTIONAL FLOW    1
FLOWING ENTHALPY
```

Figure 4. Example data block OUTPU.

```
"              ELEM","          PRES_GAS","          VISC_GAS","          DENS_GAS","          X_AIR_LIQ"
"                 ","              (PA)","            (PA*S)","         (KG/M**3)","                (-)"
"TIME [sec]  0.10000000E+01"
"              A1101",  0.401236819721E+06,  0.184570692607E-04,  0.465919882449E+01,  0.644956432615E-04
"              A2101",  0.147507792677E+06,  0.184570734485E-04,  0.171287344952E+01,  0.237110757024E-04
"              A3101",  0.116720218365E+06,  0.184570720594E-04,  0.135536558761E+01,  0.187621756067E-04
"              A4101",  0.113034689131E+06,  0.184570711567E-04,  0.131256899168E+01,  0.181697497798E-04
"              A5101",  0.112593498390E+06,  0.184570707619E-04,  0.130744588788E+01,  0.180988311033E-04
"              A6101",  0.112540683073E+06,  0.184570706029E-04,  0.130683260888E+01,  0.180903413682E-04
"              A7101",  0.112534360176E+06,  0.184570705403E-04,  0.130675919361E+01,  0.180893250018E-04
"              A8101",  0.112533603089E+06,  0.184570705159E-04,  0.130675040489E+01,  0.180892033046E-04
...
"TIME [sec]  0.30000000E+01"
"              A1101",  0.400995615118E+06,  0.184570681555E-04,  0.465639836140E+01,  0.644568724749E-04
"              A2101",  0.193338269214E+06,  0.184570735777E-04,  0.224506094596E+01,  0.310779891137E-04
"              A3101",  0.131359741970E+06,  0.184570735571E-04,  0.152536085210E+01,  0.211153850575E-04
"              A4101",  0.116601086798E+06,  0.184570724951E-04,  0.135398217228E+01,  0.187430259548E-04
"              A5101",  0.113380367818E+06,  0.184570716453E-04,  0.131658299037E+01,  0.182253154829E-04
"              A6101",  0.112706320690E+06,  0.184570711235E-04,  0.130875595137E+01,  0.181169665889E-04
...
```

Figure 5. Element-related output variables in CSV format based on `OUTPU` block shown in Figure 4.

```
"              ELEM1","              ELEM2","          FLOW_GAS","           VEL_GAS"
"                 ","                 ","            (KG/S)","             (M/S)"
"TIME [sec]  0.10000000E+01"
"              A1101","              A2101", -0.352311719350E-08, -0.150029808352E-05
"              A2101","              A3101", -0.422441958852E-09, -0.322959016651E-04
"              A3101","              A4101", -0.504964190656E-10, -0.487876354994E-05
"              A4101","              A5101", -0.604382074745E-11, -0.602969130852E-06
"              A5101","              A6101", -0.723495860628E-12, -0.724632771562E-07
"              A6101","              A7101", -0.866146117386E-13, -0.867914300296E-08
"              A7101","              A8101", -0.103710092602E-13, -0.103927648693E-08
"              A8101","              A9101", -0.124248446811E-14, -0.124509924289E-09
...
"TIME [sec]  0.30000000E+01"
"              A1101","              A2101", -0.288336365213E-08, -0.122860138353E-05
"              A2101","              A3101", -0.852256531812E-09, -0.497104674654E-04
"              A3101","              A4101", -0.202352019394E-09, -0.173716307947E-04
"              A4101","              A5101", -0.441276789154E-10, -0.426779727997E-05
...
```

Figure 6. Connection-related output variables in CSV format based on `OUTPU` block shown in Figure 4.

```
"              ELEM","            SOURCE","               GEN","           FF_GAS","               ENTG"
"                 ","                 ","            (KG/S)","               (-)","            (J/KG)"
"TIME [sec]  0.10000000E+01"
"              AQ102","             TOP 2", -0.162800000000E-10,  0.100000000000E+01,  0.105761536312E+06
"              AQ103","             TOP 3", -0.193900000000E-10,  0.100000000000E+01,  0.105761536231E+06
"TIME [sec]  0.30000000E+01"
"              AQ102","             TOP 2", -0.162800000000E-10,  0.100000000000E+01,  0.105761535951E+06
"              AQ103","             TOP 3", -0.193900000000E-10,  0.100000000000E+01,  0.105761535806E+06
...
```

Figure 7. Generation-related output variables in CSV format based on `OUTPU` block shown in Figure 4.

3.2 Printout Control (Variable *KDATA*)

In standard TOUGH2, printout of simulation results is controlled by variables *KDATA* and *MCYPR* in block PARAM.1, as well as by the additional printout times *TIS* given in block TIMES. In iTOUGH2, the calibration times are also stored in array *TIS* which means that at each calibration time the amount of printout specified by variable *KDATA* is written to the output file. This may make the TOUGH2 output file extremely long, and requires unnecessary CPU time for disk writing, since the TOUGH2 output file is overwritten each time a new TOUGH2 simulation is initiated by iTOUGH2.

If a negative number is specified for variable *KDATA*, the amount of printout is reduced (see Table 5), saving both disk space and CPU time. Recall that printout at the calibration points is always written to the iTOUGH2 plot file (see command >>> FORMAT). Also, if a TOUGH2 run is terminated due to a convergence failure or using command kit, the full output is automatically generated for the last time step.

Full output is always provided for the times specified in the TOUGH2 block TIMES; the amount of printout is given by the absolute value of *KDATA*. The times provided in TOUGH2 block TIMES can be in arbitrary order; they are sorted internally. For *KDATA* = 4 or 5, select variables for, respectively, elements and connections, along with coordinates are printed in Tecplot format to files named **_eleme.tec* and **_conne.tec*.

Table 5. Amount of Printout as a Function of Variable *KDATA*

-5	-4	-3	-2	-1	Printout of …	1	2	3	4	5
-	-	-	+	+	volume and mass balance	+	+	+	+	+
-	-	-	-	+	generation rates	+	+	+	+	+
-	-	-	-	-	element variables	+	+	+	+	+
-	-	-	-	-	connection variables	-	+	+	+	+
-	-	-	-	-	primary variables	-	-	+	+	+
-	-	-	-	-	elements Tecplot	-	-	-	+	+
-	-	-	-	-	connections Tecplot	-	-	-	-	+

3.3 Excluding Domains from Global Material Balance Printout (Variable *SPHT*)

Domains with an element volume larger than 10^{20} m^3 or a negative rock grain specific heat *SPHT* or with *SPHT* greater than 10^4 J/kg °C are excluded from global material balance calculations. The absolute value of *SPHT* is used in the heat balance equation. Material balances for individual rock types are available through iTOUGH2 commands >> TOTAL MASS and >>> VOLUME.

4 Mesh Generation

4.1 Meshmaker Options

The following options have been added for more convenient generation of Cartesian and radial meshes using the built-in mesh generator MESHM:

- Meshmaker can generate meshes with element names consisting of 5 (default), 6, 7, 8, and 9 characters. The length of the element name (variable *IELL*) can be indicated in Column 6, immediately after keyword MESHM, or by setting *MOP2(2) = IELL* (see Table 3). The first three characters of the element name are alphanumeric; the remaining (*IELL* - 3) positions of the element name must be integer numbers.
- If an asterisk ("*") is specified in Column 7 on the line that contains keyword MESHM, lists of grid increments (for Cartesian meshes), radial distances and layer thicknesses (for radial meshes) can be specified in free format (rather than format E10.4).
- If MOP2(27) is set to 1, or if a plus sign ("+") or "C" is specified in Column 8 on the line that contains keyword MESHM, the generated element names are identical to the element index, i.e., integers from 1 for the first element to NEL for the last element.
- The coordinates can be shifted by ΔX, ΔY, and ΔZ (variables *SHIFTZ*, *SHIFTY*, and *SHIFTZ*) provided in Columns 11–20, 21–30, and 31–40 on the line that contains keyword XYZ, RZ2D, RZ3DL, or MINC.
- The mesh can be rotated (variable *DEG*, provided in Columns 41–50 on the line that contains keyword XYZ, RZ2D, or RZ3DL). If a non-zero value is given in columns 51–60, the coordinates will be rotated in addition to the gravity vector *BETAX*.
- For radial meshes (keywords RZ2D or RZ2DL), a material number or the corresponding five-character material name can be specified in Columns 6–10 on the line following keywords RADII, EQUID, LOGAR. The material name will be added to the corresponding elements, allowing, for example, to directly indicate whether the elements represent the well, skin zone, or formation.
- If mesh information (i.e., blocks ELEME and CONNE) are specified in the input file, they will be written to file MESH in the same format, i.e., without a potential loss of accuracy.
- Coordinates can be written in 20 (instead of 10) character long fields; see *MOP2(22)*.
- After mesh generation using MESHM and after completion of a simulation, file MESH is written with maximum accuracy of coordinates and other output variables (specifically permeability modifiers).
- The input and/or output MESH files can be either ASCII text files (default) or binary files (see Table 3, *MOP2(25)*).

4.2 Secondary Mesh (Blocks ELEM2, CONN2)

This enhancement was introduced especially for the estimation of a skin radius (see command >> SKIN) or a MINC parameter (see command >> MINC), both requiring that a new TOUGH2 mesh is generated automatically each iteration without user interference. In order to achieve this, a primary mesh must be generated using the MESHMAKER utility. Elements and connections of this primary mesh can then be overwritten by a secondary mesh provided through blocks ELEM2 and CONN2, which have the same format as blocks ELEME and CONNE, respectively. The names of the elements and connections to be modified by the secondary mesh must be identical with the corresponding ones of the primary mesh. If a secondary mesh is specified, both blocks, ELEM2 and CONN2, must be given; either of the two keywords may be followed by an empty line to indicate that no modification is made.

Figure 8 illustrates an application. A radial mesh is generated using MESHMAKER for simulating a pump test. In block ELEM2, the volume of the first grid block is changed to represent the actual interval volume. In block CONN2, the nodal distance from the first element to the interface is reduced to a very small number as usually done for connections to boundary elements. These last two modifications, which are usually made by editing the mesh file, are now automatically performed whenever a new mesh is generated during an iTOUGH2 run for the estimation of the skin radius.

```
MESHMAKER-----*----2----*----3----*----4----*----5----*----6----*----7----*----8
RZ2D
RADII
    2WELLB
 0.000E+00 0.100E+00
LOGAR
   20SINKZ 0.300E+00 0.100E-01
LOGAR
   80RESER 1.000E+01
LAYER
    1
 0.100E+01

ELEM2----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
A1  1          BOUND .1250E+50

CONN2----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
A1  1A1  2                   1 .1000E-10 .5332E-02 .6283E+00
```

Figure 8. Primary and secondary mesh generation.

4.3 Mesh Rotation/Translation (Block COORD)

Most mesh generators (including MESHM) build a mesh in a local coordinate system, while spatial information of the simulated reservoir often refers to a different coordinate system. A simple coordinate transformation (translation and rotation) can be executed to adapt the local mesh to the global coordinate system. This coordinate transformation can be done for the coordinates in block ELEME, but also those of the GSLIB mesh (see Section 5.9), the definition of property regions (see Sections 5.6 and 6.13), as well as observations referring to elements or connections that are given by coordinates or regions (see iTOUGH2 commands >>> ELEME COORDINATE or >>> CONNECTION COORDINATE). While only a single set of transformation parameters can be given, the user may select which types of coordinates (i.e., referring to the TOUGH2 mesh, GSLIB mesh, region definitions, or observations) are given in a local coordinate system (i.e., will be transformed) and which ones are already in the global coordinate system (i.e., will not be transformed). Note that (except for polygonal prisms) the geometrical objects defining regions (see Sections 5.6 and 6.13) can be rotated individually. It is therefore possible (and may be preferable, specifically for sink/source regions) to define these regions directly in the global coordinate system to avoid repeated mesh transformations. If regions defined by polygonal prisms (see *IREGGEOM*=5) are not aligned with the axes of the local coordinate system (see *IJKPRISMAXIS*), they need to be transformed using the method described in this section.

If a rotation outside of the horizontal plane is performed, the cosine of a connection to the gravitational vector (variable *BETAX*) will be recalculated based on the transformed coordinates. The rotation angles are those defined by GSLIB (see *Deutsch and Journel* [1992] or *Finsterle and Kowalsky* [2007]).

Parameters for the coordinate transformation are specified in block COORD.

COORD Invokes coordinate transformations. If an asterisk ("*") is given immediately after keyword COORD, the transformation parameters can be specified in free format.

Record COORD.1

Format (5I5) or free format if * after keyword COORD.
(*ICOORDTRANS*(I), I=1,5)

ICOORDTRANS(1) : Coordinates of block ELEME.

0: Don't transform TOUGH2 mesh coordinates.

1: Transform TOUGH2 mesh coordinates.

-1: Transform TOUGH2 mesh coordinates; back transform before writing final MESH file.

ICOORDTRANS(2) : Coordinates of GSLIB mesh.

0: Don't transform GSLIB mesh coordinates.

1: Transform GSLIB mesh coordinates.

ICOORDTRANS(3) : Coordinates of region definitions.

0: Don't transform region definitions mesh coordinates.

1: Transform region definitions mesh coordinates.

ICOORDTRANS(4) : Coordinates of observation points.

0: Don't transform observation points coordinates.

1: Transform observation points coordinates.

ICOORDTRANS(5) : Recalculate gravity component BETAX.

0: Don't recalculate gravity component.

1: Recalculate gravity component.

Record COORD.2

Format (3E10.4,/,3E10.4,/,3E10.4) or free format if * after keyword COORD. (*RCOORDTRANS*(I), I=1,9)

RCOORDTRANS(1-3): Origin of rotation.

RCOORDTRANS(4-6): Rotation angles.

RCOORDTRANS(7-9): Translations in X, Y, and Z directions.

Figure 9 shows coordinated transformation block COORD. Block MESHM generates a structured grid (which is shifted using *SHIFTZ*, *SHIFTY*, and *SHIFTZ*, see Section 4.1). The mesh is then transformed by rotation around the point at (2/5/-1) using the rotation angles (-20/-15/-10) before being shifted by (1/6/-1). Moreover, the source term in block GENER uses a region definition (see Section 6.13), which is defined using coordinates of the local coordinate system; this region is transformed along with the TOUGH2 mesh by setting *ICOORDTRANS*(3)=1. The rotation leads to a mesh in which none of the connections is horizontal; the gravity component is therefore recalculated for all connections by setting *ICOORDTRANS*(5)=1.

The original mesh in the local coordinate system (as generated by MESHM) is shown in Figure 10; the transformed mesh is shown in Figure 11. The source region is highlighted in red, indicating its transformation along with the coordinate system.

```
Demonstrate transformation of mesh and region coordinates
DIMEN----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
    2
ROCKS----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
DOMAN          2650.       .10    1.E-13    1.E-13    1.E-13       2.0     1000.
&WELL          2650.       .10    1.E-12    1.E-12    1.E-10       2.0     1000.
    3    0
       2.0       3.0       0.0
       2.0       3.0      -4.0
      0.01

COORD*---1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
    1    0    1    1    1
       2.0       5.0      -1.0
     -20.0     -15.0     -10.0
       1.0       6.0      -1.0
PARAM----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
   2   1       11000009000000004000050000
                         1.0
     1.E-5                                          1.E-8
             1.0E+06                0.00               20.00

START----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
MOMOP----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
                        4                        1
MESHM----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
XYZ
       0.0      -0.5      -0.5       0.5
NX       2       1.0
NX       5       0.2
NX       2       1.0
NY       3       1.0
NY       5       0.2
NY       7       1.0
NZ       1       1.0
NZ       5       0.2
NZ       1       1.0


GENER----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
    1XYZ 1                          WATE        1.0
    3    0    0    1
       2.0       3.0      -0.9
       2.0       3.0      -1.1
      0.01

OUTPU----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
CSV
2
COORDINATES
PRESSURE

ENDCY----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
```

Figure 9. Coordinate transformation using block `COORD`.

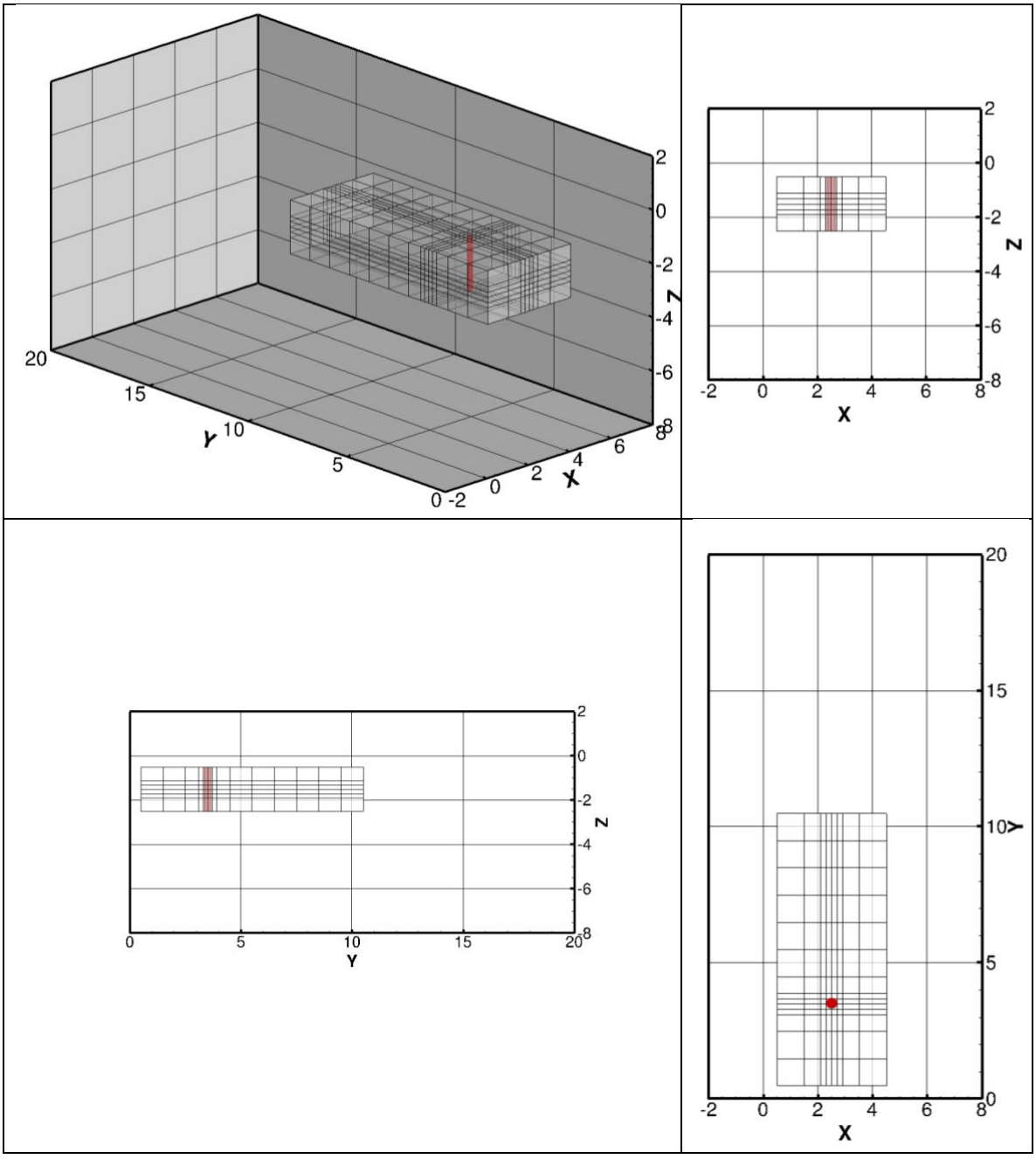

Figure 10. Computational mesh in local coordinate system prior to transformation.

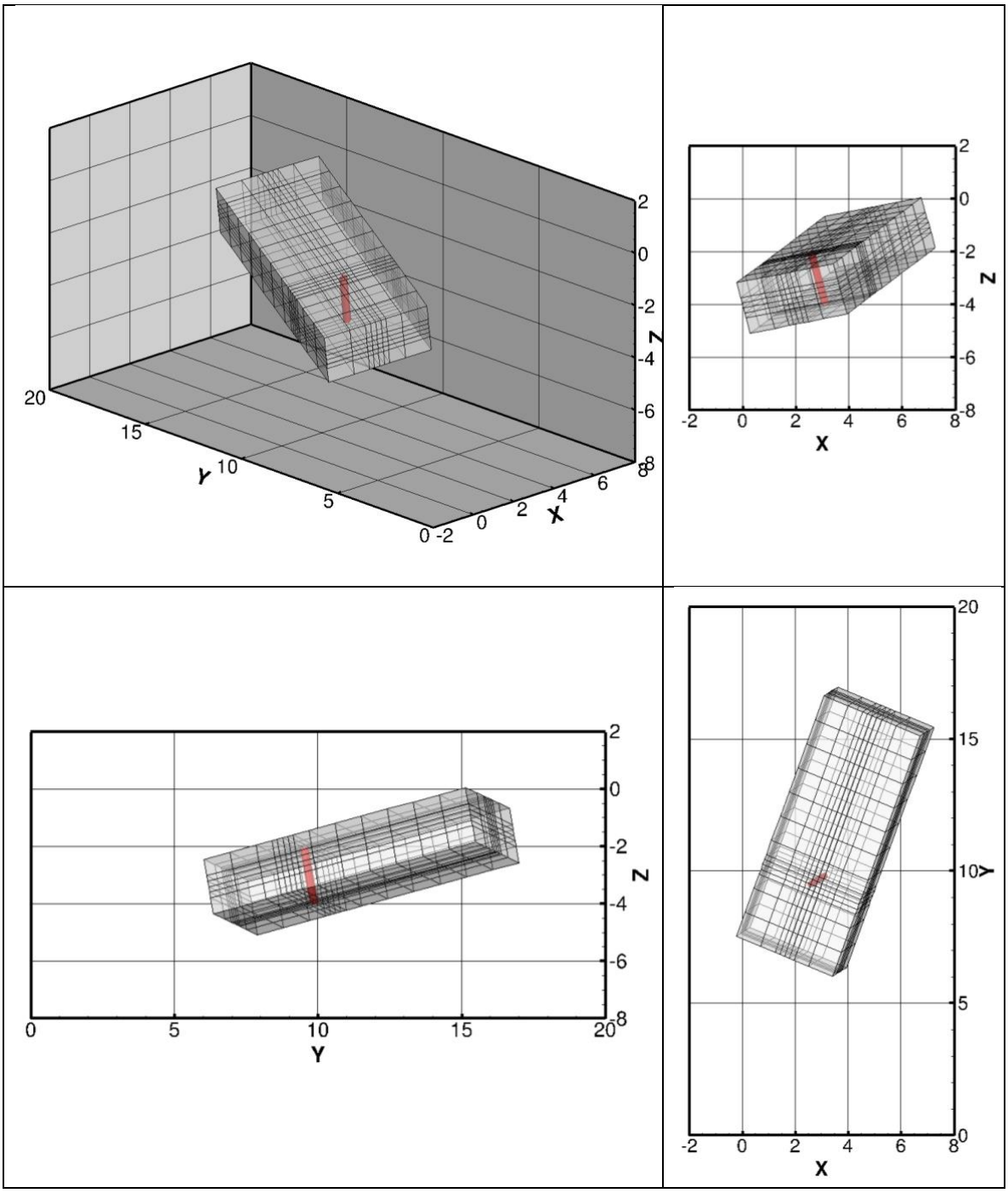

Figure 11. Computational mesh after coordinate transformation.

5 Material Property Assignment

5.1 Element-by-Element Permeabilities (Blocks ELEME and INCON)

Heterogeneity is introduced into TOUGH2 models by assigning a certain rock type, for which material properties are defined in block ROCKS, to each grid block. The maximum number of rock types one can specify is given by variable *MAXROC* (see file *maxsize.inc*). For high-resolution simulations of permeability heterogeneity, it is inconvenient to specify hundreds of rock types, one for each grid block. TOUGH2 has been extended so that grid block permeabilities or permeability modifiers can be directly specified in block ELEME, columns 41-50. If a positive value is given, it is interpreted as absolute permeability; if a negative value is provided, it is interpreted as a permeability modifier, i.e., a factor with which the absolute permeability specified in block ROCKS is multiplied. If columns 41–50 are blank for the first element, the element-by-element permeabilities are ignored.

The anisotropy ratio as given by ratios of *PER(1)*, *PER(2)*, and *PER(3)* in block ROCKS is preserved, i.e., the single permeability modifier is applied to all three permeabilities. To read three permeabilities or permeability modifiers, i.e., one for each of the three directions, *MOP2*(20) has to be set according to Table 3. (Permeabilities and permeability modifiers can also be read from an external file and mapped onto the TOUGH2 mesh; see block MAPPI described in Section 5.2). Alternatively, the same information can be provided through block INCON, columns 31–60. Finally, element-specific permeabilities can also be provided in block ELEME following the approach described in Section 5.3. The permeabilities specified in block ROCKS will be overwritten by those provided in block ELEME, which in turn will be overwritten by those given in block INCON. Note that the permeability or permeability modifiers are stored in array *USERX(i,n)*, $i = 1 \ldots 3$, $n = 1 \ldots NEL$. To write element-by-element parameters to file *SAVE*, set *MOP*(13) to 1.

Using a permeability modifier instead of permeability itself has the advantage that the mean of the permeability field can easily be changed (or estimated) by adjusting parameter *PER(ISOT)* in block ROCKS.

5.2 Element-by-Element Permeabilities from External Files (Block MAPPI)

Instead of specifying element-by-element permeabilities or permeability modifiers in block ELEME as described in Section 5.1, heterogeneous and anisotropic, potentially transformed permeabilities or permeability modifiers may be generated externally and provided on a file, which is read in and then mapped onto the TOUGH2 mesh. Various file formats as well as mapping procedures are supported; the related information has to be provided in block MAPPI, which contains the following records (read in free format):

Record MAPPI.1

FILENAME : Filename containing (potentially transformed) permeabilities in one of the supported formats (see *MAPFMT*); make sure the file is available in the local directory

Record MAPPI.2

MAFMT : File format

0: GSLIB format as internally generated by iTOUGH2 [*Finsterle and Kowalsky*, 2007]

1: *VALUE*(*i*), *i* = 1, …, *MAPNVAL*
Values need to be provided in same order as elements in block ELEME

2: *X Y Z VALUE*(*i*), *i* = 1, …, *MAPNVAL*
Coordinates followed by values; the coordinates may or may not be identical to those in block ELEME

3: *I J K VALUE*(*i*), *i* = 1, …, *MAPNVAL*
Grid indices followed by values; assumes values are provided on a regular grid (see record MAPPI.6); the TOUGH2 grid does not need to be regular)

4: *I J K NFRACT COND*(*i*), *i* = 1, …, *MAPNVAL*
Format used by discrete fracture network to continuum grid mapper of *Parashar and Reeves* [2011]; assumes values are provided on a regular grid with uniform grid spacing (see record MAPPi.6); the TOUGH2 grid does not need to be regular

MAPHEADER : Number of header lines to be skipped before data reading

MAPNVAL : Number of values to be read per grid block (1 for single permeability or modifier applied to all three permeabilities; 3 for permeabilities or modifiers applied to different directions)

MAPPOINTS : Number of data lines (points in space where permeability information is provided)

Record MAPPI.3

IFIELD1 : Identifies variable

0: porosity, *PHI(N)*

1–3: permeabilities or permeability modifiers, *USERX(i,N)*

4: capillary-strength parameter (see Section 7), *USERX(4,N)*

5: user specified, *USERX(5,N)*

IFIELD2 : Interpret as value or modifier

0: value

1: modifier

IFIELD3 : Variable transformation

0: no transformation

1: value represents decadic logarithm

2: value represents natural logarithm

Record `MAPPI.4`

NMAPROCK : Number of material types to be given (see *IMAPROCK*).

Mapping can be restricted to certain zones within the TOUGH2 model, where a zone is defined by all elements that belong to one or more material domains. This option may be useful to exclude the properties of special elements (e.g., those representing boundary elements, wells, engineered structures, etc.) If *NMAPROCK* is zero, the heterogeneous values will be mapped onto the entire TOUGH2 grid.

IMAPROCK : Material type numbers (sequence number of material as entered in block `ROCKS`) of domains to be included in mapping, *IMAPROCK*(*i*), *i* = 1, …, *NMAPROCK*

Record `MAPPI.5`

MAPPING : Type of mapping and averaging procedure

1: nearest neighbor; the property value from the point nearest to the TOUGH2 element center point will be taken

2: arithmetic mean; the arithmetic mean of all property values within a user-specified radius from the TOUGH2 element center point will be taken

3: geometric mean; the geometric mean of all property values within a certain radius from the TOUGH2 element center point will be taken

4: harmonic mean: the harmonic mean of all property values within a certain radius from the TOUGH2 element center point will be taken

SEARCHR : Search radius; specifying a non-zero search radius for *MAPPING* = 1 may speed up the mapping, as the nearest point is considered identified as soon as the distance is less than the search radius. For *MAPPING* = 2, 3, or 4, a non-zero search radius has to be specified. Three options exist to specify the search radius *r*,

where *SR* is an input parameter to be provided by the user, *V* is the volume of the given TOUGH2 element, and *D* = 1, 2, or 3 is the dimension of the model:

$$r = \begin{cases} -SR \cdot V^{1/D} & SR < 0 \\ V^{1/D} & \text{for} \quad SR = 0 \\ SR & SR > 0 \end{cases}$$

Record MAPPI.6 (only if *MAPFMT* = 3 or 4)

X0, *Y0*, *Z0* : Origin of coordinate system used in the external file (the TOUGH2 mesh may be shifted by specifying *SHIFTX*, *SHIFTY*, and *SHIFTZ* in columns 51–60, 61–70, and 71–80, respectively, on the line containing keyword ELEME).

DX, *DY*, *DZ* : Grid spacings

During the mapping process, each element of the TOUGH2 grid is visited. First, it is checked whether the element belongs to a material type that defines the mapping zone. If so, the coordinates of the element are compared to the coordinates of grid points specified in the external file, and if the distance between the two points is less than the search radius, the property values are retained, averaged, and eventually assigned to the element.

Figure 12 shows an example of block MAPPI in a TOUGH2 input file; Figure 13 shows an excerpt of the data file *Log10PermModifier.dat*.

```
MAPPIng--1----*----2----*----3----*----4----*----5----*----6----*----7
Log10PermModifier.dat
    3    1    3       9600
    1    1    1
    3                1    4    5
    1   1.0E-03
   5.0  5.0 -28.75
  10.0 10.0   2.50
```

Figure 12. TOUGH2 input file for mapping permeability modifiers from external file.

```
X-index  Y-index  Z-index  log10(pmx1)  log10(pmx2)  log10(pmx3)
      1        1        1        -1.03        -1.14        -1.98
      1        1        2         0.65         0.83        -0.76
      1        1        3         2.98         2.22         1.56
    ...
     40       20       12        -1.71        -0.98        -2.83
```

Figure 13. Data file containing log-10 permeability modifiers on indexed regular mesh.

5.3 Element-by-Element Property Assignment (Blocks HETER and ELEME)

As an extension to the element-by-element permeability assignment described in Section 5.1, other isotropic and anisotropic properties as well as initial conditions can be assigned to individual elements in block ELEME. In block HETER, which must be placed after block MULTI and before block ELEME, the user may specify the property variables that will be provided in block ELEME after Column 80.

On the line following keyword HETER, the number of columns (variable *NHETERO*) with element-specific property values or initial conditions must be provided in Columns 1–10. An optional FORTRAN format statement (enclosed in parentheses) can be provided after Column 11; the values in block ELEME will be read in that format. If no format statement is provided, values will be read in (E10.4) format; if a single asterisk "*" is provided, values will be read in free format. Note that if free-format reading is selected, values must be provided for all elements and all variables; if a FORTRAN format statement or the default format is selected, values may be provided only for elements selected for heterogeneous treatment.

The next line contains the name of the variables for which element-specific values will be provided in block ELEME. The names must be provided in (A10) format. If *NHETERO* > 8, names are provided on additional lines. Table 6 shows the list of recognized variable names. An integer index may follow the variable name. These indices typically refer to anisotropic properties (i.e., the reference to direction *ISOT*), the index of relative permeability and capillary pressure parameters, or the index for initial conditions. The index must be the last character of the variable name. If no index is provided, the value read in block ELEME will be interpreted as an isotropic value and assigned to all directions. The special name DUM deactivates the corresponding column without the need to delete it in block ELEME. An example is shown in Figure 14.

Columns with element-specific property values corresponding to the variables defined in block HETER are provided in block ELEME after the standard TOUGH2 input (i.e., in slots > 80). If a positive value is given, the value represents the property; if a negative value is given, its absolute value is the modification factor with which the corresponding variable provided in block ROCKS is multiplied. If the value is identical to zero, no element-specific value is assigned (i.e., a very small, non-zero value has to be specified to approximate an element-specific value of zero.)

```
HETER----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
         7   (2E10.4,E15.5E3,4F5.2)
      PERM      PER3      INIT1      PORO      CWET1       DUM      CWET3
```

Figure 14. HETER block with names of property variables provided in block ELEME.

Table 6. Names of Element-Specific Property Variables Provided in Block ELEME.

Name	Index	Property	Comment
DUM	none	n/a	Ignore corresponding column in ELEME.
PERM	1–3	Anisotropic absolute permeability	If no index, permeability will be isotropic; these permeabilities overwrite *PERM* provided in ROCKS; they are overwritten by *USERX* in INCON, permeabilities provided through MAPPI or generated by GSLIB, and may be modified by property regions (see Section 5.3).
CWET	1–3	Anisotropic wet thermal conductivity	If no index, thermal conductivity will be isotropic; overwrites *CWET* in ROCKS.
TORT	1–3	Anisotropic tortuosity factor	If no index, tortuosity factor will be isotropic; overwrites *TORTX* provided in ROCKS.
PORO	none	Porosity	These porosities overwrite *POR* provided in ROCKS; they are overwritten by *PORX* provided in INCON, and porosities provided through MAPPI or generated by GSLIB, and may be modified by property regions (see Section 5.3).
INIT	1–*NK+1*	Initial condition	These initial conditions overwrite *DEP* in PARAM.4 and *X* in INDOM; they are overwritten by *X* in INCON. Only positive values (i.e., no modifiers) can be specified for this variable type.
RP	1–23	Parameters of relative permeability functions	These relative permeability parameters overwrite *RPD* in RPCAP and *RP* provided in ROCKS.1.2; they refer to the function *IRP* assigned by *MATX* to the specific element.
CP	1–23	Parameters of capillary pressure functions	These capillary pressure parameters overwrite *CPD* provided in RPCAP and *CP* in ROCKS.1.3; they refer to the function *ICP* assigned by *MATX* to the specific element.
COMPR	None	Pore compressibility	These pore compressibilities overwrite *COM* provided in ROCKS.1.1.
EXPAN	none	Pore thermal expansivity	These pore thermal expansivities overwrite *EXPAN* provided in ROCKS.1.1.
ARCHM	none	Archie's Law coefficient *m*	These Archie's Law coefficients *m* overwrite *ARCHIEM* provided in ROCKS (see Section 15).
ARCHN	none	Archie's Law coefficient *n*	These Archie's Law coefficients *n* overwrite *ARCHIEN* provided in ROCKS (see Section 15).
ARCHA	none	Archie's Law coefficient *A*	These Archie's Law coefficients *A* overwrite *ARCHIEA* provided in ROCKS (see Section 15).
ARCHB	none	Archie's Law coefficient *B*	These Archie's Law coefficients *B* overwrite *ARCHIEB* provided in ROCKS (see Section 15).
ARCHP	none	Archie's Law porosity ε	These Archie's Law porosities ε overwrite *ARCHIEP* in ROCKS (see Section 15).

An example of an ELEME block with heterogeneous properties is shown in Figure 15, based on the variable selection made in block HETER of Figure 14. The values will be read in the format as specified in block HETER, starting in column 81.

The material properties and initial conditions for the first three elements are identical and consistent with those that would be used if no element-by-element properties were defined (i.e., if all properties were only specified in blocks ROCKS). Elements 4 and 5 also have the same properties, with (positive) values or (negative) modifiers used. Note that the sixth column, which holds values for the thermal conductivity for *ISOT* = 2, is ignored, because the corresponding column header in block HETER reads DUM. The value of 2.2 (specified in block ROCKS) will be used for all elements. Elements 6 and 7 have unique properties and initial conditions assigned to them. Specifically, the permeabilities *PERM(1)*, *PERM(2)* and *PERM(3)* specified in block ROCKS are first replaced by the value or multiplied by the modifier given in Columns 81–90, after which *PERM(3)* is overwritten again by the values given in Columns 91–100.

Recall that the permeabilities, porosities, and initial pressures specified here in block ELEME may be overwritten by those given in block INCON. Permeabilities and porosities may also be overwritten if spatially correlated random property fields were generated using GSLIB (see Section 5.9) or provided externally using block MAPPI (see Section 5.2).

If the number of heterogeneous variables *NHETERO* in block HETER is a negative number, the element-specific property values are not provided as part of block ELEME but need to be contained in a separate file named *HETER*. File *HETER* must contain exactly as many lines with values in the same order as the elements in block ELEME, each line containing heterogeneous, element-specific parameters in the format specified in block HETER.

```
Demonstrate heterogeneous and anisotropic property assignment
MULTI----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
    2    3    2    6
HETER----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
         7   (2E10.4,E15.10E3,4F5.2)
      PERM      PER3     INIT1      PORO     CWET1       DUM     CWET3
ROCKS----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
SAND     1     2650.     .3000 1.000E-12 1.000E-12 1.000E-12     -1.00     800.0
      2.50      2.20      2.00
 1.000E-08

PARAM----1--MOP:123456789012345678901234----*----5----*----6----*----7----*----8
   2   1       110000900000000400003000
 0.000E+00 6.000E+02 2.500E+00 1.000E+01

 100000.000000000000   10.30000000000000  20.00000000000000
                                                                                 per(all)    per(3) init. pressure  por cw-1 cw-2 cw-3
ELEME----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8----*----9----*---10----*---11----*---12----*---13----*
1                  10.1000E+010.1000E+01              0.000     0.000     0.000 1.000E-12 1.000E-12 1.0000000E+005 0.30  2.5  2.5  2.0
2                  10.1000E+010.0000E+00              0.000     0.000    -1.000 -1.00E+00 -1.00E+00                -1.0 -1.0 -1.0 -1.0
3                  10.1000E+010.0000E+00              0.000     0.000    -2.000
4                  10.1000E+010.0000E+00              0.000     0.000    -3.000     -10.0 1.000E-13                0.45  3.0  2.5  1.5
5                  10.1000E+010.0000E+00              0.000     0.000    -4.000   1.0E-11    -0.100                -1.5  3.0  3.0  1.5
6                  10.1000E+010.0000E+00              0.000     0.000    -5.000     -10.0 2.345E-13 1.2345678E+005 0.12 1.23 4.56 7.89
7                  10.1000E+010.0000E+00              0.000     0.000    -6.000 1.000E-11 1.000E-13 2.0000000E+005 0.99
  ...            ...       ...       ...             ...       ...       ...       ...       ...            ...  ...  ...  ...  ...
```

Figure 15. ELEME block with heterogeneous properties.

5.4 Depth- or Temperature-Dependent Permeability and Heat Capacity

Permeability and heat capacity can be specified as a function of temperature or depth. To invoke this feature, set *NAD*=5 in block ROCKS for the relevant material.

Record ROCKS.1.4 (optional, NAD=5 only)

Format (I5,5X,3E10.4)
IPFT, (*PFT(I)*,*I*=1,3)

|*IPFT*|	If positive, permeability is a function of temperature T If negative, permeability is a function of depth (provide Z-coordinate in columns 71–80 in block ELEME) 1: Linear permeability change between *PFT*(1) and *PFT*(2) 2: Log-linear permeability change between *PFT*(1) and *PFT*(2)
PFT(1)	Do not change permeability if $T <$ *PFT*(1) or $Z >$ *PFT*(1)
PFT(2)	Maximum permeability change if $T >$ *PFT*(2) or $Z <$ *PFT*(2)
PFT(3)	Maximum permeability modifier if |*IPFT*| = 1; Maximum log-permeability modifier if |*IPFT*| = 2

Record ROCKS.1.5 (optional, *NAD*=5 only)

Format (I5,5X,3E10.4)
ICFT, (*CFT(I)*,I=1,3)

|*ICFT*|	If positive, heat capacity is a function of temperature T If negative, heat capacity is a function of depth (provide Z-coordinate in columns 71–80 in block ELEME) 1: Linear heat-capacity change between *CFT*(1) and *CFT*(2) 2: Log-linear heat-capacity change between *CFT*(1) and *CFT*(2)
CFT(1)	Do not change heat capacity if $T <$ *CFT*(1) or $Z >$ *CFT*(1)
CFT(2)	Maximum heat capacity change if $T >$ *CFT*(2) or $Z <$ *CFT*(2)
CFT(3)	Maximum heat capacity modifier if |*ICFT*| = 1; Maximum log-heat capacity modifier if |*ICFT*| = 2

Figure 16 shows an example of a ROCKS block invoking permeability and heat capacity reduction. It reduces permeability log-linearly by 6 orders of magnitude between Z = -3000 m and Z = -5000 m; moreover, it reduces heat capacity linearly by a factor of 2 between T = 1050°C and T = 1150°C.

```
ROCKS----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
POMED    5     2650.       .01    6.E-15    6.E-15    6.E-15       2.1     1000.
 1.000E-09
    3           0.30       0.05
    1           0.00       0.00      1.00
   -2       -3000.00   -5000.00     -6.00
    1        1050.00    1150.00      0.50
```

Figure 16. Specifying depth- or temperature-dependence of permeability and heat capacity.

5.5 Pressure-Dependent Porosity and Permeability

Pressure-dependent porosity and permeability can be invoked by specifying the pore compressibility *COM* in block ROCKS1.1 as a negative value, followed by one additional line that defines the pressure-dependence of porosity and permeability. If *IPDPP* is positive, the empirical function of *Rutqvist et al*. [2002] is used to calculate porosity as a function of effective stress; if *IPDPP* is negative, the absolute value of *COM* is used as the pore compressibility with either the Terzaghi, Bishop, or Pereira models used to calculate the effective stress. If *IPDPP* is positive, neither *POR* nor *COM* are used to calculate porosity or porosity change. For all effective stress models, porosity change remains affected by the thermal expansion coefficient *EXPAN*.

Record ROCKS.1.1.1 (optional, negative *COM* only)

Format (I5,5X,7E10.4)
IPDPP, (*PFT(I)*, *I*=1,7)

|*IPDPP*| Choice of permeability-porosity relationship and porosity change model.
Permeability-porosity relationship:

1: $k(\phi) = k_0 \cdot (\phi/\phi_0)^{\beta}$ (5.5-1a)

2: $k(\phi) = k_0 \cdot e^{\beta \cdot \left(\frac{\phi}{\phi_0} - 1\right)}$ (5.5-1b)

3: $k(\phi) = 10^{(\gamma + \beta \cdot \phi)}$ (5.5-1c)

Porosity change model:

positive: *Rutqvist et al*. [2002]:

$$\phi(P) = \phi_r + (\phi_0 - \phi_r) \cdot e^{-\beta \sigma'} \quad (5.5\text{-}2a)$$

negative: Use *COM* and *EXPAN* to update porosity at each time step as a function of $\sigma'(P(S_l))$ and T

$$\Delta\phi = (-COM \cdot \Delta\sigma' + EXPAN \cdot \Delta T) \quad (5.5\text{-}2b)$$

PFT(1) σ_T: Constant confining stress [Pa]

positive: effective stress according to *Terzaghi* [1936]:

$$\sigma' = \sigma_T - P \quad (5.5\text{-}3a)$$

negative: effective stress according to *Pereira et al.* [2010] (see Eq. (5.5-4) for calculation of χ):

$$\sigma' = (\sigma_T - P) - (\chi \cdot P_c) \quad (5.5\text{-}3b)$$

PFT(2) S_r^m: Fraction of small pores for calculation of factor χ for weighting the capillary pressure contribution to effective stress:

$$\chi = \frac{S_l - S_r^m}{1 - S_r^m} + \frac{1}{n} ln \left[1 + exp\left(-n \frac{S_l - S_r^m}{1 - S_r^m}\right)\right] \quad (5.5\text{-}4)$$

where $n = \text{abs}(PFT(7))$ if $PFT(7) < 0$, else $n = 20$.
For $S_r^m = 0$, $\chi = S_l$, i.e., the model is identical to that of Bishop [*Nuth and Laloui*, 2008].

POR(NMAT)	ϕ_0:	Porosity at zero effective stress [-]
PFT(3)	ϕ_r:	Residual porosity at high effective stress [-]
PFT(4)	α:	Stiffness parameter [MPa^{-1}] ($\alpha = 1$: stiff; $\alpha < 1$: soft)
PFT(5)	β:	Exponent in porosity-permeability models [-]; coefficient
PFT(6)	k_{min}:	Minimum permeability [m^2] or γ [-] in Eq. (5.5-1c) if $\lvert$ *IPDPP* $\rvert = 3$
PFT(7)	k_{max}:	Maximum permeability [m^2] or n [-] in Eq. (5.5-4) if *PFT(7)* > 1.0

Figure 17 shows an example of a ROCKS block invoking pressure-dependent porosity according to *Pereira* et al. [2010] combined with the porosity-permeability model according to *Rutqvist et al*. [2002].

```
ROCKS----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
POMED     1      2650.      0.125     1.E-20     1.E-20     1.E-21         2.1      1000.
-2.000E-09
   +2         -15.0E6         0.6      0.100         0.8       20.0    1.0E-23    1.0E-18
```

Figure 17. Specifying pressure-dependent porosity and permeability.

5.6 Material Regions (Block ROCKS)

The option described in this section allows one to assign a material to all elements within a geometric region. The region option can only be used if element coordinates are provided in columns 51–80 in block ELEME. The material assignment will be made either for all elements within the user-specified region, or only elements with specific material names within that region. The region is defined by the location of its center and the extent. The shape can be an ellipsoid, a rectangular box or cube, a cylinder or truncated cone, or a polygonal prism.

The location and size of this region is parameterized and can thus be subjected to parameter estimation by inverse modeling (e.g., for sensitivity analysis on fault location). For the system response to be a smooth function of the location and size of the permeability region, the region must comprise multiple elements, with a smooth function describing the transition of some of its properties between the region and the background domain as a function of distance from the center of the region; a sharp contrast between the two domains may also be requested (see variable *IREGINFF* below). Currently, such a smooth transition is only applied to permeability and porosity.

Regions are defined as follows. The box region includes all elements *i* with coordinates X_i, Y_i, and Z_i that lie within a rectangular box, i.e.,

$$\begin{aligned} d_X &= |X_i - X_c| \le L_X \\ d_Y &= |Y_i - Y_c| \le L_Y \\ d_Z &= |Z_i - Z_c| \le L_Z \end{aligned} \qquad \text{(5.6-1a)}$$

The ellipsoidal region includes all elements i with coordinates X_i, Y_i, and Z_i that satisfy the equation

$$d^2 = \left(\frac{X_i - X_c}{L_X} \right)^2 + \left(\frac{Y_i - Y_c}{L_Y} \right)^2 + \left(\frac{Z_i - Z_c}{L_Z} \right)^2 < 1 \qquad (5.6\text{-}1b)$$

Here, X_i, Y_i, and Z_i are the center coordinates of the box or ellipsoid, and L_X, L_Y, and L_Z are the three half-lengths of the box or the three semi-axes of the ellipsoid.

A cylindrical region is defined by the starting and end coordinates of its axis, and a radius.

The irregular base of the prism is defined by an arbitrary number of points of a polygon.

The ellipsoid, box and cube can be rotated. However, the axis of the irregular prism must be aligned with one of the coordinate axes.

The permeabilities and porosities of the elements within (or, alternatively, outside) the region will be calculated as a simple weighted average of the respective background values k_0 and ϕ_0 (i.e., the values in the ROCKS block assigned to the element in block ELEME) and those specified for the particular region, i.e., k_{reg} and ϕ_{reg}:

$$k_i = \omega \cdot k_{reg} + (1-\omega) \cdot k_0 \qquad (5.6\text{-}2a)$$

$$\phi_i = \omega \cdot \phi_{reg} + (1-\omega) \cdot \phi_0 \qquad (5.6\text{-}2b)$$

where ω_i is one of the following "influence functions":

$$\omega = 1 \qquad (5.6\text{-}3a)$$

$$\omega(d) = 1 - d \qquad (5.6\text{-}3b)$$

$$\omega(d) = \left(1 - \left(\frac{3}{2} d - \frac{1}{2} d^3 \right) \right) \qquad (5.6\text{-}3c)$$

$$\omega(d) = a^{-d} \qquad (5.6\text{-}3d)$$

$$\omega(d) = \min(1, (1 - d)/a) \qquad (5.6\text{-}3e)$$

Here, d is a normalized distance from the center of the region (for boxes and ellipsoids) or from the axis of the cylinder. In Eqs. (5.6-3d) and (5.6-3e), a is a user-specified parameter, provided at the end of the region-definition parameters (see Table 7). The effect of the influence function is that the farther away the element is from the region's center, the lower is the weight assigned to the region-specific permeability and porosity, i.e., the properties near the edges of the region approach those of the background material. Note that if Eq. (5.6-3a) is chosen, each element within the region will have the same property value. However, changing the geometry of the region will not lead to a smooth, differentiable change of properties; consequently, this parameter value cannot be used identifying the permeability structure using a gradient based algorithm.

The option is invoked by selecting "XYZ" as the first three characters or "&" as the first character of the material name (see variable *MAT* in block ROCKS.1). If a region is requested, the following parameters are read in format (16I5), starting on the next line:

IREGGEOM Defines geometry of the region; set negative to also provide rotation angles (see Table 7) (not available for prism).
1 : Box
2 : Ellipsoid
3 : Cylinder
4 : Cube
5 : Polygonal prism (aligned with one of the coordinate axes)
6 : Right, circular, truncated cone
7 : Cylinder, cut by an angled plane at one end

IREGINFF Defines influence function. If negative, the complementary region is selected (i.e., all elements *outside* the defined geometrical region).
0 or 9: Constant (5.6-3a)
1 : Linear (5.6-3b)
2 : Spherical (5.6-3c)
3 : Exponential (5.6-3d)
4 : Constant-linear (5.6-3e)
5 : Inverse distance weighting

NREGROCK Number of materials listed in following slots. If positive, only elements belonging to the listed materials will be replaced by the region material; if negative, all elements within the region except those belonging to the listed materials will be replaced by the region material. Set to zero if all elements within the region should be assigned to the region material.

IREGROCK(I, i=1,ABS(NREGROCK))
Material sequence numbers of elements to be included (positive *NREGROCK*) or excluded (negative *NREGROCK*) from region.

NPREGPOLY (only if *IPREGGEOM* = 5): Number of polygon points, *NP*

IJKPRISMAXIS (only if *IREGGEOM* = 5): Alignment of polygonal prism axis
1 : prism axis aligned with *X*-axis
2 : prism axis aligned with *Y*-axis
3 : prism axis aligned with *Z*-axis

The following parameters are read in free format:

XREGION(*i*) (see Table 7)

XREGINFF Parameter *a* (only if |*IREGINFF*| = 3 or 4)

If the region is not aligned with the coordinate axes, set *IREGGEOM* negative and provide three correction angles (azimuth, dip, and plunge). A polygonal prism must be aligned with one of the prism axes. Figure 18 shows an input file that uses a rotated ellipsoidal region to

represent an inclined fault. In addition, a prismatic engineered structure is generated by indicating the start and end of the prism axis in *X*-direction, and by defining the prism base using an 11-point polygon. Figure 19 shows how region definitions are reported in the TOUGH2 output file. The resulting permeability distribution is visualized in Figure 20. Note that the order in which regions are defined affects the final property distribution if regions overlap or intersect each other (later regions modifying the previously defined regions). The parameters `XREGION(i)` can be varied through the iTOUGH2 command `>> REGION` and thus be estimated using inverse modeling. For example, the location and extent of the fault can be varied.

Table 7. Geometrical Parameters Defining Region

`IREGGEOM`	`XREGION(i)`								
	1	2	3	4	5	6	7	8	9
1 (box)	X_{min}	Y_{min}	Z_{min}	X_{max}	Y_{max}	Z_{max}	*azimuth*	*dip*	*plunge*
2 (ellipsoid)	X_c	Y_c	Z_c	L_X	L_Y	L_Z	*azimuth*	*dip*	*plunge*
3 (cylinder)	X_B	Y_B	Z_B	X_E	Y_E	Z_E	R	-	-
4 (cube)	X_c	Y_c	Z_c	L_X	L_Y	L_Z	*azimuth*	*dip*	*plunge*
5 (prism)	$Axis_{min}$[@]	$Axis_{max}$[@]	$C_{1,1}$[%]	$C_{1,2}$[%]	…[%]	…[%]	$C_{NP,1}$[%]	$C_{NP,2}$[%]	
6 (cone)	X_B	Y_B	Z_B	X_E	Y_E	Z_E	R_B	R_E	-
7 (cut cylinder)	X_B	Y_B	Z_B	X_E	Y_E	Z_E	R	*icut*[$]	*angle*[&]

[@] Minimum and maximum coordinate of prism axis, i.e., location of prism bases; coordinate variable given by `IJKPRISMAXIS`

[%] Coordinates for `NP` polygon points (clockwise or anti-clockwise); two coordinates per point: (*Y*,*Z*) for `IJKPRISMAXIS` = 1; (*X*,*Z*) for `IJKPRISMAXIS` = 2; (*X*,*Y*) for `IJKPRISMAXIS` = 3

[$] +/–: start/end face of cylinder is cut
|*icut*|=1: Cutting plane rotates around axis perpendicular to cylinder axis in *Y*-*Z* plane
|*icut*|=2: Cutting plane rotates around axis perpendicular to cylinder axis in *X*-*Z* plane
|*icut*|=3: Cutting plane rotates around axis perpendicular to cylinder axis in *X*-*Y* plane
For a cylinder axis at arbitrary angle in Y-Z, |*icut*|=1 only
For a cylinder axis at arbitrary angle in X-Z, |*icut*|=2 only
For a cylinder axis at arbitrary angle in X-Y, |*icut*|=3 only
For a cylinder axis parallel to the X axis (in X-Y plane and X-Z plane), |*icut*|=3 or |*icut*|=2
For a cylinder axis parallel to the Y axis (in X-Y plane and Y-Z plane), |*icut*|=3 or |*icut*|=1
For a cylinder axis parallel to the Z axis (in X-Z plane and Y-Z plane,) |*icut*|=2 or |*icut*|=1

[&] Rotation *angle* of the cutting plane, measured counter-clockwise from the cylinder axis within the chosen plane. The rotation axis of the cutting plane is perpendicular to and must be within the same plane as the cylinder axis.

```
Ellipsoidal, prismatic, and cylindrical property regions
DIMEN----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
         5
MESHM----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
XYZ

NX      100       0.2
NY      100       0.2
NZ      100       0.2


ROCKS----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
GRANI     0    2670.0      0.01  1.00E-16                                800.0
&FRAC     0    2670.0      0.40  1.00E-10                                800.0
   -2     2    0
       10.0      11.9      -8.2
        1.0      10.0      22.0
      130.0     -30.0      65.0
XYZBE     0    2670.0      0.40  1.00E-20                                800.0
    5     4    0   11     1
       -1.0      17.2
        5.0     -15.0
        5.0     -10.0
        7.0      -8.0
       10.0      -7.0
       13.0      -8.0
       15.0     -10.0
       15.0     -15.0
       11.0     -15.0
       11.0     -12.0
        9.0     -12.0
        9.0     -15.0
      0.98
XYZHE     0    4000.0      0.01  1.00E-09                                800.0
    3     0    0
       -1.0      10.0     -10.5
        2.5      10.0     -10.5
        1.5

OUTPU----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
TECPLOT
3
COORD
PERMEABILITY          1
POROSITY

START----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
PARAM----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
 8 3   1       11000009000010004000050000
        0.                  1.0
     1.E-5
 1.1000000000000E+05
MOMOP----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
                        1
ENDCY----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
```

Figure 18. TOUGH2 input file for assigning ellipsoidal, prismatic, and cylindrical regions.

```
REGIONS: PERMEABILITY (K), SINK/SOURCE (S), OBSERVATION (O)
============================================================
REGION SKO  SOURCE    GEOMETRY     INFLUENCE  X-(S|C)   Y-(S|C)   Z-(S|C) X-END|DX/2 Y-END|DY/2 Z-END|DZ/2 RADIUS
           MATERIAL               FUNCTION
-------------------------------------------------------------------------------------------------------------------

     1  K   &FRAC    ELLIPSOID    SPHERICAL   10.000    11.900    -8.200      1.000     10.000     22.000
          AZIMUTH          DIP        PLUNGE  130.000   -30.000    65.000

     2  K   XYZBE        PRISM  CONST.-LIN.
           MIN AND MAX AXIS COORDINATES      -1.000    17.200
           POLYGON POINT                      5.000   -15.000
           POLYGON POINT                      5.000   -10.000
           POLYGON POINT                      7.000    -8.000
           POLYGON POINT                     10.000    -7.000
           POLYGON POINT                     13.000    -8.000
           POLYGON POINT                     15.000   -10.000
           POLYGON POINT                     15.000   -15.000
           POLYGON POINT                     11.000   -15.000
           POLYGON POINT                     11.000   -12.000
           POLYGON POINT                      9.000   -12.000
           POLYGON POINT                      9.000   -15.000
           INFLUENCE FUNCTION PARAMETER       0.980

     3  K   XYZHE     CYLINDER      CONSTANT   -1.000    10.000   -10.500      2.500     10.000    -10.500  1.500
-------------------------------------------------------------------------------------------------------------------
```

Figure 19. Excerpt from TOUGH2 output file, documenting region definitions.

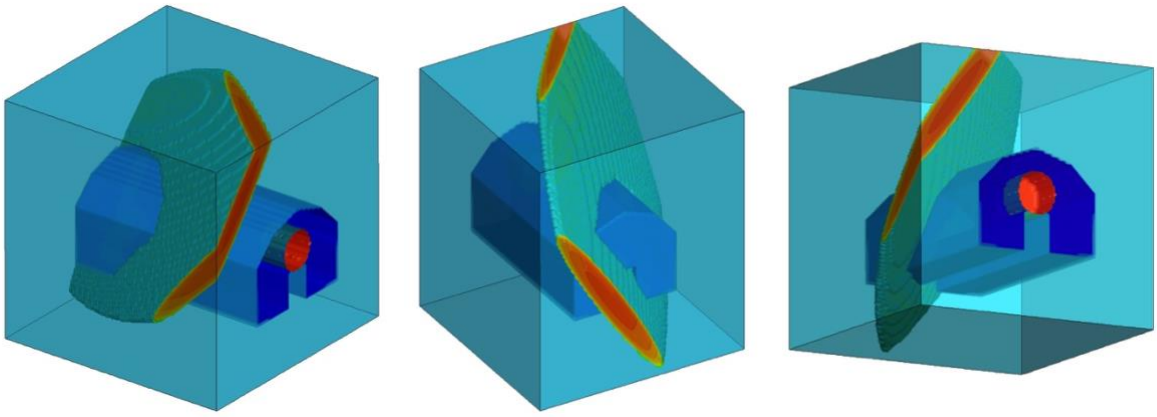

Figure 20. Permeability distribution generated by region assignment using the input file shown in Figure 18.

5.7 Time-Dependent Material Properties (Block ROCKS)

By default, the material properties specified in block ROCKS remain constant during a simulation. A notable exception is porosity, which may change as a function of pressure and/or temperature provided a non-zero pore compressibility and/or expansivity be specified.

All properties (except porosity) specified in block ROCKS can be made time-dependent. This option assumes that the material properties are not dynamically calculated as a function of the solution vector (i.e., the primary variables), but instead are known beforehand and can thus be prescribed as a function of time. Time-dependent property values must be provided in tabular form on separate files, one file for each material type for which at least one property is time-dependent. The name of the file needs to be given at the end of record ROCKS.1, at any position with a column number greater than 80. An example is shown in Figure 21, where the names of the files containing time-dependent material property values are given in Columns 81–91. Note that the properties of material LAY02 do not change with time.

Figure 22 shows the content of the user-supplied file with time-dependent material properties. The file contains a single header line identifying which input parameters are considered time-dependent. The heading must be identical to the variable names shown for the parameters in block ROCKS in Appendix E of the TOUGH2 manual [*Pruess et al.*, 1999], i.e., *DROK*, *PER(i)* (or *PERM*, if all three permeabilities are updated to the same value), *CWET*, *SPHT*, *COM*, *EXPAN*, *CDRY*, *TORTX*, *GK*, *FOC*, *XKD(i)*, *RP(i)*, and *CP(i)*. In addition, parameters *PFT(i)* and *CFT(i)* (see *Magnúsdóttir and Finsterle* [2015]) can be made time-dependent. Index *i* in the variable names has to be replaced by the appropriate integer to identify the desired array element. The header line must also contain the keyword TIME to identify the column holding time (in seconds). The first time must be equal or earlier than the simulation starting time TSTART, and the last time must be equal or later than the end time of the simulation.

Following the header line, an arbitrary number of lines with time-dependent property values can be given. In the example shown in Figure 22, all properties remain constant for the first day, after which permeability is decreasing (variable *PERM* is used to first change the permeabilities in all three directions, and *PER(3)* is used to then overwrite the permeability in the third direction), pore compressibility is increasing (e.g., due to some chemically or thermally induced degradation of rock strength), and associated changes in the pore size distribution and capillary-strength parameters of the van Genuchten (1980) curves.

As a simulation proceeds through time, property values are assigned after each time step to values that are linearly interpolated between the points provided in the table. Note that different time-dependent variables may be specified for different materials.

If a property is declared time-dependent, it is a user-specified quantity and thus cannot be estimated or updated in iTOUGH2 (potential approaches suitable for estimating time-dependent properties include the use of iTOUGH2-PEST [*Finsterle*, 2010] the RESTART option of iTOUGH2, or defining and estimating appropriate user-specified parameters through subroutine USERPAR).

```
ROCKS----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8----*----9--
LAY01    2    2680.0      0.39  1.00E-12  1.00E-12  1.00E-13       2.0    1000.0 MAT1fT.dat
     1.E-9
   7            0.80      0.20      1.00      0.00
   7            0.80      0.15  6.40E-04  1.00E+08      1.00
LAY02    2    2680.0      0.22  2.62E-13  2.62E-13  2.62E-13       2.0    1000.0
     1.E-9
   7            0.90      0.25      1.00      0.00
   7            0.90      0.20  6.40E-04  1.00E+08      1.00
LAY03    2    2680.0      0.25  1.00E-13  1.00E-13  1.00E-14       2.0    1000.0 MAT3fT.dat
     1.E-9
   7            0.66      0.10      1.00      0.00
   7            0.66      0.08  7.38E-04  1.00E+08      1.00
```

Figure 21. ROCKS block with names of files containing time-dependent material properties.

```
      TIME       PERM     PER(3)        COM      RP(1)        CP(3)
         0  1.00E-12  1.00E-13  1.00e-09       0.80       6.40E-04
    864000  1.00E-12  1.00E-13  1.00e-09       0.80       6.40E-04
   8640000  1.00E-14  1.00E-15  5.00e-09       0.60       6.40E-05
```

Figure 22. Example file with time-dependent material properties.

5.8 Anisotropic Properties (Block ROCKS)

Three material- or element-specific absolute permeability values, *PER(1)*, *PER(2)*, and *PER(3)*, can be specified using the methods described in Sections 5.1, 5.2, and 5.3. For each connection, one of these three values can be selected through index *ISOT* in block CONNE. If the connections are aligned with the principal axes of the permeability tensor, this scheme can be used to represent anisotropy in permeability. The same concept can also be applied to other potentially anisotropic properties, including thermal conductivity, *CWET*, and tortuosity, *TORTX*. In addition, anisotropic parameters of the relative permeability (*RP(i)*, *i*=1…7) and capillary pressure (*CP(i)*, *i*=1…7) functions can be specified (see below); however, this feature is only available for two-phase (not three-phase) EOS-modules.

To specify material-specific anisotropic values for *CWET* and *TORTX*, provide a negative value for the variable for which three (instead of only one) value will be given; then supply these three values on the following line in format (3E10.4) (note that *NAD* should not be increased despite this additional line in block ROCKS).

An example input file is shown in Figure 23. The upper half of the model cube has anisotropic thermal conductivities and anisotropic tortuosity factors, whereas the lower half has isotropic properties. Heat and dissolved air (used as a tracer) are injected in the center of the cube. Isosurfaces of the resulting temperature and concentration distributions are shown in Figure 24.

Note that the input file shown in Figure 23 features a number of other useful enhancements, including: (a) Generation of 6-character element names in block MESHM (see *MOP2(2)* in Table 3); (b) overwriting of element geometry (see block ELEM2 described in Section 4.2); (c) shifting of element coordinates (see Section 4.1); and (d) assignment of material &DOWN to a region defined by coordinates (Section 5.4).

Heterogeneous anisotropic property values (for permeability, thermal conductivity, and tortuosity) can also be specified using the approach described in Section 5.3. The anisotropic properties provided in block ELEME will overwrite those in block ROCKS.

```
samANISO: Material-specific anisotropic thermal conductivity and tortuosity
MULTI----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
    2    3    2    8
DIMEN----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
         2
ROCKS----1----*----2----*----3----*-TORT----*----5----*----6----*-CWET----*----8
UPPER    1     2650.     .3000 1.000E-12 1.000E-12 1.000E-12     -1.50      800.
      0.50      1.50      3.00
 1.000E-07                          -0.50
      1.00      0.50      0.20
&DOWN    1     2650.     .3000 1.000E-12 1.000E-12 1.000E-12      1.50      800.
    1    0
     -0.01     -0.01    -10.01
     10.01     10.01     -5.01
 1.000E-07                           0.50

RPCAP----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
    3
    8
PARAM----1--MOP:123456789012345678901234----*----5----*----6----*----7----*----8
   2 100      10100000900000000400005000
 0.000E+00           1.000E+01

 100000.000000000000    0.00000000000000  20.000000000000000
MOMOP----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
 6
DIFFU----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
     1.e-6     1.e-6
     1.e-6     1.e-6
MESHM6---1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
XYZ
       0.0      -0.1     -0.1       0.1
NX      51       0.2
NY      51       0.2
NZ      51       0.2


ELEM2----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
262626             10.8000E+620.0000E+00           5.000     5.000     -5.000

CONN2

START----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
INCON----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
262626
 100000.000000000000 8.0000000000000E-06  80.000000000000000

GENER----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8

ENDCY----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
```

Figure 23. ROCKS block with material-specific anisotropic thermal conductivity and tortuosity factors.

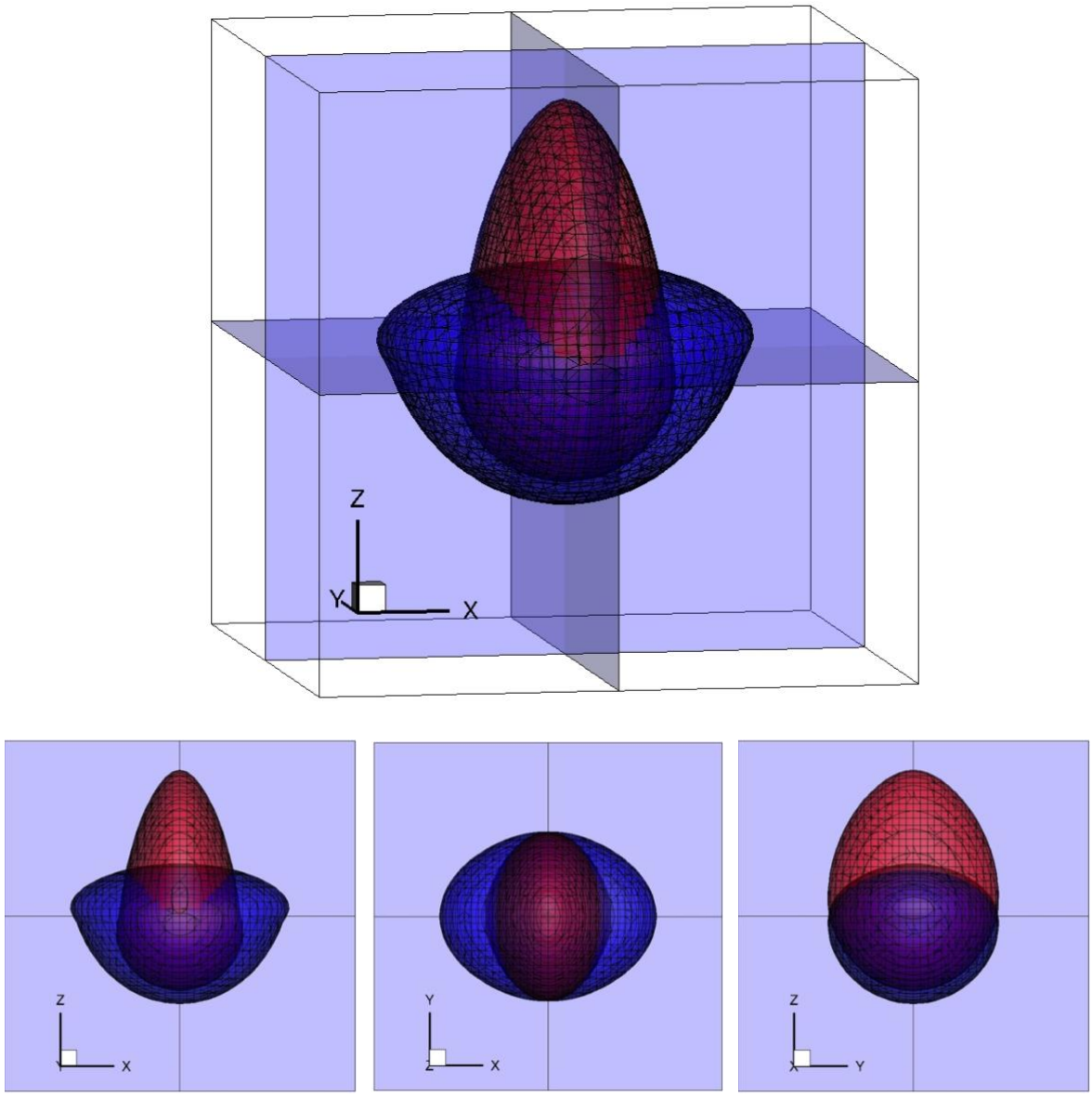

Figure 24. Distribution of temperature (red) and tracer concentration (blue) obtained in a uniform material with isotropic and anisotropic properties in the lower and upper half space, respectively.

To specify anisotropy in relative permeabilities and capillary pressures, three (instead of one) sets of parameters *RP* and *CP* need to be provided, one for each direction *ISOT*. Reading of three instead of one set of parameters is indicated by adding 300 to the value of *IRP* and *ICP*, respectively. Note that the relative permeability parameters for *ISOT* = 1 are provided in slots *RP(1)–RP(7)*, those for *ISOT* = 2 in slots *RP(8)–RP(14)*, and those for *ISOT* = 3 in slots *RP(15)–RP(21)* (and analogous for the capillary pressure curve parameters *CP*). The orientation-specific relative permeabilities are stored in the *PAR* array ins slots 2, 9, and 10, and the orientation-specific capillary pressures are stored in slots 6, 11, and 12. This requires an increase in the dimension of the *PAR* array, which needs to be

indicated using a new parameter *NBRPCP* in Columns 26–30 of block MULTI. An example of an input file is shown in Figure 25. Note that the FOFT block is used to output the orientation-specific relative permeability can capillary pressures (selected in block OUTPU) for the injection element (see Figure 26). Anisotropic permeability and capillary pressure parameters can also be specified on an element-by-element basis (see Section 5.3).

```
Anisotropic relative permeability and capillary pressure functions
MULTI----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
    2    3    2    6         6
ROCKS----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
DIRTY    2     2650.       .20    1.E-12    1.E-12    1.E-12       2.1     1000.

  303           0.20       0.05
                0.30       0.15
                0.40       0.25
  304           0.20
                0.30
                0.40

PARAM----1----*-123456789012345678901234----*----5----*----6----*----7----*----8
 8 3 100       11000009000000000400005000
            1.0E+03   1.0E+01                    9.8066e+00
     1.E-5                                          1.E-8
            1.0E+06               10.50               20.00
MESHM6---1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
XYZ
       0.0
NX       3       1.0
NY       3       2.0
NZ       3       5.0


START----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
GENER6---1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
 2 2 2INJ 1                        WATE     1.0E00  1600.0E3

OUTPU----1----*--PHASE--SLOT-3----*----4----*----5----*----6----*----7----*----8
CSV
9
RELATIVE            1    2    krgx
RELATIVE            2    2    krlx
SECONDARY           1    9    krgy
SECONDARY           2    9    krly
SECONDARY           1   10    krgz
SECONDARY           2   10    krlz
CAPILLARY           2    6    Pcx
SECONDARY           2   11    Pcy
SECONDARY           2   12    Pcz

FOFT ----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
 20202

ENDCY----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
```

Figure 25. ROCKS block with material-specific anisotropic parameters of the relative permeability and capillary pressure functions.

TIME [sec]	REL1_GAS	REL1_LIQ	REL2_GAS	REL2_LIQ	REL3_GAS	REL3_LIQ	CP1_LIQ	CP2_LIQ	CP3_LIQ
0.1000e+02	0.2937E+00	0.2737E-01	0.3386E+00	0.1932E-01	0.4456E+00	0.8119E-02	-0.3398E+04	-0.1360E+05	-0.6932E+05
0.3000e+02	0.2767E+00	0.3118E-01	0.3138E+00	0.2342E-01	0.4013E+00	0.1172E-01	-0.2860E+04	-0.1188E+05	-0.5613E+05
0.7000e+02	0.2445E+00	0.3998E-01	0.2672E+00	0.3354E-01	0.3197E+00	0.2238E-01	-0.1907E+04	-0.9104E+04	-0.3855E+05
0.1500e+03	0.1870E+00	0.6273E-01	0.1868E+00	0.6282E-01	0.1864E+00	0.6301E-01	-0.9778E+02	-0.5311E+04	-0.2055E+05
0.3100e+03	0.1023E+00	0.1296E+00	0.7860E-01	0.1641E+00	0.3990E-01	0.2606E+00	-0.9778E+02	-0.1281E+04	-0.7497E+04
0.6300e+03	0.3581E-01	0.2765E+00	0.1298E-01	0.4242E+00	0.1065E-04	0.9317E+00	-0.9778E+02	-0.9778E+02	-0.1541E+04
0.9500e+03	0.1749E-01	0.3817E+00	0.2513E-02	0.6274E+00	0.0000E+00	0.1000E+01	-0.9778E+02	-0.9778E+02	-0.9778E+02
0.1000e+04	0.1561E-01	0.3981E+00	0.1825E-02	0.6600E+00	0.0000E+00	0.1000E+01	-0.9778E+02	-0.9778E+02	-0.9778E+02

Figure 26. File `FOFT` shows orientation-specific relative permeabilities and capillary pressures for the injection element as a function of time.

5.9 Geostatistics (Block `GSLIB`)

Spatially correlated permeability fields can be generated internally using methods of the geostatistical library GSLIB [*Deutsch and Journel*, 1992]. A special user's guide [*Finsterle and Kowalsky*, 2007] describes the approach and user options.

5.10 Material-Specific Thermal Conductivity Functions (Block `ROCKS`)

The interpolation formula used to calculated thermal conductivity as a function of saturation is selected through *MOP(10)*, where K_{wet} and K_{dry} are variables *CWET* and *CDRY* in Blocks `ROCKS.1` and `RCOKS1.1`, respectively. The model chosen through *MOP(10)* applies to all materials.

Material-specific thermal-conductivity models can be selected in block `ROCKS.1` by a non-negative integer in Column 6 immediately after the material name. If Column 6 is blank for a given material, the default function selected through *MOP(10)* is used. The available functions are summarized in Table 8.

Table 8. Models to calculate thermal conductivity as a function of saturation

MOP(10) or *MOPR(1)*	Interpolation function for unsaturated thermal conductivity	Comment
0	$K(S_l) = K_{dry} + \sqrt{S_l} \cdot \left(K_{wet} - K_{dry}\right)$	Square-root (default)
1	$K(S_l) = K_{dry} + S_l \cdot \left(K_{wet} - K_{dry}\right)$	Linear
2	$K(S_l) = a + bT + cS_l + d\phi$	Coefficients after Rautman (1995)
3	$K(S_l) = K_{dry}^{(1-\phi)} + K_{wat}^{(S_l \cdot \phi)} + K_{ice}^{(1-S_l)\phi}$	For freezing model
4	$K(S_l) = K_{dry} \cdot \left(1 - \sqrt{S_l}\right) + K_{dry} \cdot \sqrt{S_l}$	Papafotiou et al. (2018)
5	$K(S_l) = K_{dry}^{(1-S_l)} \cdot K_{wet}^{S_l}$	BGR (pers. comm.)

6 Boundary Conditions and Sink/Source Terms

6.1 Time-Dependent Dirichlet Boundary Conditions (Block TIMBC)

Time-dependent Dirichlet boundary conditions can be read from the input file following keyword TIMBC, or from the file given on the line following keyword "TIMBC F".

FILENAME	:	Filename with boundary condition data (only if "TIMBC F"); make sure the file is available in the local directory. All the information below will be provided in that file.
NTPTAB	:	Number of elements or materials with time-dependent boundary conditions Repeat the following entries *NTPTAB* times
NBCP, *NBCPV*	:	Number of times and identification number of primary variable
BCELM	:	Name of boundary element or material (start in Column 1)
TIMEBCV, *PGBCEL*	:	Time and value of primary variable *NBCPV* at boundary element or material *BCELM*; repeat this entry *NBCP* times

All values are read in free format. Boundary values will be linearly interpolated between table entries. Block TIMBC must be specified *before* the optional block LOAD (see Section 11). An example is given in Figure 27.

```
TIMBC----1----*----2----*----3----*----4----*----5----*----6----*----7
2
3  1
BOUND       pressure
    0.0     1.0E5
86400.0     1.1E5
 1.0E50     1.1E5
3  2
  T 1       sat+10
    0.0     10.1
  1.0E7     10.9
 1.0E50     10.9
```

Figure 27. TOUGH2 input file for assigning time-dependent Dirichlet boundary conditions.

6.2 Free-Drainage Boundary Condition (Material *DRAIN*)

A free drainage boundary condition for liquid flow can be implemented, in which gravity is the only driving force, i.e., (capillary) pressure gradients are ignored across an interface to the boundary gridblock. This type of boundary condition comes into effect at each connection in which one of the gridblocks belongs to rock type DRAIN.

6.3 User-Specified Boundary Conditions (Subroutine *USERBC*)

In TOUGH2, Neumann boundary conditions are specified by introducing sinks and sources in block GENER. Dirichlet boundary conditions can be implemented by assigning very large volumes to grid blocks adjacent to the boundary so that the thermodynamic conditions in those elements do not change from fluid and heat exchange with finite-size grid blocks in the model domain.

Prescribed, but time-varying boundary conditions can be implemented by specifying appropriate (large) sinks and sources in grid blocks having a very large volume or by using keyword TIMBC as described in Section 6.1. Moreover, for simple step changes, iTOUGH2 offers an alternative option (see command >> RESTART TIME). Another possibility is described in this section. The user can provide values of the primary variables for selected elements as a function of time. The function has to be programmed into subroutine USERBC, which can be found in file *it2user.f*. In subroutine USERBC, the user has the possibility to provide the value of one or more primary variables for selected elements as a function of time. For example, these values can be calculated internally or read from a file. The header of subroutine USERBC is shown in Figure 28. The element number *N* or element name *CELEM* can be used to identify the boundary grid block. The user is supposed to return a value for one or several of the primary variables through array *X*. In the example given below, a table of time versus pressure data is read from file *atm_pres.dat* and assigned to element 'ATM 0' using the linear interpolation function INTERP1. Note that either the full path to file *atm_pres.dat* must be given, or the file must be copied to the temporary directory using option -fi filename.

Subroutine USERBC is called only if *MOP(22)* is either 1 or 2. If *MOP(22)* is 2, the EOS module is called after completion of a time step to ensure that the user-specified thermodynamic conditions are updated. If *MOP(22)* is 1, subroutine USERBC is called, but no additional call to subroutine EOS is made. This allows one to make time-dependent changes to TOUGH2 variables that are not primary variables, and that do not require a recalculation of the thermodynamic state.

```
************************************************************************
      SUBROUTINE USERBC(N,CELEM,VOLUME,TIME,X)
************************************************************************
*  User specified boundary condition                                   *
*  Set MOP(22).GE.1                                                    *
*      MOP(22)=1 don't call EOS                                        *
*      MOP(22)=2 call EOS                                              *
*  Return user specified boundary condition (vector X)                 *
*  for element CELEM at time TIME and/or change the VOLUME of CELEM    *
************************************************************************

      CHARACTER CELEM*5
      PARAMETER (MDATA=5000)
      DIMENSION X(*),DTIME(MDATA),DVALUE(MDATA)
      SAVE DTIME,DVALUE,IREAD

      IF (CELEM.EQ.'ATM 0') THEN
C --- Read table from a file
         IF (IREAD.EQ.0) THEN
            IREAD=IREAD+1
            OPEN(UNIT=39,FILE='atm_pres.dat',STATUS='OLD')
            I=0
 1001       CONTINUE
            I=I+1
            READ(39,*,END=1002) DTIME(I),DVALUE(I)
            GOTO 1001
 1002       CONTINUE
            NDATA=I-1
            CLOSE(39)
         ENDIF
         CALL INTERP1(TIME,X(1),DTIME,DVALUE,NDATA)
      ENDIF
```

Figure 28. Subroutine *USERBC* for specifying time-dependent boundary conditions.

6.4 Tabular Input of Time-Dependent Rates (Block GENER)

In addition to the standard input format, time-dependent generation rates (i.e., if *LTAB* > 1 in block GENER.1) can be provided as a free-format table with time in the first column, injection or production rate in the second column, and (if *ITAB* is not left blank) specific enthalpy in the third column. The number of table rows is given by *LTAB*. If *LTAB* = 'COUNT', the number of table entries is automatically counted. The tabular format is chosen by providing the character "T" or "D" in Column 7 after keyword GENER. Moreover, time and rate conversion factors can be given in Columns 11–20 and 21–30. If character "D" is specified in Column 7, time can be given in (any) date format; it will be converted to seconds (relative to the first date given; the first and last dates in a time series can be set as "-infinity" and "infinity"; negative times in regular time units may also be specified without affecting the reference date). These conversion factors only apply to sinks/source with time-dependent generation rates (i.e., constant rates given in Columns 41–50 of block GENER.1 are not affected). Figure 29 shows an example, in which time-dependent water injection rates (in m^3/hour) are given as a four-entry table, with time given in hours. The options discussed in this section are only available if sinks/sources are given directly in the TOUGH2 input deck. The external file *GENER* has to be provided in the

standard format. These options can only be used if the GENER block is provided in the TOUGH2 input file; the external GENER file must be provided in standard TOUGH format.

```
MOMOP----1----*----2----*----3----*----4----*----5----*----6----*----7
  1
----*----1----GTFACT----GRFACT----*----4----*----5----*----6----*----7
GENER T       3600.0 2.7778E-1 ---*----4----*----5----*----6----*----7
A11 1INJ 1               COUNT     WATE
  0.00  1.0
  6.00  1.0
  6.01  0.0
1.0e50  0.0
```

Figure 29. TOUGH2 input format for specifying time-dependent generation rates in tabular format and using conversion factors for time and rate.

6.5 Logarithmic Interpolation of Time-Dependent Rates (Block GENER)

For time-dependent generation rates that approximately follow a power-law function or an exponential decay curve (e.g., heat release from decaying radionuclides), it may be advantageous to perform a logarithmic (rather than linear) interpolation between the data points provided in block GENER (with *LTAB* > 1).

If all sinks/sources specified in block GENER shall be logarithmically interpolated, set MOP(12) = 3. To select individual sinks/sources for logarithmic interpolation, the rates must be provided in tabular form (see Section 6.4), and variable *LTAB* must contain the string 'LOG' between Columns 26 and 30. The number of data points will be counted automatically. Figure 30 shows an example of specifying decay heat release from radioactive waste using just a few data points, where the values used at intermediate simulation times are calculated by logarithmic interpolation.

```
GENER T -1----*----2----*----3----*----4----*----5----*----6----*----7
A11 1WASTE                 LOG     HEAT
             0.0      100000.0
    3.155760E+07       32587.6
    3.155760E+08        7241.7
    3.155760E+09        1255.2
    3.155760E+10         173.8
    3.155760E+11          53.1
    3.155760E+12           4.3
    3.155760E+13           1.4
          1.0E16           0.0
```

Figure 30. TOUGH2 input format for providing time-dependent generation rates with logarithmic interpolation between the data points.

6.6 Cycling Generation Rate Sequences (Block GENER)

In standard TOUGH2, the first time in any GENER block that defines time-dependent rates must be smaller than the starting time of the simulation, and the last time must be greater than the end time of the simulation. In iTOUGH2, starting and end times in the GENER block do not need to fulfill this requirement; instead, the following rates will be taken:

- For times smaller than the first time in GENER, the rate at the first time is taken.
- If the simulation time exceeds the last tabulated time (i.e., that with index *LTAB*), the sequence of rates specified in *F2(i)* (and *F3(i)*), $i = 1, \ldots, LTAB$, is repeated, allowing to simulate repeated cycles by specifying a single cycle.

Figure 31shows an example in which, starting after 100 days, a 20-day injection-production cycle (with three-day waiting periods in between) is repeated until the end of the simulation. This feature is only invoked if $MOP(12) \leq 1$.

```
0---*----1----*----2----*----3----*----4----*----5----*----6----*----7
GENER T      86400.0
A5612INJ 1                    5     WATEe       5.0
  100.0     0.0  1E6
  100.05    5.0  1E6
  107.00    5.0  1E6
  107.05    0.0  1E6
  120.00    0.0  1E6
A7612PRO 1                    6     MASS       -4.0
   100.0    0.0
   110.00   0.0
   110.05  -5.0
   117.00  -5.0
   117.05   0.0
   120.00   0.0
```

Figure 31. TOUGH2 input format for specifying a cycling injection-production schedule.

6.7 Sinusoidal Generation (Block GENER)

Sinusoidal generation rates of the form

$$Q(t) = Q_0 + Q_A \cdot sin\left(2\pi \cdot \frac{t_0+t}{t_p}\right) \qquad (6.6\text{-}1)$$

where t is simulation time, t_0 is the time lag, t_p is the time period (1/frequency), Q_0 is the reference rate, and Q_A is the rate amplitude. Sinusoidal generation rates can be modeled by specifying the following source term in block GENER:

(1) Define source code name "SIN**". If a material-related source is specified (see Section 6.12), define the source element name as "SIN**" and the source code name as "MAT*ii*", where *ii* is the sequence number of the material domain as entered in block ROCKS.

(2) Set *LTAB* = 4 and provide time information in GENER.1.1, i.e., *F1(1)* = t_p, *F1(2)* = t_0, *F1(3)* = start time, and *F1(4)* = end time of sinusoidal generation. If sinusoidal generation rates are effective over the entire simulation period, set *F1(3)* and *F1(4)* to 0.0.

(3) Provide rate information in GENER.1.2, i.e., *F2(1)* = Q_A, and *F2(2)* = Q_0. Set *F2(3)* and *F2(4)* to 0.0.

(4) If *ITAB* is not blank, provide enthalpy information in GENER.1.3, i.e., *F3(1)* = Q_A, and *F3(2)* = Q_0. Set *F3(3)* and *F3(4)* to 0.0.

An example GENER block for an element-specific sinusoidal source and a material-specific, alternating sink / source is shown in Figure 32.

```
GENER----1----*----2----*----3----*----4----*----5----*----6----*----7
A11 1SIN 1                   4     WATE
       86400.0       43200.0      172800.0        1.0E50
           2.0           5.0           0.0           0.0
SIN 2MAT 3                   4     WATE
        3600.0           0.0           0.0           0.0
           1.0           0.0           0.0           0.0
```

Figure 32. TOUGH2 input format for specifying sinusoidal generation rates.

6.8 Reinjection of Produced Fluid Components (Block GENER)

Circulation by reinjection of produced fluids is common practice in some environmental remediation technologies as well as geothermal and other applications. As the composition and enthalpy of the produced fluid mixture is a simulation result that depends on changing reservoir conditions, the reinjection rate cannot be specified a priori, but must be calculated internally. Fluid circulation can be modeled by specifying the following sink and source terms in block GENER:

(1) Define production wells, using type MASS followed by a mass production rate. All production wells need to be defined first, before injection wells are specified.

(2) Define reinjection wells using the source code name "CIR*ii*", where "*ii*" is the order number of the sink entry in block GENER of the production well to which the injection well is linked.

(3) For the reinjection well, specify (i) the component that shall be reinjected (Columns 36–39), (ii) the fraction of the produced component rate that shall be reinjected (Columns 41–50; e.g., specify a value less than 1.0 to represent fluid losses in the surface system), and (iii) an enthalpy if different from that of the produced fluid (Columns 51–60).

(4) Repeat Step (3) for all components that shall be reinjected, using the same element and source code name.

An example GENER block for the reinjection of produced water, NaCl, and CO_2 is shown in Figure 33.

```
GENER----1----*----2----*----3----*----4----*----5----*----6----*----7
A1110PRO 1                   1     MASS       -2.0
A11 1CIR 1                   1     COM3        1.0
A11 1CIR 1                   1     COM2        1.0
A11 1CIR 1                   1     WATE        1.0
```

Figure 33. TOUGH2 input format for specifying fluid reinjection.

6.9 Wells on Deliverability with Time-Dependent Bottomhole Pressure (Block GENER)

Wells on deliverability with user-specified, time-dependent bottomhole pressure may be useful if such bottomhole pressures are determined by a wellbore simulator or to examine future production scenarios. A new generation type FIXP implements this option as follows:

(1) Set variable *TYPE* in block GENER to 'FIXP' (Columns 36–39).

(2) Specify the productivity index in variable *GX* (Columns 41–50).

(3) Specify the maximum rate in variable *HG* (Columns 61–70).

(4) Set the number of table entries with time-dependent bottomhole pressures in variable *LTAB* or use the automatic counter by providing keyword 'COUNT' (Columns 26–30).

(5) Provide table of times and corresponding bottomhole pressures through variables *F1* and *F2*, respectively (use either standard format for records GENER1.1 and 1.2, or use the tabular format described in Section 6.4). Bottomhole pressures will be interpolated according to the scheme selected through variable *MOP(12)*.

Figure 34 shows an example. The well can only produce fluids from a single feedzone. If the bottomhole pressure exceeds the local reservoir pressure, production ceases (but no injection occurs).

```
GENER-T--1----*----2----*----3----*----4----PDI--5----*----6---Qmax--7
 KA 1PRO 1                 COUNT     FIXP    1.0E-12                -3.70
   0.0E+00              8.0E+06
   2.0E+07              5.0E+06
   4.0E+07              5.0E+06
   5.0E+07              1.0E+08
   6.0E+07              1.0E+00
   7.0E+07              5.0E+06
   1.0E+09              5.0E+06
```

Figure 34. TOUGH2 input format for specifying well on deliverability with time-dependent bottomhole pressure and maximum production rate.

6.10 Make-up Wells (Block GENER)

If an injection or production well is turned off because of a violation of a pressure constraint (see *MOP2(32)*), injection or production may be shifted to a make-up well. The location of a make-up well is predefined. Its rate schedule is typically identical to that of the original well. Only one make-up well can be assigned to a sink/source term (if multiple make-up wells replace a single original well, multiple sink/source terms must be specified in the element of the original well, each pointing to a single make-up well). A make-up well can itself be linked to a make-up well.

A make-up well is specified as a regular sink/source term in block GENER; it must be defined *after* the original well. To link the two wells, the sink/source code name of the make-up well must be indicated in Columns 31–35 of the original well.

Figure 35 shows an example of a GENER block with a pressure constraint and a make-up well. Variable *MOP2(32)* in block MOMOP indicates that an original well is permanently taken off-line if it violates a pressure constraint. Injection from well "INJ 1" is relocated from element " AA 1" to a make-up well "MUW 1" located in element " CA 1" if the pressure exceeds 10 MPa. If the pressure in element " KA 1" falls below 1 MPa, half of the total production rate of 4 kg/s will be relocated to make-up well "MUW 2" in element " HA 1". A make-up well of an injection well may be a production well and vice versa.

```
MOMOP----1----*----2----*----3----*----4----*----5----*----6----*----7
                                2

GENER----1----*----2----*----3----*----4----*----5----*----6----*----7
 AA 1INJ 1                     MUW 1WATE       4.00     5.0E5    1.00E7
 CA 1MUW 1                          WATE       4.00     5.0E5
 KA 1PRO12                     MUW 2MASS      -2.00              1.00E6
 KA 1PRO22                          MASS      -2.00
 HA 1MUW 2                          MASS      -2.00
```

Figure 35. TOUGH2 input format for specifying make-up wells.

6.11 Honoring Generation Times (Variable *MOP2(3)*)

By setting *MOP2(3)* = 1 in block MOMOP (see Figure 29), each time specified for any sink/source with time-dependent generation rates will be honored in the simulation, i.e., time stepping will automatically be adjusted to coincide with the time when the generation rate changes.

6.12 Material-Related Sinks/Sources (Block GENER)

Instead of providing sinks/sources for individual elements, the option described in this section allows one to specify sinks/sources for all elements that belong to a certain material domain. The injection or production mass rates specified this way are volume-specific, i.e., they have units of kg/s per m^3, which will be internally converted to a mass rate in kg/s based on the volume of the element that belongs to the identified material type.

The option is invoked by providing the material name (variable *MAT* in block ROCKS.1) as the sink/source code name, or the special sink/source code name "MAT*ii*" (variables *SL* and *NS* in block GENER.1), where the integer *ii* is the sequence number of the material domain as entered in block ROCKS.1. See Figure 36 for an example. A valid element name must also be provided (variables *EL* and *NE* in block GENER.1), even though it will internally be replaced by the multiple elements belonging to the indicated material.

For example, infiltration (or evaporation) can be conveniently specified if a one-meter thick land surface layer assigned to a unique material type is provided. Using material-related, volume-specific mass flow rates, the infiltration in kg/s (≈ mm/s) can be provided in a single GENER-block entry; it will be internally converted to mass flow rates that are proportional to the surface areas of all infiltration elements.

The model-related sinks/sources option is also convenient to specify volumetric generation rates, e.g., for hydrogen generation due to corrosion, gas from biodegradation in landfills, or radionuclide and gas generation rates in rock masses.

```
GENER----1----*----2----*----3----*----4----*----5----*----6----*----7
 AA 1INFIL                  1     WATE    3.0E-05
 AA 1MAT 3                  1     MASS   -1.0E-03
```

Figure 36. TOUGH2 input format for specifying material-related, volume-specific mass generation rates.

6.13 Sink/Source Regions (Block GENER)

Instead of providing sinks/sources for individual elements, the option described in this section allows one to specify a sink/source over a certain geometric region. The location and size of this region is parameterized and can thus be subjected to parameter estimation by inverse modeling (e.g., for contaminant source identification or optimization of well locations). For the sink/source response to be a smooth function of its location and size, the sink/source region must comprise multiple elements, with a smooth function describing the decline of the rate with distance from the center of the region. This option can only be used if element coordinates are provided in columns 51–80 in block ELEME.

The injection or production rate will be distributed over all elements within a user-specified region. This region is defined by the location of its center and the extent. The shape can be an ellipsoid, a rectangular box or cube, a cylinder or truncated cone, or a prism.

The box region includes all elements i with coordinates X_i, Y_i, and Z_i that lie within a rectangular box, i.e.,

$$\begin{aligned} d_X &= |X_i - X_c| \le L_X \\ d_Y &= |Y_i - Y_c| \le L_Y \\ d_Z &= |Z_i - Z_c| \le L_Z \end{aligned} \qquad (6.23\text{-}1a)$$

The ellipsoidal region includes all elements i with coordinates X_i, Y_i, and Z_i that satisfy the equation

$$d^2 = \left(\frac{X_i - X_c}{L_X} \right)^2 + \left(\frac{Y_i - Y_c}{L_Y} \right)^2 + \left(\frac{Z_i - Z_c}{L_Z} \right)^2 < 1 \qquad (6.12\text{-}1b)$$

Here, X_i, Y_i, and Z_i are the center coordinates of the box or ellipsoid, and L_X, L_Y, and L_Z are the three half-lengths of the box or the three semi-axes of the ellipsoid.

A cylindrical region is defined by the starting and end coordinates of its axis, and a radius. The irregular base of the prism is defined by a polygon.

The ellipsoid, box and cube can be rotated. However, the axis of the irregular prism must be aligned with one of the coordinate axes.

The total generation rate Q of the sink/source is distributed among these elements as follows:

$$q_i = Q \frac{\omega_i}{\sum \omega_i} \qquad (6.12\text{-}2)$$

where ω_i is one of the following "influence functions":

$$\omega = 1 \qquad (6.12\text{-}3a)$$

$$\omega(d) = V \cdot (1 - d) \qquad (6.12\text{-}3b)$$

$$\omega(d) = V \cdot \left(1 - \left(\frac{3}{2} d - \frac{1}{2} d^3 \right) \right) \qquad (6.12\text{-}3c)$$

$$\omega(d) = V \cdot a^{-d} \qquad (6.12\text{-}3d)$$

$$\omega(d) = \min(1, (1 - d)/a) \qquad (6.12\text{-}3e)$$

Here, V is the element volume, and d is a normalized distance from the center of the region (for boxes and ellipsoids) or from the axis of the cylinder. In Eqs. (6.12-3d) and (6.12-3e), a is a user-specified parameter, to be provided after the region-definition parameters (see Table 9). The effect of the influence function is that the farther away the element is from the region's center, the smaller is the rate assigned to this element. The rate is also weighted by the element's volume. Note that if Eq. (6.12-3a) is chosen, each element within the region will have the same volume-weighted sink/source strength. However, changing the geometry of the region will not lead to a smooth, differentiable reallocation of the generation rates; consequently, this parameter value cannot be used for contaminant source identification or well location optimization using a gradient-based algorithm.

The option is invoked by the sink/source code name "`XYZ..`" (see variable *SL* in block `GENER.1`; variable *NS* is an arbitrary two-digit integer) or the sink/source code name starts with "`&`". Note that a valid element name (*EL*, *NE*) must also be provided, even though it will internally be replaced by the multiple elements belonging to the indicated region. If a region

is requested, the integer parameters are read in format (16I5), starting on the line following GENER.1, followed by real-valued parameters read in free format.

IREGGEOM Defines geometry of the region set negative to also provide rotation angles (see Table 7) (not available for prism).
1 : Box
2 : Ellipsoid
3 : Cylinder
4 : Cube
5 : Polygonal prism (aligned with one of the coordinate axes)
6 : Right, circular, truncated cone
7 : Cylinder, cut by an angled plane at one end

IREGINFF Defines influence function. If negative, the complementary region is selected (i.e., all elements *outside* the defined geometry).
0 or 9: Constant (6.12-3a)
1 : Linear (6.12-3b)
2 : Spherical (6.12-3c)
3 : Exponential (6.12-3d)
4 : Constant-linear (6.12-3e)

NREGROCK Number of materials listed in following slots. If positive, only elements belonging to the listed materials will be assigned to the region sink/source; if negative, all elements within the region except those belonging to the listed materials will be assigned to the region sink/source. Set to zero if all elements within the region should be assigned to the region sink/source.

IREGROCK(I, i=1,ABS(NREGROCK))
Material sequence numbers of elements to be included (positive *NREGROCK*) or excluded (negative *NREGROCK*) from the region.

NPREGPOLY (only if *IREGGEOM* = 5): Number of polygon points, *NP*

IJKPRISMAXIS (only if *IREGGEOM* = 5): Alignment of polygonal prism axis
1 : prism axis aligned with *X*-axis
2 : prism axis aligned with *Y*-axis
3 : prism axis aligned with *Z*-axis

XREGION(i) (see Table 9)
XREGINFF Parameter *a* (only if |*IREGINFF*| = 3 or 4)

If the region is not aligned with the coordinate axes, set *IREGGEOM* negative and provide 3 correction angles (azimuth, dip, and plunge). Figure 37 shows an input file that uses an elliptical source region. Note that the vertical semi-axis (variable *XREGION(6)*) of the ellipsoid is very small, so only elements at an elevation of -4.5 m (variable *XREGION(1)*)

will be selected. The parameters *XREGION(i)*, *i*=1,…,7 can be varied through the iTOUGH2 command >> REGION and thus be estimated using inverse modeling.

Since the flow generation rates are calculated internally, set *MOP*(4) = 2 to see the allocation of the total flow rate of 1 kg/s to the eight elements within the ellipsoid (see Figure 38). See Section 5.3 for additional region definition examples.

Table 9. Geometrical Parameters Defining Region

IREGGEOM	*XREGION(i)*								
	1	2	3	4	5	6	7	8	9
1 (box)	X_{min}	Y_{min}	Z_{min}	X_{max}	Y_{max}	Z_{max}	*azimuth*	*dip*	*plunge*
2 (ellipsoid)	X_c	Y_c	Z_c	L_X	L_Y	L_Z	*azimuth*	*dip*	*plunge*
3 (cylinder)	X_B	Y_B	Z_B	X_E	Y_E	Z_E	R	-	-
4 (cube)	X_c	Y_c	Z_c	L_X	L_Y	L_Z	*azimuth*	*dip*	*plunge*
5 (prism)	$Axis_{min}$[@]	$Axis_{max}$[@]	$C_{1,1}$[%]	$C_{1,2}$[%]	…[%]	…[%]	$C_{NP,1}$[%]	$C_{NP,2}$[%]	
6 (cone)	X_B	Y_B	Z_B	X_E	Y_E	Z_E	R_B	R_E	-
7 (cut cylinder)	X_B	Y_B	Z_B	X_E	Y_E	Z_E	R	*icut*[$]	*angle*[&]

[@] Minimum and maximum coordinate of prism axis, i.e., location of prism bases; coordinate variable given by *IJKPRISMAXIS*

[%] Coordinates for *NP* polygon points (clockwise or anti-clockwise); two coordinates per point: (*Y*,*Z*) for *IJKPRISMAXIS* = 1; (*X*,*Z*) for *IJKPRISMAXIS* = 2; (*X*,*Y*) for *IJKPRISMAXIS* = 3

[$] +/–: start/end face of cylinder is cut
|*icut*|=1: Cutting plane rotates around axis perpendicular to cylinder axis in *Y*-*Z* plane
|*icut*|=2: Cutting plane rotates around axis perpendicular to cylinder axis in *X*-*Z* plane
|*icut*|=3: Cutting plane rotates around axis perpendicular to cylinder axis in *X*-*Y* plane
For a cylinder axis at arbitrary angle in Y-Z, |*icut*|=1 only
For a cylinder axis at arbitrary angle in X-Z, |*icut*|=2 only
For a cylinder axis at arbitrary angle in X-Y, |*icut*|=3 only
For a cylinder axis parallel to the X axis (in X-Y plane and X-Z plane), |*icut*|=3 or |*icut*|=2
For a cylinder axis parallel to the Y axis (in X-Y plane and Y-Z plane), |*icut*|=3 or |*icut*|=1
For a cylinder axis parallel to the Z axis (in X-Z plane and Y-Z plane,) |*icut*|=2 or |*icut*|=1

[&] Rotation angle of the cutting plane, measured counter-clockwise from the cylinder axis within the chosen plane. The rotation axis of the cutting plane is perpendicular to and must be within the same plane as the cylinder axis.

```
Test ellipsoidal source region option
ROCKS----1----*----2----*----3----*----4----*----5----*----6----*----7
SAND1                     0.5 1.000E-12 1.000E-12 1.000E-12

MESHM----1----*----2----*----3----*----4----*----5----*----6----*----7
XYZ

NX      10       1.0
NY      10       1.0
NZ      10       1.0


PARAM----1-MOP: 123456789012345678901234----*----5----*----6----*----7
   2   1       1   2
                       1.0E-5

                 0.5
START----1----*----2---------3----*----4---------5---------6----*----7
INCON----1----*----2---------3----*----4---------5---------6----*----7

GENER----1----*----2---------3----*----4---------5---------6----*----7
    1XYZ 1                   1     WATE        1.0
2   1    0
       5.0       4.7      -4.5       2.5       1.1       0.1

ENDCY----1----*----2----*----3----*----4----*----5----*----6----*----7
```

Figure 37. TOUGH2 input file for testing ellipsoidal sink/source regions.

```
QQQQQQQQQQ SUBROUTINE QU QQQQQQQQQQ  --- [KCYC,ITER] = [   1,  1]
ELEMENT A55 4   SOURCE XYZ 1 --- FLOW RATE = 0.132698E+00
ELEMENT A56 4   SOURCE XYZ 1 --- FLOW RATE = 0.203361E-01
ELEMENT A55 5   SOURCE XYZ 1 --- FLOW RATE = 0.259560E+00
ELEMENT A56 5   SOURCE XYZ 1 --- FLOW RATE = 0.874065E-01
ELEMENT A55 6   SOURCE XYZ 1 --- FLOW RATE = 0.259560E+00
ELEMENT A56 6   SOURCE XYZ 1 --- FLOW RATE = 0.874065E-01
```

Figure 38. Excerpt of TOUGH2 output file showing allocation of generation rates within ellipsoidal source region.

6.14 STAR-Type Mass and Energy Sinks/Sources (Block GENER)

The STAR simulator [*Pritchett*, 2002] provides options to specify fluid mass and energy sink/source terms as linear functions of local instantaneous pressure and/or temperature:

$$\frac{\partial}{\partial t}\frac{m^{\kappa}}{V} = q + r \cdot P + s \cdot T \qquad (6.13\text{-}1)$$

The term on the left-hand side is a mass or energy rate per gridblock volume (in kg s^{-1} m^{-3} or J s^{-1} m^{-3}, respectively), and *q*, *r*, and *s* are user-specified coefficients.

If *r* and *s* are zero, the sink/source is a time-constant injection or production rate (equivalent to the standard TOUGH2 constant generation rate variable *GX* with *LTAB* ≤ 1). To specify a pressure-dependent mass flow rate—representing, for example, a region whose leakage rate is $\dot{m}_0$ at the initial reservoir pressure P_0 and is zero at the limiting reservoir pressure P^*, one might set $q = \dot{m}_0/(1 - P_0/P^*)$ and $r = -\dot{m}_0/(P^* - P_0)$. To approach and maintain a particular pressure $\tilde{P}$, set $r = -q/\tilde{P}$ with q chosen sufficiently positive. To simulate heat flow into the system by conduction from adjacent strata caused by production-induced cooling, one might set $s = -q/T^0$, where T^0 is the local initial temperature, and q is an appropriately chosen positive constant. As before, if q is chosen large enough, the effect will be to approach and approximately maintain the gridblock temperature T^0 at throughout the simulation.

STAR-type mass and energy sinks/sources are initiated by writing the *TYPE* variable with a capitalized first letter and lower-case characters in the second through fourth position, e.g., `Mass`, `Heat`, `Wate`, and `Com2` (instead of `MASS`, `HEAT`, `WATE`, and `COM2`).[i] Coefficients *q*, *r*, and *s* are provided through TOUGH2 input variables *GX*, *HG*, and *GMOD* in columns 41–50, 61–70, and 71–80 of block `GENER.1`, respectively.

STAR-type mass and energy sinks/sources can be combined with material-related sink/sources (see Section 6.12) or sink/source regions (see Section 6.13). For *r* and/or *s* not equal zero, the actual rates become non-constant; the option therefor leads to time-changing rates despite *LTAB* being set to ≤ 1.

6.15 Infiltration type (Block GENER)

The aim of the infiltration generation type is to reduce the size of the `GENER` block in groundwater models. The keyword `INFL` is used instead of `MASS`. It is followed by the net mass flow rate across the land surface (kg s^{-1} m^{-2}, parameter *GX*), the specific enthalpy (J/kg, parameter *EX*, not required for isothermal simulations), and the area (m^2, parameter *HX*). Time-dependent infiltrations can be specified. When several elements belong to the same subarea, the infiltration time-series do not have to be repeated. Setting *LTAB* to –1 indicates that the infiltration data (*GX* and *EX*) from the preceding entry are to be used. However, an area (*HX*) must be specified for each entry.

[i] If the `TYPE` variable is provided in all lower-case characters, the standard TOUGH2 sink/source options are used, albeit with the rates given in mass or energy per unit volume, i.e., kg s^{-1} m^{-3} and J s^{-1} m^{-3}.

In the example shown in Figure 39, the infiltration in element CCE72 is constant at 2.8199×10^{-5} kg s^{-1} m^{-2}. In element CCE73 the infiltration is time dependent. Elements CBT80, CBT82, and CBT87 have the same infiltration as CCE73, but the areas of the elements are different (HX).

```
GENER T--1----*----2----*----3----*----4----*----5----*--- Surface Area ---*----8
CCE72AES01                    1     INFL 2.8199e-05           4.664e+02
CCE73AES02                COUNT     INFL                      4.664e+02
    -1.0e15 2.8199e-05
 1383264000 0.0000e+00
 1383350400 0.0000e+00
…
 1419984000 3.3296e-06
 1420156800 0.0000e+00
 1420243200 0.0000e+00
    +1.0e15 8.5377e-07
CBT80AES02                   -1     INFL                      1.877e+02
CBT82AES02                   -1     INFL                      3.069e+02
CBT87AES02                   -1     INFL                      1.448e+02
```

Figure 39. Infiltration type in GENER block.

Alternatively, the element surface can be specified as parameter *AHTX* in the ELEME block. When this is the case, then the *HX* parameter should be set to zero or left empty. Note that, when *AHTX* is used for infiltration, then *MOP(15)* must be set to zero (no heat exchange with confining layers).

```
ELEME----1----*----2----*--- Surface Area ---*----5----*----6----*----7----*----8

…
CCE72          MORSG3.7874e+02 4.664e+02           355856.83 407034.25     +4.85

…


GENER T--1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
CCE72AES01                COUNT     INFL 2.8199e-05
…
```

Figure 40. Infiltration type in GENER block with infiltration area specified in block ELEME.

7 Relative Permeability and Capillary Pressure Functions

Subroutines `RELP` and `PCAP` provide the relative permeability and capillary pressure functions, respectively (see *Pruess* [1987]). They are frequently modified to accommodate particular needs. The user should therefore carefully check the functional form and required parameters before selecting a certain curve through variable *`IRP`* and *`ICP`*, respectively (see also command `>>> CHARACTERISTIC`).

The functions provided in this version include the three-phase curves used by the T2VOC module [*Falta et al.*, 1995]. Additional functions are provided as described in the following subsections.

7.1 Modified Brooks-Corey Model (*IRP* = 10 and *ICP* = 10)

A modified version of the Brooks-Corey model [*Luckner et al.*, 1989] has been implemented. In order to prevent the capillary pressure from decreasing towards negative infinity as the effective saturation approaches zero, a linear function is used for saturations S_l below a certain value $(S_{lr} + \varepsilon)$, where ε is a small number. The slope of the linear extrapolation is identical with the slope of the capillary pressure curve at $S_l = S_{lr} + \varepsilon$. Alternatively, the capillary pressure is prevented from becoming more negative than $-p_{c,\max}$.

The modified Brooks-Corey model is invoked by setting both *`IRP`* and *`ICP`* to 10. The model is described by the following set of equations (the input parameters are listed in Table 10):

$$S_{ec} = \frac{S_l - S_{lrc}}{1 - S_{lrc}} \tag{7.1-1a}$$

$$S_{ek} = \frac{S_l - S_{lrk}}{1 - S_{lrk} - S_{gr}} \tag{7.1-1b}$$

$$p_c = -p_e \left(S_{ec}\right)^{-1/\lambda} \qquad \text{for} \quad S_l \geq (S_{lrc} + \varepsilon) \tag{7.1-2a}$$

$$p_c = -p_e \left(\frac{\varepsilon}{1 - S_{lrc}}\right)^{1/\lambda} + \frac{p_e}{\lambda} \frac{1}{1 - S_{lrc}} \left(\frac{\varepsilon}{1 - S_{lrc}}\right)^{\frac{1+\lambda}{\lambda}} \left(S_l - S_{lrc} - \varepsilon\right) \qquad \text{for} \quad S_l < (S_{lrc} + \varepsilon) \tag{7.1-2b}$$

$$p_c \geq -p_{c,\max} \tag{7.1-3}$$

$$k_{rl} = S_{ek}^{\frac{2+3\lambda}{\lambda}} \tag{7.1-4a}$$

$$k_{rg} = (1 - S_{ek})^2 \left(1 - S_{ek}^{\frac{2+\lambda}{\lambda}}\right) \tag{7.1-4b}$$

$$k_{rg} = 1 - k_{rl} \tag{7.1-4c}$$

Table 10. Input Parameters for Modified Brooks-Corey Model

Parameter	Variable	Description
`IRP`	10	select Brooks-Corey relative permeability model
`RP(1)`	S_{lrk}	residual liquid saturation for relative permeability functions
`RP(2)`	S_{gr}	residual gas saturation
`RP(3)`	(flag)	if zero, use (11b), otherwise (11c)
`ICP`	10	select Brooks-Corey capillary pressure model
`CP(1)`	λ	pore size distribution index
`CP(2)`	p_e	gas entry pressure [Pa]
		if `CP(2)` negative and `USERX(1,N)` non-zero, apply Leverett's rule: $p_e = -CP(2) \cdot \sqrt{\frac{PER(NMAT)}{USERX(1,N)}}$
		if `USERX(4,N)` positive then $p_e =$ `USERX(4,N)`
		if `USERX(4,N)` negative then $p_e =$ `-USERX(4,N)·CP(2)`
`CP(3)`	ε or	if `CP(3)` $= 0$ then $p_{c,\max} = 10^{50}$, $\varepsilon = -1$
	$p_{c,\max}$	if $0 <$ `CP(3)` < 1 use linear model (9b) for $S_l < S_{lr} + \varepsilon$
		if `CP(3)` ≥ 1, then $p_{c,\max} =$ `CP(3)`, $\varepsilon = -1$
`CP(6)`	S_{lrc}	if zero, then $S_{lrc} = S_{lrk}$

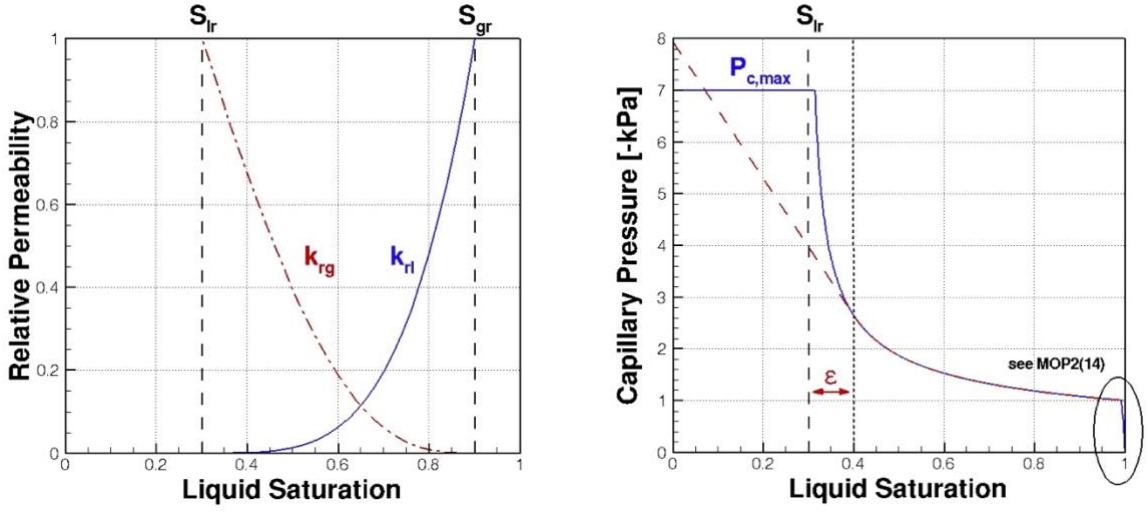

Figure 41. Modified Brooks-Corey relative permeability and capillary pressure curves.

7.2 Modified van Genuchten Model (*IRP* = 11 and *ICP* = 11)

A modified version of the van Genuchten model [*van Genuchten, 1980; Luckner et al.*, 1989] has been implemented. In order to prevent the capillary pressure from decreasing towards negative infinity as the effective saturation approaches zero, a linear function is used for saturations S_l below a certain value $(S_{lr} + \varepsilon)$, where ε is a small number. The slope of the linear extrapolation is identical with the slope of the capillary pressure curve at $S_l = S_{lr} + \epsilon$. Alternatively, the capillary pressure is prevented from becoming more negative than $-p_{c,max}$.

The modified van Genuchten model is invoked by setting both *IRP* and *ICP* to 11. The model is described by the following set of equations (the input parameters are listed in Table 11):

$$S_{ec} = \frac{S_l - S_{lrc}}{1 - S_{lrc}} \tag{7.2-1a}$$

$$S_{ekl} = \frac{S_l - S_{lrk}}{1 - S_{lrk}} \tag{7.2-1b}$$

$$S_{ekg} = \frac{S_l}{1 - S_{gr}} \tag{7.2-1c}$$

$$S_{ek*} = \frac{\varepsilon}{1 - S_{lrc}} \tag{7.2-1d}$$

$$S_e = \frac{S_l - S_{lr}}{1 - S_{lr} - S_{gr}} \tag{7.2-1e}$$

$$p_c = -\frac{1}{a}\left[\left(S_{ec}\right)^{(g-1)/m} - 1\right]^{1/n} \quad \text{for } S_l \geq (S_{lrc} + \epsilon) \tag{7.2-2a}$$

$$p_c = -\frac{1}{a}\left[S_{ec*}^{(g-1)/m} - 1\right]^{1/n} - b \times \left(S_l - S_{lrc} - e\right) \quad \text{for } S_l < (S_{lrc} + \epsilon) \tag{7.2-2b}$$

with

$$\text{linear extension:} \quad b = -\frac{(1-g)}{anm} \times \frac{1}{(1 - S_{lrc})} \times \left(S_{ec*}^{(g-1)/m} - 1\right)^{\frac{1}{n}-1} S_{ec*}^{\left(g-1-\frac{m}{m}\right)}$$

$$p_c = -\frac{1}{a}\left[S_{ec*}^{(g-1)/m} - 1\right]^{1/n} \times 10^{b \times \left(S_l - S_{lrc} - e\right)} \quad \text{for } S_l < (S_{lrc} + \epsilon) \tag{7.2-2c}$$

with

$$\text{log-linear extension:} \quad b = -\log_{10}(e) \times \left(\frac{1-m}{m} \times \frac{g-1}{e} \times \frac{1}{S_{ec*}^{(1-g)/m} - 1}\right)$$

$$p_c \geq -p_{c,\max} \tag{7.2-2d}$$

$$k_{rl} = S_{ekl}^{\gamma} \cdot S_{ekl}^{(1-\gamma)\eta} \cdot \left[1 - \left(1 - S_{ekl}^{(1-\gamma)/m}\right)^{m}\right]^{2} \tag{7.2-3a}$$

$$k_{rg} = \left(1 - S_{ekg}\right)^{\xi} \left(1 - S_{ekg}^{1/m}\right)^{2m} \tag{7.2-3b}$$

$$k_{rg} = 1 - k_{rl} \tag{7.2-3c}$$

$$k_{rg} = (1 - S_l)^{\beta} \tag{7.2-3d}$$

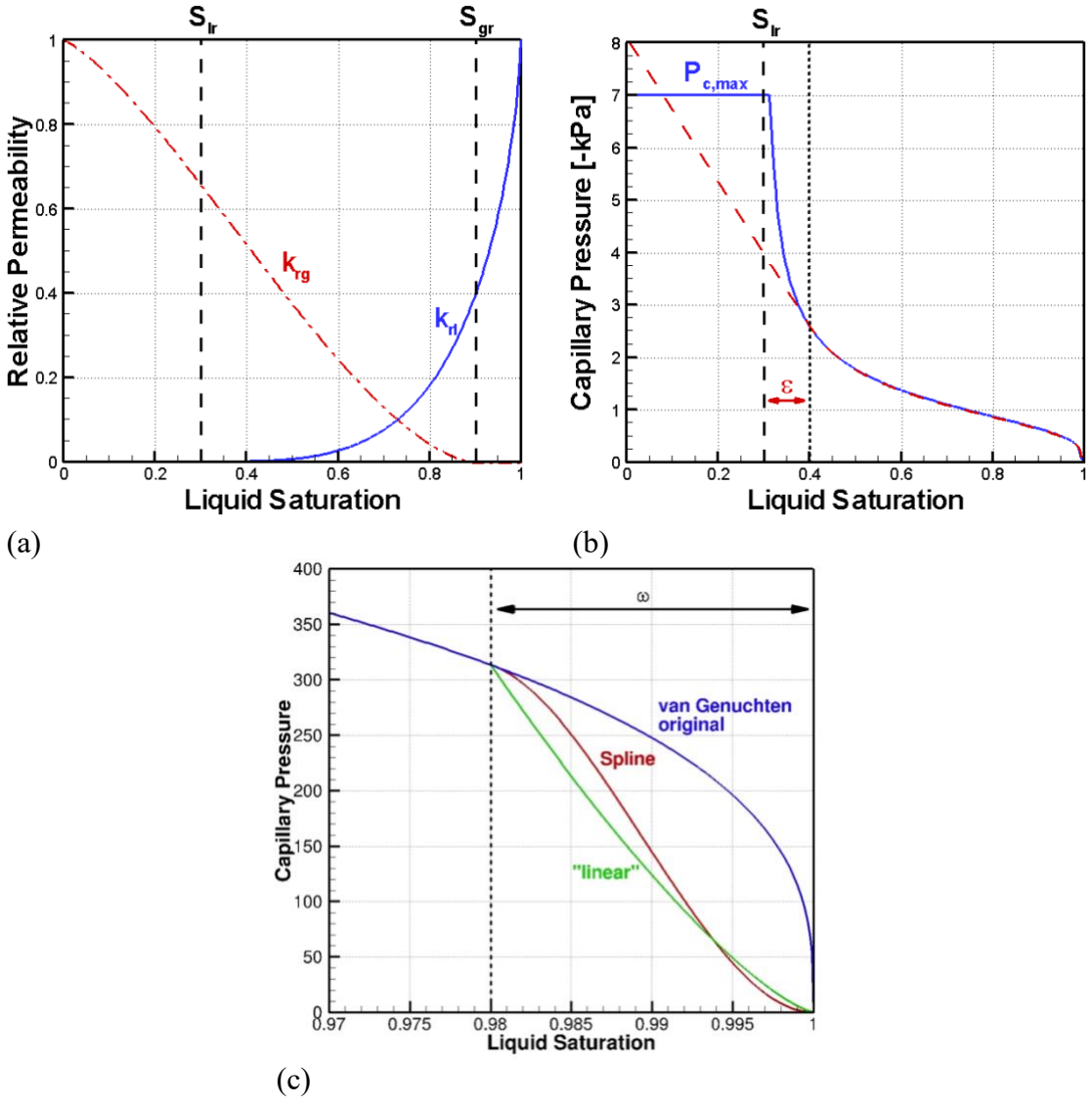

Figure 42. Modified van Genuchten (a) relative permeability, and (b) capillary pressure curves with (c) options for capillarity near full saturation (for |*CP(5)*| = 0.02).

Table 11. Input Parameters for Modified van Genuchten Model

Parameter	Variable	Description
		Relative Permeability Function
IRP	11	select van Genuchten relative permeability model
RP(1)	S_{lrk}	residual liquid saturation for relative permeability functions, if negative; $S_{lrk} = 0$ for gas relative permeability, absolute value is used for liquid relative permeability
RP(2)	S_{gr}	residual gas saturation; if negative, $S_{gr} = 0$ for liquid relative permeability, absolute value is used for gas relative permeability
RP(3)	(flag)	if zero, use (7.2-3b); if > 0, use (3c); if < 0, use (3d) with β = −RP(3)
RP(4)	η	exponent in (7.2-3a), default = ½
RP(5)	ε_k	use linear function between $k_{rl}(S_e = 1-\varepsilon_k)$ and 1.0.
RP(6)	f_g or a_{fm}	if positive: gas permeability enhancement factor, g_f if negative: constant fracture-matrix interaction reduction factor, a_{fm}, in combination with Active Fracture Model (see Section 8)
RP(7)	ξ	exponent in (7.2-3b), default = 1/3
		Capillary Pressure Function
ICP	11	select van Genuchten capillary pressure model
CP(1)	n	parameter related to pore size distribution index (see also *CP(4)*); if negative, use (7.2-1e) so that $k_{rl}(S > 1\text{-}S_{gr}) = 1$ and $k_{rg}(S_l < S_{lr}) = 1$
CP(2)	$1/\alpha$	parameter related to gas entry pressure [Pa] USERX(4,N)>0 : $1/\alpha_i$ = USERX(4,N) USERX(4,N)<0 : $1/\alpha_i$ = USERX(4,N) · CP(2) CP(2)<0: apply Leverett scaling rule: $1/\alpha_i = 1/\alpha_{ref} \cdot \sqrt{k_{ref}/k_i}$ where: $1/\alpha_{ref}$ = |CP(2)| k_{ref} = PER(NMAT) USERX(1,N)>0: k_i = USERX(1,N) USERX(1,N)<0: k_i = USERX(1,N) · PER(NMAT)
CP(3)	ε or $p_{c,\max}$	CP(3)=0 : $p_{c,\max} = 10^{50}$, $\varepsilon = -1$ 0<CP(3)<1 : ε = CP(3); use linear extension (7.2-2c) CP(3) ≥ 1 : $p_{c,\max}$ = CP(3), $\varepsilon = -1$ –1<CP(3)<0 : ε = |CP(3)|; use log-linear extension (7.2-2d)
CP(4)	m	if zero then m=1-1/CP(1), else m=CP(4) and n=1/(1-m)
CP(5)	ω	zero: Use original van Genuchten curve near full saturation positive: Use cubic spline for $S_l > 1 - \omega$ negative: Use "linear" interpolation for $S_l > 1 - \lvert\omega\rvert$
CP(6)	γ	parameter of Active Fracture Model (see Section 8)
CP(7)	S_{lrc}	if zero or >10, then $S_{lrc} = S_{lrk}$; if >10, then τ (see Section 7.5)

7.3 Tabular Relative Permeability and Capillary Pressure Curves (*IRP* = 23 and *ICP* = 23)

Measured relative permeability or capillary pressure data can be entered in tabular form by setting *IRP* = 23 and *ICP* = 23, respectively. Tabular curves can be provided for individual materials in block ROCKS (by setting *NAD* = 2) or in block RPCAP as default characteristic curves. The tables holding relative permeability or capillary pressure data must be provided on separate files for each material. The file name must be provided after Column 81 on the line containing variable *IRP* or *ICP*, respectively.

The file for relative permeabilities must contain *NPH*+1 columns without a header line. The first column is phase saturation (preferably covering the entire range from 0.0 to 1.0); the second column is the relative permeability of Phase 1 (typically the gas phase); the third column is the relative permeability of the second Phase (typically the aqueous phase); etc. The number of table entries (rows), *MTAB*, is given by parameter *RP(1)* (note that while the number of rows is naturally an integer value, *RP(1)* = *MTAB* has to be entered as a real number and thus must include a decimal point). If no data for gas relative permeability are available, the corresponding column may consist of special values to invoke simple relative permeability functions as follows: If $\left(k_{rg}^{*}(i) = -1.0, i = 1, MTAB\right)$, then $k_{rg} = 1 - k_{rl}(S_l)$; if $\left(k_{rg}^{*}(i) = -2.0, i = 1, MTAB\right)$, then $k_{rg} = S_g^{RP(2)}$.

The file for capillary pressure must contain *NPH* columns without a header line. The first column is phase saturation (preferably covering the entire range from 0.0 to 1.0); the second column is the capillary pressure between Phases 1 and 1 (typically the gas-liquid capillary pressure), and the third column is the capillary pressure between Phases 1 and 3 (typically the gas-oil capillary pressure). The number of table entries (rows), *MTAB*, is given by parameter *CP(1)* (note that while the number of rows is naturally an integer value, *CP(1)* = *MTAB* has to be entered as a real number and thus must include a decimal point). Figure 43 shows examples of the ROCKS and RPCAP block for specifying characteristic curves in tabular form.

```
ROCKS----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8  Filename
SAND1    0     2650.     .3500 1.000E-12                          2.51      920.
SAND2    2     2650.     .3500 2.000E-12                          2.51      920.
 1.000E-09
    3           0.20      0.05
   23           51.0                                                          pcap.dat

RPCAP----1----*-MTAB----*----3----*----4----*----5----*----6----*----7----*----8
   23           101.0       3.0                                             relpd.dat
   23           101.0                                                       pcapd.dat
```

Figure 43. Blocks ROCKS and RPCAP for specifying tabular relative permeability and capillary pressure curves

7.4 Logarithmic and Zero Capillary Pressure Functions (*ICP* = 8 and *ICP* = 9)

For two-phase EOSs, *ICP* = 8 calculates $p_c = CP(1) \cdot \log(S_l)$. For three-phase EOSs, *ICP* = 8 is the model by *Parker et al.* [1987] (see also *Falta et al.* [1995] and *Pruess and Battistelli* [2002]). To set capillary pressures to zero for any saturation, set *ICP* = 9.

7.5 Dynamic Capillary Pressures

Capillary-saturation relationships are typically obtained experimentally under equilibrium conditions. However, if equilibrium between the wetting and nonwetting phase pressures is not reached within a given time step, the capillary pressure curve depends on the rate of saturation change. On a macroscopic level, this dynamic effect can be approximated by the following equation [*Hassanizadeh et al.,* 2002]:

$$p_c^{dyn} = p_c^{equ} - \tau \frac{\partial S_l}{\partial t} \tag{7.5-1}$$

Here, τ [Pa·s] is a material parameter that can be interpreted as capillary damping coefficient, i.e., a measure of the speed with which a change in saturation takes place, with small values indicating fast equilibration of the capillary pressure. For porous media of low permeability, where the dynamic capillary pressure effect may be relevant, large τ values (on the order of 10^{12} Pa·s or higher) can be expected. In this simplified model, τ is considered to be non-hysteretic and independent of saturation.

Dynamic capillary pressures are activated in iTOUGH2 by providing τ as a non-negative parameter in `CP(7)` in block `RPCAP` or `ROCKS.1.2`. Dynamic capillary pressures are not available for three-phase capillary pressure curves. Furthermore, it is assumed that `CP(7)` is void, i.e., not used by the selected capillary pressure function (for a notable exception, see Table 11).

8 Active Fracture Model

8.1 Active Fracture Concept

There is evidence that only a portion of the connected fracture network conducts water under unsaturated conditions. The fractures contributing to liquid flow are referred to as "active fractures". The Active Fracture Concept (AFC) was developed by *Liu et al.* [1998] to describe gravity-dominated, non-equilibrium, preferential liquid flow in fractures, which is expected to be similar to fingering in unsaturated porous media. AFC is based on the hypothesis that (1) the number of active fractures is small compared to the total number of connected fractures, (2) the number of active fractures within a grid block is large so that the continuum approach is valid, and (3) the fraction of active fractures, f_a, is related to water flux and equals one for a fully saturated system, and zero if the system is at residual saturation. The following power function of effective liquid saturation, S_e, fulfills these conditions:

$$f_a = S_e^{\gamma} \tag{8.1-1}$$

Here, γ is a positive constant depending on properties of the fracture network, and S_e is the effective liquid saturation given by

$$S_e = \frac{S_l - S_{lr}}{1 - S_{lr}} \tag{8.1-2}$$

Capillary pressure and relative permeability functions are modified to account for the fact that the effective saturation in the active fractures, S_{ea}, is larger than the effective saturation of the total fracture continuum:

$$S_{ea} = \frac{S_e}{f_a} = S_e^{1-\gamma} \tag{8.1-3}$$

Using the van Genuchten model, capillary pressure and liquid relative permeability are given, respectively, by

$$p_c = -\frac{1}{\alpha}\left[S_e^{(\gamma-1)/m} - 1\right]^{\frac{1}{n}} \tag{8.1-4}$$

and

$$k_{rl} = S_e^{(\gamma+1)/2}\left\{\left[1 - S_e^{(\gamma-1)/m}\right]^m\right\}^2 \tag{8.1-5}$$

The fracture-matrix interface area reduction factor is given by

$$a_{fm} = S_e^{1+\gamma} a_{fm} = S_e^{1+\gamma} \tag{8.1-6}$$

The AFC is invoked by selecting $\gamma > 0$, which is provided as an additional parameter of the standard van Genuchten model (`ICP`=7) through variable `CP(6,NMAT)`. Fracture-matrix interface area reduction is invoked by selecting `ISOT` between -10 and -12 (see Table 12).

8.2 Reduction of Fracture-Matrix Interface Area

There is evidence that fracture-matrix interaction in the unsaturated zone is reduced as a result of fracture coatings as well as preferential flow in the fractures as invoked by flow instabilities (fingering) and small-scale heterogeneities. A number of options for reducing fracture-matrix interface area have been implemented for use in a dual-permeability flow simulation. Interface area reduction is applied to connections with a negative value for variable *ISOT*, which is provided in the CONNE block. Different modifiers are used depending on the value of *ISOT* and *MOP(8)* as summarized in Table 12.

Table 12. Option for Reducing Fracture-Matrix Interface Area

ISOT	*MOP*(8)	Interface area reduction factor a_{fm}
positive	any	No interface area reduction, i.e., $a_{fm} = 1$
negative	1	$a_{fm} = RP(6, NMAT)$
-1, -2, -3	0	$a_{fm} = S_\beta$
	2	$a_{fm} = S_\beta \cdot RP(6, NMAT)$
-4, -5, -6	0	$a_{fm} = k_{r\beta}$
	2	$a_{fm} = k_{r\beta} \cdot RP(6, NMAT)$
-7, -8, -9	0	$a_{fm} = RP(6, NMAT)$
-10, -11, -12	0	$a_{fm} = S_e^{1+\gamma}$ (see Section 8.1)

a_{fm}	:	Fracture-matrix interface area reduction factor.
S_β	:	For flow of phase β, upstream saturation of phase β.
$k_{r\beta}$	:	For flow of phase β, upstream relative permeability of phase β.
$RP(6, NMAT)$	:	6th parameter of rel. perm. function of upstream element; if zero (i.e., not specified), reset to one.

9 Coupled Overland – Subsurface Flow (Material *SURWA*)

iTOUGH2 provides the capability to fully couple overland flow (solving the non-inertial, diffusion wave form of the Saint-Venant equations) with subsurface flow using an approach similar to that proposed by *Weill et al.* [2009]. The momentum and continuity equations are given by:

$$S_{f,i} = -\nabla \cdot (z_l + h_s) \tag{9-1}$$

$$\frac{\partial h_s}{\partial t} + \nabla \cdot \left(h_s \, \vec{U}\right) = q_s \tag{9-2}$$

where $S_{f,i}$ is the friction slope [-] in the direction i, z_l is land surface elevation [L], h_s is the water depth on the surface, $\vec{U}$ is the depth averaged flow velocity [LT^{-1}], and q_s is a source/sink term [LT^{-1}]. The Manning-Strickler formula is used for relating velocity to friction slopes:

$$U_i = \frac{h_s^{2/3}}{n_{man}} \sqrt{S_{f,i}} \tag{9-3}$$

where n_{man} is the Manning roughness coefficient [$L^{-1/3}T$]. The diffusion-wave form of the Saint-Venant equations assumes slowly varying flow.

In order to couple the surface and subsurface flow equations, the approach developed by *Weill et al.* [2009] is followed. A surface layer of thickness *e* is expected to be present at the top of the numerical model. For liquid flow within the surface layer, Eqs. (9-1)–(9-3) are combined into a form that is similar to that describing flow in a porous medium:

$$\frac{\partial h_s}{\partial t} - \nabla \cdot [K_s \nabla (z + l + h_s)] = q_s \tag{9-4}$$

Here, the non-diagonal terms of the hydraulic conductivity tensor K_s are zero and the diagonal components are

$$K_{s,xx} = \frac{h_s^{5/3}}{n_{man}\sqrt{\nabla_x (z_l + h_s)}} \tag{9-5}$$

$$K_{s,yy} = \frac{h_s^{5/3}}{n_{man}\sqrt{\nabla_y (z_l + h_s)}} \tag{9-6}$$

$$K_{s,zz} = k_{zz} \frac{k_{rl}}{\mu_l} \tag{9-7}$$

The horizontal hydraulic conductivities describe surface water flow, while the vertical hydraulic conductivity describes resistance to liquid flow between the surface and subsurface layer, with k_{zz} equal to the vertical permeability of the subsurface layer. The liquid pressure in the surface layer is assumed hydrostatic. Because liquid and gas pressures are continuous across the surface/subsurface boundary, negative water depths occur when there is no runoff. The volumetric liquid content in the surface layer is defined as

$$\theta_l = \begin{cases} 0 & \text{for } h_s < 0 \\ h_s/e & \text{for } h_s \geq 0 \end{cases} \tag{9-8}$$

For vertical liquid flow, the liquid relative permeability is set to one, unless $h_s/e < 10^{-5}$

when it is specified as zero. To capture the pressure head due to ponding in the surface layer, a positive capillary pressure is calculated as a function of h_s.

For gas flow within the surface layer and between the surface and subsurface layers, the regular subsurface flow equations are used. If runoff occurs in the surface layer, i.e., $\theta_l > 0$ then $k_{rg} = 0$ for pressure gradients from the surface to the subsurface layers such that no gas flows between the surface and subsurface layers (note, however, that it is possible for pressurized gas to escape the subsurface and flow to the surface layer), and $k_{rg} = 1$ within the surface layer such that gas flows freely in the surface layer. If there is no runoff, $\theta_l = 0$ $k_{rg} = 1$ and the intrinsic permeability of the surface layer is assumed isotropic and equal to the vertical intrinsic permeability of the subsurface layer.

To implement coupled surface water – groundwater flow, the user has to set up a TOUGH2 model—similar to that shown in Figure 44—as follows:

(1) Create a material in block `ROCKS` named `SURWA`. All elements representing surface water must be related to this material type.

(2) For material `SURWA`, set porosity close to 1.0, set *NAD*=1, and provide two Manning's coefficients referring to the first and second direction (i.e., for *ISOT*=1 and 2) and the surface layer thickness (consistent with its definition in block ELEME) in columns 51–60, 61–70, and 71–80, respectively. Do *not* specify material-dependent relative permeability and capillary pressure function using *NAD*=2; these functions are provided internally.

(3) To specify a zero-water-depth-gradient boundary condition, create boundary elements (either inactive or of large volume), assign them to a material named `SURZG`, and connect them to the surface water elements at the desired locations.

(4) Create a material in block `ROCKS` named `ATMOS` and assign properties suitable for representing atmospheric conditions.

(5) Note that the absolute permeabilities in materials `SURWA`, `SURZG`, and `ATMOS` will be used to calculate gas flow only; liquid flow will be determined by the Saint-Venant equation.

(6) Generate a mesh with an atmospheric element (or layer) and a surface-water layer (typically of thickness 1 m). Assign materials `ATMOS` and `SURWA` to these two layers.

(7) Connect the atmosphere to the surface-water layer, and the surface-water layer to the subsurface system. Connections between surface-water elements must be assigned to directions *ISOT*=1 and 2; the slope of the surface is provided through variable *BETAX*. Connections between surface-water and subsurface elements must be assigned to direction *ISOT*=3; the nodal distance from the surface-water element to the interface with the subsurface element is internally set to zero.

Surface-water flow is solved simultaneously and fully coupled with subsurface flow using the standard TOUGH2 implicit scheme. Note that time-step size may be governed by the relatively fast flow occurring in the surface-water layer.

```
Surface water - subsurface water flow test problem
ROCKS----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
SURWA    1     2650.     0.999   1.0e-12   1.0e-12    1.e-12        1.     1000.
                                                        0.1       0.1       1.0
SANDY    2     2650.     0.300   1.0e-12   1.0e-12    1.e-12        1.     1000.

    7           .818      0.25       1.0
    7           .818      0.25 .00023488      1.e7       1.0
ATMOS    2     2650.     0.999   1.0e-12   1.0e-12    1.e-12        1.     1000.

    3            0.9
    1            0.0       0.0       1.0

START----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
----*----1 MOP: 123456789*123456789*1234----*----5----*----6----*----7----*----8
PARAM----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
  -4 100       1100000900000000400006000
    0.0E00               0.001    1.0e20               10.0
     1.E-5
               1.0E5              10.750              20.00
MOMOP----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
2
ELEME----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
A11 0          ATMOS0.1000E+520.2240E-02               0.75     0.500+.0000E+00
SWA 1          SURWA0.1000E+010.1000E+01              0.500     0.500    -0.500
A21 1          SANDY0.1000E+010.1000E+01              0.500     0.500    -1.005
A21 2          SANDY0.1000E+010.1000E+01              1.500     0.500    -1.005
SWA 2          SURWA0.1000E+010.1000E+01              1.500     0.500    -0.500

CONNE
SWA 1SWA 2                  10.5000E+000.5000E+000.1000E+01       0.1
SWA 1A21 1                  30.0000E-000.5000E-000.1000E+010.1000E+01
A21 1A21 2                  10.5000E+000.5000E+000.1000E-01
SWA 2A21 2                  30.0000E-000.5000E-000.1000E+010.1000E+01
A11 0SWA 1                  30.0000E-000.5000E-000.1000E-100.1000E+01
A11 0SWA 2                  30.0000E-000.5000E-000.1000E-100.1000E+01

INCON----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
A11 0           0.99900000E+00
 0.1000000000000E+06 0.1000000000000E+01 0.2000000000000E+02

ENDCY----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
```

Figure 44. TOUGH2 input file for testing coupled surface water – subsurface flow.

10 Semi-Analytical Radial Heat Exchange (Variable *MOP(15)*)

Radial, conductive heat exchange between fluids in a discretized wellbore and the formation is calculated using a semi-analytical, time-convolution method. The time-dependent temperature evolution in the wellbore is calculated numerically using TOUGH2. At each time step, radial heat transfer with the formation is calculated by superposition of analytical solutions of heat flow that are dependent on the temperature differences between subsequent time steps.

Carslaw and Jaeger [1959, pp. 334–339] provided an approximate solution for heat conduction between a cylinder and surrounding media where the temperature of the cylinder is maintained constant. If the initial temperature difference between the two domains is $\Delta T = T_w - T_f$ (where T_w and T_f are the temperatures in the well and the formation, respectively), the heat flux q from the wellbore to the formation can be calculated using as the product of a heat transfer function and the temperature using Eq. (10-1) for small values of the dimensionless time $t_d = \alpha t / r_0^2$, where α is the thermal diffusivity, and Eq. (10-2) for large values of t_d:

$$q = f_1(t_d) \cdot \Delta T = \frac{k\Delta T}{r_0}\left\{(\pi t_d)^{-0.5} + \frac{1}{2} - \frac{1}{4}\left(\frac{t_d}{\pi}\right)^{0.5} + \frac{1}{8}t_d - \cdots\right\} \qquad (10\text{-}1)$$

$$q = f_2(t_d) \cdot \Delta T = \frac{2k\Delta T}{r_0}\left\{\frac{1}{\ln(4t_d) - 2\gamma} - \frac{\gamma}{[\ln(4t_d) - 2\gamma]^2} - \cdots\right\} \qquad (10\text{-}2)$$

Here, k is thermal conductivity (W m^{-1} K^{-1}), r_0 is the wellbore radius (m), and γ is the Euler constant (0.57722).

The heat transfer functions f_1 and f_2 express the amount of heat flux with time due a unit temperature difference. As shown in *Zhang et al.* [2011], the heat transfer functions f_1 and f_2 are approximately the same at the dimensionless time $t_d = 2.8$. Therefore, $t_d = 2.8$ is considered the critical dimensionless time to switch from f_1 to f_2.

During fluid injection and production, and as a result of the heat exchange processes, temperature changes continuously over time at any point within the wellbore and at the wellbore-formation interface. Based on superposition, the radial heat flux across each wellbore element to the surrounding formation is a time-convolution result of varying temperature. The discretized form at each time step can be expressed by the following:

$$q_{total} = \sum_{i=1}^{d-1} f(t_d - t_i) \cdot \Delta T(t_i) \qquad (10\text{-}3)$$

Here, t_d represents the current time after d time steps, and t_i represents the time after i time steps; the function f is f_1 if $(t_d - t_i) \leq 2.8$, and f_2 if $(t_d - t_i) > 2.8$. The temperature difference $\Delta T(t_i)$ is the temperature in the well at time step i, minus the formation temperature at the interface at the previous time step, i.e., $\Delta T(t_i) = T_w(t_i) - T_f(t_{i-1})$.

To implement the solution into the TOUGH2 simulator, we need to calculate q_{total} and its derivative at each time step, and incorporate these two terms into the heat balance equation and the corresponding linearized form, which is needed for the implicit solution of the fully coupled system of mass and heat flow equations in the well. This requires the algorithm to

store the temperature history for each wellbore element. This may be problematic if the time history becomes very long, which increases the computational demand of the time-convolution approach, and potentially reaches the limit of the computer's storage capacity. To mitigate this problem, a maximum number of time steps can be defined, beyond which the contributions from earlier temperature changes are lumped into a single term, which is calculated based on the time-weighted average of the individual temperature changes $\Delta T(t_i)$ and associated times t_i.

To make the algorithm flexible for handling various wellbore configurations and thermal conditions in the rock formation, the code gives the user an option to choose between uniform or depth-dependent formation properties, wellbore radii, and geothermal gradients.

The radial semi-analytical heat exchange model is currently incompatible with the option to perform semi-analytical linear heat exchange with confining beds (see *MOP(15)*=1 in *Pruess et al.* [1999]). There are two options (selected by *MOP(15)*=5 or 6, respectively) to invoke radial heat exchange:

Option 1 (MOP(15)=5): Constant Well and Formation Properties

Provide a material named QLOSS with the following parameters:

DROK: Rock grain density [kg/m^3] of formation near well
POR: Well radius [m]
PER(1): Reference elevation [m]; specify *Z* coordinate in block ELEME, Columns 71–80
PER(2): Reference temperature [°C]
PER(3): Geothermal gradient [°C/m]
CWET: Heat conductivity near well [W/kg °C] of formation near well
SPHT: Rock grain specific heat [J/kg °C] of formation near well

Option 2 (MOP(15)=6): Variable Well and Formation Properties

Provide an external file named *radqloss.dat* with information in the following format:

First line:
NMATQLOSS: number of elevations with geometric and thermal data
Provide *NMATQLOSS* lines with the following data in free format:

Elevation [m], well radius [m], initial temperature [°C], *CWET*, *DROK*, *SPHT*

Between elevations, properties are calculated using linear interpolation. Figure 45 shows and example input file using Option 1, for the simple heat injection into a single gridblock [*Zhang et al.*, 2011].

```
Semi-analytical radial heat exchange
ROCKS----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
WELL0          2650.      0.01    1.E-09    1.E-09    1.E-09      2.10     1000.
QLOSS          2650.      0.05    -.5000       20.      0.00      2.10     1000.

ELEME----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
A1  1              10.7854E-020.3142E+00          0.2500E-01          -.5000E+00

CONNE----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8

MULTI----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
    2    3    2    6
PARAM----1 MOP: 123456789*123456789*1234 ---*----5----*----6----*----7----*----8
   29999    99991000001000020054  1
             5.0e+10   1.0e+03
    1.e-05      1.e0                                    1.e-7
           1.013E+05             0.20e+2              0.0e+0

GENER----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
A1  1INJ 1                COUNT     HEATA
         0.0E7          1.0E7          2.0E7          3.0E7
         4.0E7          5.0E7          6.0E7          7.0E7
         8.0E7          9.0E7          1.0E8          5.e10
         8.e+1          2.e+1          8.e+1          2.e+1
         8.e+1          2.e+1          8.e+1          2.e+1
         8.e+1          2.e+1          8.e+1          2.e+1
         8.e+5          2.e+5          8.e+5          2.e+5
         8.e+5          2.e+5          8.e+5          2.e+5
         8.e+5          2.e+5          8.e+5          2.e+5

ENDCY----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
```

Figure 45. TOUGH2 input file for testing radial heat exchange option.

11 One-Dimensional Deformation Calculation (Block SUBSI)

Approximate, one-dimensional deformations are calculated based on porosity changes induced by changes in fluid pressure and/or temperature. No stress-strain calculation is performed. Relative porosity changes calculated by TOUGH2 expand or contract the distance between elements belonging to the same column. These changes are then integrated along the column to calculate new element coordinates. Volumes and nodal distances are not changed, i.e., the TOUGH2 mesh is not dynamically adapted.

Columns and their orientations are defined by a new parameter *ISOSUBSID*. A column consists of all elements that are linked by connections with *ISO* = *ISOSUBSID*. It is assumed that the TOUGH variable *ISO* with values of 1, 2, and 3 refer to connections that are (approximately) aligned with the X- Y-, and Z-axis, respectively. The default for *ISOSUBSID* is 3, creating vertical columns. The resulting deformation at the top of each column thus refers to subsidence (negative) or uplift (positive).

Each element in the mesh is assigned to a column and ordered according to its distance from a reference coordinate (variable *SUBSREFLEVEL*). Deformations at this reference point are defined as being zero. By default, *SUBSREFLEVEL* is set to a large negative value. For vertical columns, this means that a no-deformation boundary condition is defined at the bottom of the model.

No stress-strain calculation is performed, and there is no mechanical interaction between neighboring columns, i.e., the entire volumetric strain is assumed to lead to deformation in only one direction (defined by *ISOSUBSID*).

To invoke this calculation, non-zero values must be given for pore compressibility (TOUGH2 variable *COM*) or pore expansivity (TOUGH2 parameter *EXPAN*) along with a new block labeled by keyword SUBSI. On the line following keyword SUBSI, the (optional) variables *ISOSUBSID* and *SUBSREFLEVEL* can be given in format (I5,5X,E10.4).

The results of this simplified, one-dimensional deformation calculation are reported in a separate output block in the standard TOUGH2 output file, or in CSV format using label SUBSIDANCE in block OUTPU (see Section 3.1).

Element-related one-dimensional deformations can be extracted for iTOUGH2 analyses using iTOUGH2 command >> SUBSIDANCE.

12 One-Dimensional Hydrogeomechanics (Block LOAD)

Following *Walsh et al.* [2012], the effects of an external, one-dimensional, vertical stress change on porosity are added to TOUGH2. Such enforced stress and porosity changes result in pore-pressure changes and related fluid redistributions. The model requires specification of a material-dependent loading efficiency and a one-dimensional specific storage coefficient. The underlying assumptions of horizontally bedded formations and uniaxial strain limit the applicability of this model to very simple systems in which changes in mechanical loading occur over a large area (e.g., due to continental glaciation, laterally extensive erosion and deposition events, Earth tides, and atmospheric pressure fluctuations).

Hydrogeomechanical coupling under a homogeneous and laterally extensive load is implemented within the mass accumulation term. Porosity (ϕ) in TOUGH2 is not constant but is updated at the end of each iteration to account for changes in pressure. The expression for the updated porosity for the current time step (ϕ_t), including hydro-geomechanical effects, is

$$\phi_t = \phi_{t-1} + \phi_{t-1} \cdot c_\phi \cdot dp - \phi_{t-1} \cdot S_{S-1D} \cdot \zeta \cdot d\sigma_{zz} \tag{12-1}$$

where

ϕ_{t-1} : porosity (-) at previous time step
c_ϕ : pore compressibility (Pa^{-1}), variable *COM* in block ROCKS.2
dp : pressure change (Pa) during the time step
S_{S-1D} : one-dimensional specific storage coefficient (Pa^{-1})
ζ : one-dimensional loading efficiency (-)
$d\sigma_{zz}$: change in vertical loading (Pa) during time step (positive for increased load)

The second term on the right-hand side of Eq. (12-1), i.e., $\phi_{t-1} \cdot c_\phi \cdot dp$, represents the change in porosity due to the change in pore pressure during a time step; this term is implemented in standard TOUGH2. The third term, $\phi_{t-1} \cdot S_{S-1D} \cdot \zeta \cdot d\sigma_{zz}$, is the hydromechanical term that represents the change in porosity due to the change in vertical load applied during a time step. The variables that are unique to the hydromechanical formulation are the one-dimensional loading efficiency (ζ), the change in vertical load ($d\sigma_{zz}$), and the one-dimensional (uniaxial) specific storage coefficient (S_{S-1D}). The hydromechanical capability requires the one-dimensional loading efficiency to be defined for each material type. This parameter is used to determine what percentage of the applied vertical stress is borne by the pore-fluids; it is defined as:

$$\zeta = \frac{\beta \cdot (1+\nu)}{3 \cdot (1-\nu) - 2\alpha\beta(1-2\nu)} \tag{12-2}$$

where

α : Biot-Willis coefficient (-)
β : Skempton coefficient (-)
ν : Poisson ratio (-)

The Skempton coefficient can be related to geomechanical parameters as follows:

$$\beta = \frac{\left(\frac{1}{K}-\frac{1}{K_S}\right)}{\left(\frac{1}{K}-\frac{1}{K_S}\right)+\phi\left(c_w-\frac{1}{K_S}\right)} \tag{12-3}$$

where

K : drained bulk modulus (Pa)
K_s : solid grain bulk modulus (Pa)
c_w : water compressibility (Pa^{-1}), calculated internally

The three-dimensional specific storage coefficient is given by:

$$S_{S-3D} = \left(\frac{1}{K}-\frac{1}{K_s}\right) + \phi\left(c_w-\frac{1}{K_s}\right) \tag{12-4}$$

The one-dimensional specific storage coefficient is given by:

$$S_{S-1D} = S_{S-3D}(1-\lambda\,\beta) \tag{12-5}$$

where

$$\lambda = \frac{1\alpha(1-2\nu)}{3(1-\nu)} \tag{12-6}$$

The product of the one-dimensional loading efficiency and the specific storage coefficient, $S_{S-1D}\zeta$, is independent of fluid composition and can be approximated as

$$S_{S-1D}\zeta = \frac{\left(\frac{1}{K}-\frac{1}{K_S}\right)(1+\nu)}{3(1-\nu)} = \phi(c_\phi + c_w) \tag{12-7}$$

where

c_ϕ : pore compressibility

The pore compressibility should be chosen judiciously as

$$c_\phi = \frac{S_{S-1D}}{\phi} - c_w \tag{12-8}$$

In TOUGH2, there are two options to provide the one-dimensional loading efficiency ζ. Material-dependent parameters must be given in Columns 51–80 of block `ROCKS.1.2`, which is invoked by setting `NAD` ≥ 1.

Option 1: The one-dimensional loading coefficient ζ is internally calculated as a function of drained bulk modulus *K* [Pa], the Poisson ratio ν, and porosity using Eq. (12-2). The Biot-Willis coefficient α is assumed to be 1.0 (incompressible grains). Moreover, the pore compressibility c_ϕ is also internally calculated using Eq. (12-8). Option 1 is invoked if non-zero values are provided for *K* and ν.

K : Drained bulk modulus [Pa], Columns 51–60
ν : Poisson ratio [-], Columns 61–70

Option 2: The one-dimensional loading coefficient ζ is directly provided as a user-specified material property.

ζ : One-dimensional loading coefficient [-], Columns 71–80

To invoke one-dimensional hydromechanical coupling, the user must specify, for each material, the loading efficiency ζ in record ROCKS.2, columns 71–80 (set *NAD* ≥ 1). Moreover, the time-dependent load must be specified in the input file after keyword LOAD. (Note that block LOAD must be specified *after* the optional block TIMBC described in Section 6.1.)

Record LOAD.1

Free format
LOADDEF

LOADDEF : Number of loading functions defined. The total load is the sum of all loading functions. Defining multiple functions is convenient as certain loads (e.g., Earth tides) are a superposition of multiple harmonic functions.
Repeat the following entries *LOADDEF* times.

Record LOAD.2

Format(A9,1X,I10)
BCELM, NBCP

BCELM : String indicating type of loading function (start in Column 1)

TABLE : Table with *NBCP* rows of time t (s) vs. load σ (Pa). Loads will be linearly interpolated between table entries. For simulation times before the first or after the last time specified in the table, the load is set equal to the first or last load, respectively. Loading times specified in this block can be honored by setting *MOP2*(21) = 1 in block MOMOP (see Section 2.1).

POLYNOM : Polynomial of degree (*NBCP* – 1):

$$\sigma_{zz}(t) = \sum_{i=0}^{NBCP-1} a t^i \qquad (12\text{-}9)$$

SINE : Sinusoidal function, i.e.,

$$\sigma_{zz}(t) = \sigma_0 \cdot sin\left(2\pi \frac{t-t_0}{t_p}\right) \text{ for } t_B \le t \le t_E \qquad (12\text{-}10)$$

For simulation times before t_B or after t_E, the load is set equal to $\sigma_{zz}(t_B)$ or $\sigma_{zz}(t_E)$, respectively.

SQRT : Square-root function, i.e.,

$$\sigma_{zz}(t) = \sigma_0 \cdot \sqrt{\left(\frac{t-t_0}{t_p}\right)} \text{ for } t_B \le t \le t_E \qquad (12\text{-}11)$$

For simulation times before t_B or after t_E, the load is set equal to $\sigma_{zz}(t_B)$ or $\sigma_{zz}(t_E)$, respectively.

EXP : Exponential function, i.e.,

$$\sigma_{zz}(t) = \sigma_0 \cdot e^{t_p(t-t_0)} \text{ for } t_B \le t \le t_E \qquad (12\text{-}12)$$

For simulation times before t_B or after t_E, the load is set equal to $\sigma_{zz}(t_B)$ or $\sigma_{zz}(t_E)$, respectively.

NBCP : Number of table entries if *BCELM* = 'TABLE', otherwise number of function coefficients provided in record LOAD.3.

Record LOAD.3 (for *BCELM* = 'TABLE')

Free format
(*TIMBCV*(I), *PGBCEL*(I)), *I*=1,*NBCP*

TIMBCV(I) : Time (s)
PGBCEL(I) : Load (Pa)

Repeat *NBCP* times.

Record LOAD.3 (for *BCELM* ≠ 'TABLE')

Free format
PGBCEL(*I*)), *I*=1,*NBCP*

PGBCEL(I) : Function coefficients

For *BCELM* = 'PLOYNOM', provide *NBCP* coefficients a_i
For *BCELM* = 'SINE', 'SQRT' or 'EXP', the coefficients are:

PGBCEL(1) = σ_0 (Pa)
PGBCEL(2) = t_p (sec)
PGBCEL(3) = t_0 (sec) (optional)
PGBCEL(4) = t_B (sec) (optional)
PGBCEL(5) = t_E (sec) (optional)

An example is given in Figure 46. Relevant input parameters are highlighted.

```
Terzaghi
ROCKS----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
SHALE    1     2650.     .1000                    2.040E-15      2.51      920.
 1.860E-07                                                                  0.8
BOUND    0     2650.     .9900                    2.040E-15      2.51   100000.

PARAM----1--MOP:123456789012345678901234----*----5----*----6----*----7----*----8
   1 100       11000009000000000400003000
 0.000E+00 3.1536E11 1.000E+02
    1.0E-7
 100000.000000000000   0.000000000000000  20.000000000000000
MOMOP----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
2                  1
MULTI----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
    2    2    2    6
ELEME----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
IN             BOUND .1000E+50                    .0000E+00
    1   49    1SHALE .2000E+02                    .2000E+02

CONNE----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
IN       1                   3 .1000E-10 .1000E+02 .1000E+01
    1    2   48    1    1    3 .1000E+02 .1000E+02 .1000E+01

START----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
INCON----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8

GENER----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8

LOAD ----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
    1
TABLE              2
0.0   0.0
1.0   3E6

ENDCY----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
```

Figure 46. TOUGH2 input file for simulating one-dimensional hydromechanical coupling.

13 Non-Darcy Flow Using Forchheimer Equation (Block FORCH)

The equation governing flow in partially saturated porous media as implemented in TOUGH2 is based on the multiphase version of the Darcy-Buckingham law:

$$\mathbf{F} = \rho\mathbf{u} = -k\frac{k_r\rho}{\mu}\left(\nabla P - \rho\mathbf{g}\right) \tag{13-1}$$

Here, $\mathbf{F}$ [kg s^{-1} m^{-2}] is the Darcy mass flux, $\mathbf{u}$[m s^{-1}] is the average velocity, ρ [kg m^{-3}] is the density, k [m^2] is the absolute permeability, k_r [-] is the relative permeability, which is a function of saturation, μ [Pa s] is the dynamic viscosity, ∇P [Pa m^{-1}] is the pressure gradient, and $\mathbf{g}$ [m s^{-2}] is the gravity vector.

Forchheimer [1901] empirically found the relationship between the gradient of the water potential $\nabla\Phi = (\nabla P - \rho\mathbf{g})$ [Pa m^{-1}] and average velocity to be:

$$-\nabla\Phi = \frac{\mu}{k\cdot k_r}\mathbf{u} + \beta\rho|\mathbf{u}|\mathbf{u} \tag{13-2}$$

The second term on the right-hand side of Eq. (13-2) accounts for inertial forces. On the pore scale, as velocity increases, flow tends to locally separate from the tortuous pore walls, leading to local vortices and countercurrent flow regimes. The coefficient β [m^{-1}] is a function of effective permeability ($k\cdot k_r$), phase content ($\theta = \phi S$, where ϕ is porosity and S is the saturation of the considered fluid phase), and tortuosity ($\tau = L/l$, where L is the macroscopic flow distance, and l is the length of the corresponding microscopic, tortuous flow path). This dependence indicates a transition from viscous-dominated to form-dominated flow as the Reynolds number ($\mathrm{Re} = \rho u\sqrt{K}/\mu$, where K is a form factor) increases beyond a critical value. The parameter β has been referred to as the non-Darcy flow coefficient, the Forchheimer or inertial resistance coefficient, or the turbulence factor. Note, however, that deviations from Darcy's law are observed at Reynolds numbers that are at least one order of magnitude smaller than those where actual turbulence occurs. We will refer to β as the non-Darcy flow coefficient, which can be written in a general, parameterized form as follows:

$$\beta = A_1\cdot\frac{1}{(k\cdot k_r)^{A_2}}\cdot\frac{1}{\theta^{A_3}}\cdot\frac{1}{\tau^{A_4}} \tag{13-3}$$

Table 13 summarizes various correlation models for the non-Darcy flow coefficient as implemented in iTOUGH2. Most of the original models were developed for single-phase gas flow or gas flow at immobile water saturation. They are generalized here to two-phase conditions by replacing absolute permeability k with effective permeability, $k_{eff} = k\cdot k_r(S)$, and porosity ϕ with phase content, $\theta = \phi S$. Moreover, it is assumed that the Forchheimer equation applies to both liquid and gas flow. (Note that the user can restrict its applicability to gas flow only, see Section 4.1). The appropriateness of these extensions is not examined here.

Table 13. Parameters for Non-Darcy Flow Coefficient β

#	A_1	A_2	A_3	A_4	**Reference**
1	*FORCH(1)*	0	0	0	β = user-specified constant
2	5.00×10^{-3}	0.5	5.5	0	*Geertsma* [1974]
3	1.60×10^{-13}	1.64	0	0	*Frederick and Graves* [1994], Eq. (13-3)
4	2.53×10^{-13}	1.6	0.404	0	*Frederick and Graves* [1994], Eq. (13-4)
5	3.81×10^{-13}	1.55	1.0	0	*Frederick and Graves* [1994], Eq. (13-5)
6[&]	-	-	-	-	*Frederick and Graves* [1994], Eq. (13-6)
7	3.18×10^{-9}	1.25	0.75	0	*Janicek and Katz* [1955]
8	1.11×10^{-12}	1.55	0	0	*Jones* [1987]
9	2.89×10^{-6}	1.0	1.0	1.0	*Liu et al.* [1995]
10	2.66×10^{-6}	0.98	0.29	3.35	*Thauvin and Mohanty* [1998]
11	2.16×10^{-16}	1.88	0.449	0	*Coles and Hartman* [1998]
12	4.55×10^{-6}	1.023	0	1.943	*Cooper et al.* [1999]
13	1.42×10^{-1}	0.5	1.5	0	*Ergun* [1952]
14	*FORCH(1)*	*FORCH(2)*	*FORCH(3)*	*FORCH(4)*	User specified

The following conversion factors were used to convert, as applicable, the correlations given in the original references to SI metric units:

Darcy $\times 9.869233\times10^{-13}$ = m^2
Feet $\times 0.3048$ = m

Most of the original correlations were developed for single-phase gas flow. An extension to two-phase conditions is implemented by replacing absolute permeability with effective permeability, $k_{eff} = k \cdot k_r$, and porosity with phase content, $\theta = \phi \cdot S$.

The parameters A_1 through A_4 apply to the following general form for the non-Darcy flow coefficient:

$$\beta = A_1 \cdot (k \cdot k_r)^{-A_2} \cdot \theta^{-A_3} \cdot \tau^{-A_4}$$

[&] Model No. 6 [*Frederick and Graves*, 1994, Eq. (6)]: $\beta = \frac{3.2808}{\theta^2} \cdot \exp\left[45 - \sqrt{3205.71 + 81 \cdot \ln\left(k \cdot k_r / \theta\right)}\right]$

The non-Darcy flow effects can be combined with the Klinkenberg gas-slip flow effect and expressed as a pressure- and velocity-dependent correction to the absolute permeability:

$$\tilde{k} = k\left[\frac{1+\left(\delta_g\, b/P\right)}{1+F_0}\right] \tag{13-4}$$

Here, b [Pa] is the Klinkenberg factor, which accounts for enhanced gas slip flow that occurs when the mean free path of the gas molecules is large relative to the characteristic dimension of the pores [*Klinkenberg*, 1941]. Slip flow is important at low pressures P [Pa] and in small pores, when a significant fraction of molecular collision is with the pore wall rather than with other gas molecules. The Kroenecker delta δ_g is set to 1 if the fluid is gas, and zero if the fluid is liquid.

The coefficient F_0 describes the velocity-dependent reduction in permeability as represented by the Forchheimer equation (Eq. 13-2), and is given by:

$$F_0 = \beta\rho\frac{k\cdot k_r}{\mu}|\mathbf{u}| \tag{13-5}$$

Note that the Klinkenberg effect increases gas flow rates, whereas the non-Darcy flow effects of the Forchheimer equation tends to decrease flow rates. Furthermore, conditions under which the Klinkenberg gas slip effect becomes significant are often opposite to those under which the second term of the Forchheimer equation becomes relevant.

Tortuosity effects are considered to have a porous-medium-dependent part and a saturation-dependent part. The resulting tortuosity factor τ entering some of the non-Darcy flow coefficient models (see models with $A_4 \neq 0$ in Table 13) is evaluated using one of the following options:

$$\tau = \phi^{1/3} S^{10/3} \tag{13-6a}$$

$$\tau = \tau_0 \cdot S \tag{13-6b}$$

$$\tau = \tau_0 \cdot k_r \tag{13-6c}$$

$$\tau = \tau_0 \tag{13-6d}$$

Eq. (13-6a) represents the *Millington and Quirk* [1961] model, which yields non-zero tortuosity coefficients as long as phase saturation is non-zero. A linear correlation between tortuosity and saturation is given by Eq. (13-6b). One might also argue that flow is most tortuous (i.e., the tortuosity factor approaches zero) when the fluid phase becomes discontinuous, suggesting that saturation-dependent tortuosity should be related to relative permeability (Eq. 13-6c). Finally, Eq. (13-6d) describes a constant, saturation-independent tortuosity factor.

Non-Darcy flow based on the Forchheimer equation is implemented as follows:

- The new keyword FORCH is followed by user-specified input for the non-Darcy flow coefficient model (Eq. 13-2).
- The non-Darcy flow coefficient β is evaluated for each gridblock in a connection, and then averaged according to the selected weighting scheme.
- The quadratic Forchheimer equation (Eq. 13-2) is solved for the absolute value of the non-Darcy velocity **u** using the following, numerically stable formula:

$$\mathbf{u} = \frac{-2\nabla\Phi}{\frac{\mu}{k \cdot k_r} + \sqrt{\left(\frac{\mu}{k \cdot k_r}\right)^2 + 4\beta\rho|\nabla\Phi|}} \tag{13-7}$$

  Note that as $\beta \to 0$, Eq. (13-7) converges to the solution of the Darcy flow equation.
- The non-Darcy flow coefficient β is calculated as a function of absolute permeability, relative permeability, porosity, saturation, and tortuosity (Eq. 13-3).

The simulation of non-Darcy flow according to the Forchheimer equation is invoked by providing a new data block following keyword FORCH in the TOUGH2 input file:

FORCH Invokes flow calculation according to the Forchheimer equation (Eq. 13-2); introduces information on non-Darcy flow coefficient model (Eq. 13-3) and weighting scheme.

Record FORCH.1

FORMAT (2I5, 4E10.4)

IFORCH, *MFORCH*, (*FORCH(I)*, *I*=1,4)

IFORCH Integer parameter to choose type of non-Darcy flow coefficient model (see Table 13, Column 1). If a negative number is given, the Forchheimer equation is applied to gas flow only; liquid flow is governed by Darcy's law (Eq. 13-1).

MFORCH Determines interface weighting scheme for non-Darcy flow coefficient β:

MFORCH = 0: Upstream weighting (recommended)
MFORCH = 1: Upstream weighting with *WUP* (see record PARAM.3)
MFORCH = 2: Harmonic weighting
MFORCH = 3: Geometric average
MFORCH = 4: Arithmetic average

FORCH(I) *I* = 1, …, 4; user-specified parameters for non-Darcy flow coefficient model (see Table 13, Columns 2–5; only needed for models *IFORCH* = 1 and 14).

The porous-medium-dependent tortuosity factor τ_0 and the tortuosity model (see Eq. 13-6a–d) are selected through variable *TORTX* (Columns 31–40 of record ROCKS.1.1, see *Pruess et al.* [1999]) as follows:

TORTX	= 0:	Millington-Quirk model (Eq. 13-6a)
	> 0:	Tortuosity depends on relative permeability (Eq. 13-6c)
	> 0 and *DIFF0* < 0:	Tortuosity is a linear function of saturation (Eq. 13-6b); *DIFF0* is provided in record PARAM.1, Columns 41–50, see *Pruess et al.* [1999].
	< 0:	Anisotropic tortuosity factors

The Klinkenberg factor b [Pa] for enhancing gas phase permeability k_g over liquid phase permeability k_l according to the relationship $k_g = k_l \cdot (b/P_g)$ (see also Eq. 13-5) is provided through variable *GK* (Columns 41–50 of record ROCKS.1.1, see *Pruess et al.* [1999]).

The non-Darcy flow coefficients, i.e., variable *FORCH(IPAR)*, can be estimated by iTOUGH2 using command >> FORCHHEIMER.

14 Klinkenberg Factor Correlations

Gas-slip flow is described by the equation $k_g = k(b/P_g)$, whereby the Klinkenberg factor b [Pa] is given through variable *GK* (Columns 41–50 of record ROCKS.1.1). If a negative value is provided, *GK* is internally calculated as a function of absolute permeability according to Table 14.

Table 14. Correlations for the Klinkenberg Gas-Slip Factor

GK	**Correlation**	**Reference**
>0	b = *GK*	*Pruess et al.* [2012]
-1.0	$b = 0.112\ k^{-0.39}$	*Heid et al.* [1950]
-2.0	$b = 0.977\ k^{-0.33}$	*Jones and Owens* [1979]
-3.0	$b = 1.071\times10^{-3}\ (k/\phi)^{-0.53}$	*Sampath and Keighin* [1982]
-4.0	$b = 9.615\times10^{-3}\ (k/\phi)^{-0.5}$	*Florence et al.* [2007]
-5.0	$b = 9.400\times10^{-3}\ (k/\phi)^{-0.5}$	*Civan* [2010]
-6.0	$b = 0.026\ k^{-0.43}$	*Letham and Bustin* [2016]

15 Effective Diffusion Coefficients Based on Archie's Law

TOUGH2 has the capability to simulate binary diffusion, accounting for phase partitioning under multiphase conditions. Multiple options exist to calculate the effective diffusion coefficients as a function of pressure, temperature, bulk fluid diffusion coefficient, porosity, tortuosity, and saturation (see Appendix D of Pruess et al., 1999; 2012).

Van Loon (2014) and Van Loon and Mibus (2015) suggest using an extended version of Archie's law (originally developed for electrical resistivity of porous rocks; Archie, 1942) to reproduce and estimate effective diffusion coefficients, specifically those measured in argillaceous rocks. The extended Archie's law was further modified to account for multiphase conditions; it can be written using TOUGH2-compatible notation as:

$$\Sigma_\beta^\kappa = \rho_\beta \left[d_\beta^\kappa \cdot \varepsilon^m \cdot S_\beta^n + A \cdot \left(\varepsilon \cdot S_\beta \right)^B \right] \tag{15-1}$$

Here, Σ_β^κ is the effective multiphase diffusion coefficient; ρ_β is the density of phase β; d_β^κ is the diffusion coefficient of component κ in bulk fluid phase β; m, n, A and B are empirical coefficients, ε is the diffusion-specific porosity, and S_β is the volumetric saturation of phase β. The parameters m and n are sometimes referred to as the cementation index and saturation index, respectively. Note that Van Loon (2014) does not explicitly differentiate between porosity and saturation effects in the definition of "accessible porosity" or assumes fully liquid saturated condition. Consequently, a lumped accessible porosity is defined, with a single exponent, labelled *m1*, in the first term. Moreover, parameter B is termed *m2* in Van Loon (2014). The user may consider the diffusion-accessible porosity ε to be the same as the porosity ϕ used in the accumulation term solved by TOUGH2; alternatively, a diffusion-specific accessible porosity ε value can be specified.

To model diffusion based on Archie's law, the user needs to supply material-specific or element-specific values for the empirical parameters m, n, A and B, as well as the diffusion-specific accessible porosity ε (if different from bulk porosity ϕ). Material-specific parameter values (variables *ARCHIEM*, *ARCHIEN*, *ARCHIEA*, *ARCHIEB*, and *ARCHIEP*) are to be provided on a separate line in block ROCKS, Columns 31–80 in (5E10.4) format (see Figure 47). This additional line is read if a negative value is provided for variable *TORTX* in record ROCKS.1.1, Columns 31–40 (see also Section 5.8). Element-specific Archie parameters are provided following the instructions of Section 5.3; see also Table 6.

Figure 48 shows the effective diffusion coefficient calculated as a function of accessible porosity based on Archie's law. The parameters of Archie's law were chosen as $d_\beta^\kappa = 2 \times 10^{-9}$ m^2 s^{-1}, m = 2.4, n = 0.0, $A = 10^{-11}$ m^2 s^{-1}, B = 1.0, and ε varying between 0.001 and 1.0. With this parameter set, the effective diffusion coefficient is identical to that shown in Figure 3 of Van Loon and Mibus (2015). Stronger non-linearities result under two-phase conditions with non-zero values for n and B.

```
samARCHIE: Archie's law for diffusion
MULTI----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
    2    3    2    8
ROCKS--NAD----*----2----*----3----*TORTX----*----5----*----6----*----7----*----8
ARCHI    1      2650.      .3000 1.000E-12 1.000E-12 1.000E-12      1.50       800.
                                  -1.00
                                   2.40       0.0   1.0E-11        1.0        0.1
                                      m         n         A          B    epsilon
DIFFU----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
     2.e-9     2.e-9
     2.e-9     2.e-9
ENDCY----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
```

Figure 47. Excerpt of TOUGH2 input file showing Archie's law parameters.

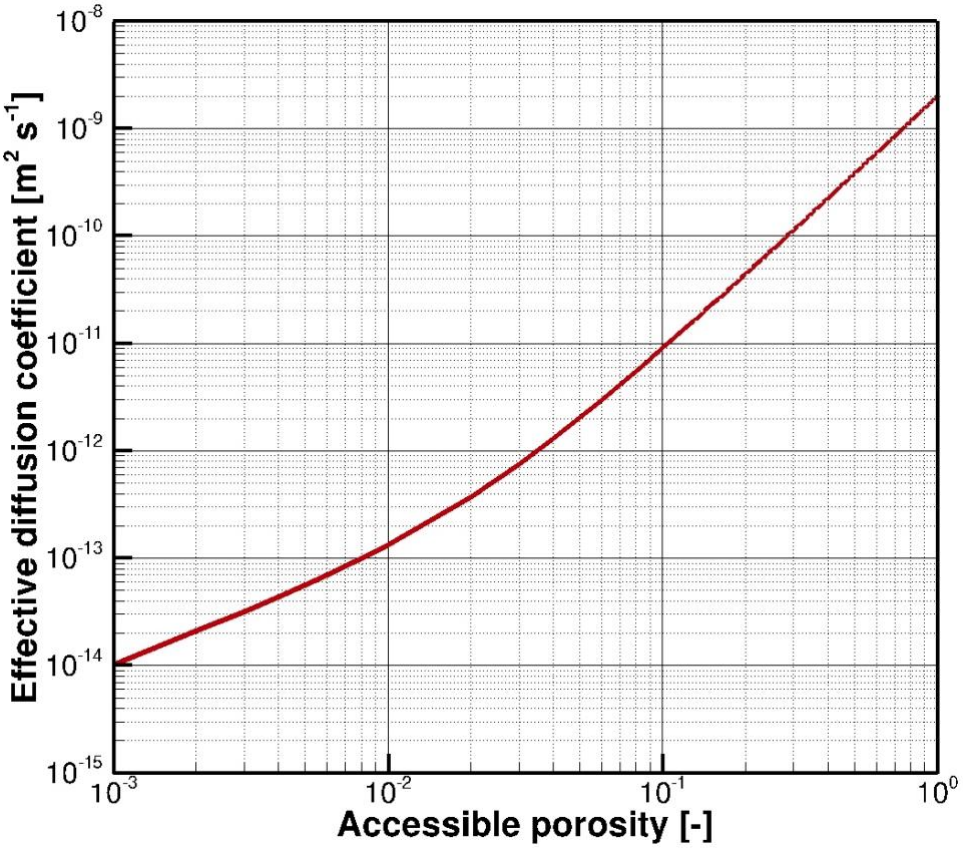

Figure 48. Effective diffusion coefficient as a function of accessible porosity based on Archie's law.

16 Tensile Clay Fracturing Model

The creation of dilatant gas pathways is discussed in detail in *Nagra* [2004; Sections 3.1 and 4.2]. A macroscopic tensile fracture develops when the gas pressure is larger than the sum of the minimum principal stress and the tensile strength of the rock. The threshold pressure for dilatant gas flow is considered a material- and depth-dependent property related to the local stress field. Pressure-dependent pathway dilation leads to increased permeability and reduced capillary strength. Pathway dilation is implemented as follows:

- Calculate the threshold pressure p_d [Pa] for dilatant gas flow, which has a depth-dependent term and a saturation-dependent clay-swelling term (see Section 18):

$$p_d = p_d(z) + p_d(S_l) = [\chi \cdot (\epsilon - z) - \varepsilon] + [\zeta \cdot S_l^{\eta}] \tag{16-1}$$

where:

z : elevation [m.a.s.l.]
χ : lithostatic pressure gradient [Pa m^{-1}] (fixed at $\chi = 2.5\times10^4$)
ε : threshold overburden pressure for pathway dilation [Pa]
ϵ : surface elevation [m.a.s.l.; Z coordinate in `ELEME` block]
ζ : maximum swelling pressure [Pa]
η : empirical exponent [-]

- Calculate the vertical permeability k_v [m^2] as a function of p_d and pore pressure p:

$$k_v(p,z) = \begin{cases} k_{v,0} & p \le p_d \\ k_{v,0} + \beta \cdot (p - p_d)^{\alpha} & p > p_d \end{cases} \tag{16-2}$$

where:

$k_{v,0}$: undisturbed vertical permeability [m^2]
p : absolute pore pressure [Pa]
α : empirical exponent [-]
β : empirical coefficient [m^2 Pa^{-a}]

- Calculate the anisotropy ratio A [-] as a function of k_v:

$$A(k_v) = \frac{k_h}{k_v} = 5^{(k_{v,0}/k_v)^{\gamma}} \tag{16-3}$$

where:

γ : empirical exponent [-]

- Calculate the horizontal permeability k_h [m^2] as a function of A:

$$k_h(A) = A \cdot k_v \tag{16-4}$$

- Porosity and the parameters of the characteristic curves are considered constant; capillarity can be related to the permeability by setting `MOP2(6)` > 0 (see Table 3).
- Single- and two-phase flow within dilatant pathways is calculated using the standard TOUGH2 multi-phase process description. As an alternative, the Forchheimer equation may be selected (see Section 13).

Pathway dilation is invoked by setting *MOP2(35)* to 1 or 3 (see Table 3). The parameters are given in an extended set of relative-permeability parameters (following the standard relative-permeability parameters in slots 8–15); the parameters are s listed in Table 15. Reading the extended list (*RP(NMAT,i)*, *i* = 8–15) is initiated by setting *IRP* to the negative value of the relative permeability function number (see Figure 49 for an example).

Table 15. Parameters of Pathway Dilation and Clay Swelling/Shrinking Models

i in *RP(NMAT,i)*	**Variable**	**Symbol**	**[Units]**	**(Equation) Typical Value**
8	empirical exponent for pathway dilation	α	[-]	(16-2) 3.0
9	empirical coefficient for pathway dilation	β	[m^2 $Pa^{-\alpha}$]	(16-2) approx. $\sqrt{k^{\alpha}}$
10	empirical exponent for pathway dilation	γ	[-]	(16-3) 0.25
11[&]	fraction of swelling minerals	δ	[-]	(18-2) 0.85 for bentonite
12	threshold overburden pressure for pathway dilation	ε	[Pa]	(16-1) ~10^6
13	surface elevation	ϵ	[m]	(16-1) *Z* coordinate
14[&]	maximum swelling pressure	ζ	[Pa]	(16-1, 17-1) approx. $1/\alpha$
15[&]	empirical exponent	η	[-]	(17-1, 17-2, 18-1) ≤1.0
[&] Also see Sections 17 and/or 18				

```
Pathway Dilation and Clay Swelling
ROCKS----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
GBENT     2    2700.0       0.46 1.000E-18 1.000E-18 1.000E-19      1.15     880.0
   3.7E-09   1.5E-04       0.47
  -11            0.1       0.05
       3.0   1.0E-28       0.25       0.85     2.0E6      10.00     5.0E6      1.00
   11            1.67    1.0E+6      1.E09                -0.02                0.00
SBMIX     2    2700.0       0.38 1.600E-18 1.600E-18 1.600E-18      1.15     880.0
   2.1E-09   1.5E-04       0.47
  -11            0.1       0.05
       3.0   1.0E-27       0.25       0.00     1.0E6      10.00       0.0      1.00
   11            1.35    1.65E+5     1.E09                -0.02                0.00

MOMOP----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8
     1                          2
```

Figure 49. Relative-permeability parameter blocks with extended line holding parameters of pathway-dilation and clay-swelling models.

17 Gas Pathway Dilation Model for Clay

The dilatancy-controlled gas flow, also called pathway dilation (after *Horseman et al.*, [1996]), is a phenomenon which leads to a gas-driven microfracture process that increases interparticle porosity (by displacing clay particles) once a certain gas threshold pressure is exceeded. As gas preferentially flows in the largest pores, this rearrangement of the particle and thus pore-size distribution leads to a relative increase in gas permeability and dynamic reduction in gas entry value; the liquid is expected to be flowing within the clay (rather than the newly created gas pathway). The threshold pressure is defined as:

$$p_d = \zeta \cdot S_l^{\eta} + p_g + p_c(S_l) \tag{17-1}$$

(see Table 15 for specification of parameters ζ and η.) If the gas pressure, p_g, exceeds the threshold pressure, p_d, gas dilation is initiated. Because of the newly created pathway, which is assumed to be exclusively occupied by gas, the gas permeability is enhanced by a factor f_g. Two options are provided to estimate the change in interparticle porosity and/or increase in gas permeability as gas invades the clay-filled interparticle pore space.

In the first option, the clay-shrinking model described in Section 18 below is used, whereby f_g is given by the exponential term of Eq. (18-3).

In the second option, the factor f_g is a function of the amount by which the gas pressure exceeds the threshold pressure, i.e.,

$$f_g = log\left[\left(p_g - p_d\right)^{\eta}\right] + 1 \tag{17-2}$$

The first option is selected by setting `MOP2(35)` = 4; the second option is selected by setting `MOP2(35)` = 5 (see Table 3). In both options, the local capillary pressure is reduced by a factor of $1/\sqrt{f_g}$; permeability to liquid remains unchanged.

18 Clay Swelling and Shrinking

Clay-based materials (specifically bentonite) tend to swell and shrink during hydration or dry-out processes, resulting in saturation-dependent property changes. In the basic conceptual model of TOUGH2, permeability is assumed to be a constant property parameter (see Section 5.7 for specifying time-dependent permeabilities), which is experimentally determined or estimated for a given saturation condition under a known confining pressure. However, clay swelling and shrinking in response to hydration or dry-out changes the pore-size distribution and thus the permeability (as well as water retention and relative permeability characteristics), resulting in a dynamically changing, heterogeneous property distribution. The dynamic change of permeability in turn affects fluid flow in a complex feedback mechanism that can only be properly captured if constitutive relations are available that describe the dependence of flow properties on either pore-size distribution or (as a surrogate for the microscale properties) the system state (saturation, effective stress, etc.). Since the pore-size distribution critically depends on the effective stress (which in turn is a function of the total stress, thermal stress, swelling pressure, and fluid pressure), a fully coupled thermal-hydrological-mechanical-chemical simulation would be required to properly estimate time-dependent hydraulic properties of the deforming bentonite.

If such a fully coupled treatment is beyond the scope of a modeling study, property changes in response to clay hydration or dry-out can be approximately represented by using a highly simplified model that calculates permeability as a function of local liquid saturation. The correlation is local in that it is solely based on the saturation in a given computational grid block, i.e., it does not account for the impact of changes in the global stress field. The correlation implemented in iTOUGH2 is a modified version of the clay swelling model developed by *Xie et al.* [2007], which conceptualizes the bentonite as consisting of swelling and non-swelling minerals, interlayer pore volume (unavailable for macroscopic fluid flow), and interparticle pore volume, which is connected and facilitates fluid flow. Under confined conditions, wetting leads to an expansion of the interlayer pore volume at the expense of the interparticle pore volume, resulting in an increase in swelling pressure (which will be ignored as no stress calculation is performed) and a decrease in permeability, as the connected interparticle pore space available for fluid flow is compressed. Note that he total pore volume remains constant under constrained conditions during this pore-deformation process; however, the partitioning of the total pore volume between the interparticle and interlayer porosity changes. The interlayer porosity ϕ_{IL} is calculated as function of liquid saturation S_l as follows:

$$\phi_{IL} = S_l^{\eta} \cdot \delta \cdot \phi_{IL,\max} \tag{18-1}$$

Here, δ represents the volume fraction of expansive minerals, η is a dimensionless exponent, and $\phi_{IL,\max}$ is the maximum interlayer porosity, which is approximated as the initial porosity plus the maximum interlayer-porosity change that can be achieved by fully saturating the bentonite. This requires knowledge of the reference saturation state, which has to be provided through block `INDOM`. The flow-relevant, effective porosity under constrained conditions is therefore:

$$\phi_i = \phi_0 - \phi_{IL} \tag{18-2}$$

A minimum porosity $\phi_{IL,\min} = 0.05$ is enforced. The final step consists of correlating permeability to effective porosity:

$$k_i = k_0 \cdot \exp[8 \cdot (\phi_i - \phi_0)] \tag{18-3}$$

The permeability modified as a result of changed interparticle porosity is multiplied by the saturation-dependent relative permeability calculated using the standard van Genuchten formulation. Thus, bentonite wetting leads to an effective liquid permeability that is the result of two counteracting effects: the drop in available interparticle pore space available for fluid flow, and an increase in relative permeability due to increased liquid saturation (which gives the liquid phase access to larger pores with reduced flow resistance, a larger cross-sectional area, and a less tortuous flow path).

A saturation-dependent permeability due to clay swelling or shrinking can be invoked by setting *MOP2(35)* ≥ 2. The parameters are given in an extended set of relative-permeability parameters, as described in Section 16 with parameters listed in Table 15 above. Reading the extended list (*RP(NMAT,i)*, *i* = 11, 14, 15) is initiated by setting the relative permeability function number *IRP* to a negative value. An example is given in Figure 49 above.

19 Waste Degradation (Blocks SOURCE and TRACE)

19.1 Introduction

Radionuclides encapsulated in a solid waste form are slowly released into the aqueous phase (and may potentially partition into the gas phase) as the waste form degrades. In iTOUGH2, waste degradation and associated radionuclide releases can be simulated using (1) a fractional waste degradation model, or (2) degradation of a cylindrical waste form.

Radioactive decay of the radionuclide inventory encapsulated in the waste form is accounted for:

$$m(t) = m_0 \cdot e^{-\lambda t} \tag{19-1}$$

where m_0 [kg] is the initial mass, $\lambda = \frac{\ln(2)}{t_{1/2}}$ [s^{-1}] is the decay constant, and $t_{1/2}$ [s] is the radionuclide's half-life. The decay equation applies to all radionuclides, regardless whether they are encapsulated in the solid waste matrix or dissolved in the pore fluid.

In addition to radioactive decay, the mass of radionuclide in the waste form declines as isotopes are released from the waste form as it degrades. It is assumed that the radionuclides are dissolved in the homogeneous waste form. This means that the rate with which radionuclides become available for mobilization is determined by the waste degradation rate, a process referred to as congruent release.

The waste degradation rate is typically described as a fraction of the remaining waste mass in [s^{-1}] (see the fractional waste degradation model described in Section 19.2) or by a dissolution rate in [kg m^{-2} s^{-1}] acting on the exposed surfaces of the waste form (see the cylindrical waste degradation model described in Section 19.3). These normalized degradation rates q may be assumed constant in time or dependent on temperature and other factors (such a pH or an affinity term). The dependence on temperature is described by the Arrhenius equation:

$$q(T) = \Upsilon \cdot q_0 \cdot \exp\left[\frac{-E_a}{R} \cdot \left(\frac{1}{T} - \frac{1}{T_0}\right)\right] \tag{19-2}$$

where T [K] is temperature, q_0 is the rate constant at reference temperature T_0 [K], E_a [J mol^{-1}] is the activation energy, and R [J K^{-1} mol^{-1}] is the universal gas constant. The dimensionless factor Υ can be used to account for other effects such as geometric factors, pH dependence (e.g., $\Upsilon_{\mathrm{pH}} = 10^{-\eta \cdot pH}$, where η is an empirical coefficient), or affinity corrections (e.g., $\Upsilon_{\mathrm{Si(OH)_4}} = (1 - Q/K)$, where Q and K are the orthosilicic acid activity and solubility, respectively). Note that unlike temperature, Υ is a constant input parameter that does not depend on the solution variables of the TOUGH2 simulation.

The rates q and q_0 in Eq. 19-2 either represent a fractional degradation rate ω in [s^{-1}], or a waste dissolution rate r in [kg m^{-2} s^{-1}].

19.2 Fractional waste degradation model

Waste form degradation is described by a fractional degradation rate, ω [s^{-1}], which is the rate with which the remaining waste mass degrades per year. The fractional waste degradation rate can be obtained from degradation experiments as the slope

$$\omega = -\frac{\ln(m(t)/m(t+\Delta t))}{\Delta t} \tag{19-3}$$

The fractional degradation rate ω can be considered temperature dependent based on the Arrhenius equation (Eq. 19-2). The rate with which the mass of radionuclides in the waste form is reduced due to decay and congruent release is given by:

$$\frac{dm}{dt} = m(t) \cdot (\lambda + \omega) \tag{19-4}$$

The radionuclide mass encapsulated in the degrading waste matrix at time t is:

$$m(t) = m_0 \cdot (1 - IRF) \cdot e^{-\lambda t} \cdot e^{-\omega(t-t_f)} \tag{19-5}$$

where m_0 [kg] is the initial inventory at time zero, and IRF is the instant release fraction[ii], and t_f is the time of canister failure, when waste degradation is expected to be initiated. The time-dependent rate q_{RN} [kg s^{-1}], with which some of the radionuclide mass remaining in the solid waste matrix is released to the pore fluid due to waste form degradation, is given by:

$$q_{RN}(\mathrm{t}) = m(t) \cdot \omega \tag{19-6}$$

19.3 Cylindrical waste degradation model

The volume of a cylinder of radius R [m] and length L [m] is given by:

$$V = \pi R^2 L \tag{19-7}$$

The cylinder's volume changes as its radius and length are changed as follows:

$$dV = 2\pi RL\, dR + \pi R^2\, dL \tag{19-8}$$

The reduction in the radius and length due to degradation is given by:

$$dR = \frac{r}{\rho_{WF}}\, dt \tag{19-9}$$

$$dL = 2\frac{r}{\rho_{WF}}\, dt \tag{19-10}$$

where r is the dissolution rate in [kg m^{-2} s^{-1}], and ρ_{WF} [kg m^{-3}] is the density of the waste form. The volumetric waste degradation rate is therefore given as:

[ii] The radionuclide mass released instantaneously upon canister breach, $(m_0 \cdot IRF)$, either needs to be specified as an initial radionuclide mass fraction if the canister is modeled as fully containing until failure time t_f, or the mass fraction corresponding to $(m_0 \cdot IRF \cdot e^{-\lambda \cdot t_f})$ can be specified at time t_f using the iTOUGH2 command >> `RESTART TIME`).

$$q_{WF}(t) = \frac{dV(t)}{dt} = 2\pi \frac{r}{\rho_{WF}} (L(t) \cdot R(t) + R(t)^2) \quad (19\text{-}11)$$

If r is constant with time, the waste-form degradation rate is given by:

$$q_{WF}(t) = 2\pi r \left(3\left(\frac{r}{\rho_{WF}}\left(t - t_f\right)\right)^2 - (L + 4R)\frac{r}{\rho_{WF}}\left(t - t_f\right) + R(L + R) \right) \quad (19\text{-}12)$$

As before, t_f [s] is the time of canister failure, when waste degradation begins. If $L \gg R$, degradation from the two circular end faces of the cylinder can be ignored, and the equation simplifies to:

$$q_{WF}(t) = 2\pi L r \left(R - \frac{r}{\rho_{WF}} \cdot \left(t - t_f\right) \right) \quad (19\text{-}13)$$

If r is time-dependent (e.g., due to the time-varying and element-specific waste temperature), Eq. (19-11) is evaluated numerically for each element, with the cylinder's radius and length updated after each time step.

The waste form mass is calculated as:

$$m_{WF}(t) = m_{WF,0} - \int_{t_f}^{t} q_{WF}(\tau)\, d\tau \quad (19\text{-}14)$$

where $m_{WF,0}$ [kg] is the initial mass of the waste form. The radionuclide mass fraction in the waste form is:

$$X_{WF}^{RN}(t) = m_{\mathrm{RN},0} \cdot (1 - IRF) \cdot e^{-\lambda t} / m_{WF,0} \quad (19\text{-}15)$$

Finally, the congruent radionuclide release rate from the waste form is given by:

$$q_{RN}(\mathrm{t}) = q_{WF}(t) \cdot X_{WF}^{RN}(t) \quad (19\text{-}16)$$

For constant r and a cylinder that is relatively long, i.e., $L \geq 2R$, the waste form is fully degraded at time $t(q_{WF} = 0) = R \cdot \rho_{WF}/r$; for a more disk-shaped waste form, i.e., $L < 2R$, the waste form is fully degraded at time $t(q_{WF} = 0) = L/2 \cdot \rho_{WF}/r$.

19.4 Implementation of radionuclide release from degrading waste form

The rate of radionuclides released from a degrading waste form is implemented in iTOUGH2 as a material-specific source term (see Section 6.12). Parameters characterizing the waste form, waste form degradation model, and released radionuclide are described in blocks `ROCKS`, `GENER`, and `TRACE`.

In block `ROCKS`, a material needs to be defined that represents the waste form, with variable *`DROK`* specifying the waste form density ρ_{WF} (not relevant if the fractional degradation model is used).

In block `GENER`, variable *`SL`* of the source code name identifies the degradation model: the fractional waste degradation model is selected by setting *`SL`* = '`FDR`', and the cylindrical waste degradation model is selected by setting *`SL`* = '`CYL`'. The integer in variable *`NS`* refers to the order number of the waste form material as specified in block `ROCKS`. Variable `TYPE`

indicates the component being released, i.e., COM3 or COM4 if EOS7r is used, or TRCn if EOS1nT [*Finsterle*, 2021] is used.

Variable *GX* holds the waste degradation rate, either ω [s^{-1}] or r [kg m^{-2} s^{-1}], depending on the model selected through variable *SL*. Variable *EX* holds the volume-specific radionuclide mass [kg m^{-3}] initially present in the waste form (see Footnote ii on page 102 for instructions on how to treat an instantaneous release fraction, IRF). Note that since the source refers to a material type rather than a specific element, the mass must be given in units of [kg m^{-3}], i.e., the mass of radionuclides present in (and released from) each element representing the waste form will internally be scaled by the element's volume. Finally, variable *HG* holds the time of canister failure. While radioactive decay starts at time *TSTART* (see block PARAM.2), waste degradation starts at time *HG* = t_f. Finally, if *SL* = 'CYL', the initial radius of the cylindrical waste form is provided through variable *GMOD* (Columns 71–80 in block GENER.1).

Temperature dependence of waste form degradation can be included by specifying the Arrhenius parameters (see Eq. 19-2) in block TRACE (see *Finsterle* [2021]) for a (non-existing) *NK*s tracer. Variable *TRHENRY(NK)* holds the rate multiplication factor Υ; variable *TRARR(NK,1)* holds the pre-exponential factor q_0 (a non-zero value serves as the flag indicating that temperature-dependent degradation is selected; q_0 overwrites the rate given through variable *GX*); variable *TRARR(NK,2)* holds the activation energy E_a; and variable *TRMW(NK)* holds the reference temperature T_{0c} in [°C] (i.e., ($T_{0c} = T_0 - 273.15$)); if $T_{0c} = 0$, T_{0c} is set to infinity, and q_0 is the so-called intrinsic rate. Table 16 summarizes the input parameters for simulating waste degradation and congruent radionuclide release. Note that all parameters listed in Table 16 are adjustable during an iTOUGH2 sensitivity analysis, uncertainty propagation analysis, or inversion.

A first example verifies the implementation for constant degradation rates. Figure 50 shows the TOUGH2 input file, consisting of four disconnected elements, each representing a different waste form and isotope. The TRACE block shows that Tracer 1, which is Component 2, is stable, whereas Tracer 2, which is Component 3, has a decay constant that leads to the decay of 90% of the initial inventory after 1 million years (the corresponding half-life is about 300,000 years). The GENER block shows that the stable isotope (COM2) is injected into the first two elements, and a decaying radionuclide (COM3) is injected into the last two elements. The material of the first and third element decay using a fractional waste form degradation model (*SL* = 'FDR') with a fractional degradation rate ω = 3.6482×10^{-13} s^{-1}, which leads to a degradation of the waste form mass by a factor of 10 every 200,000 years. The waste form of the second and fourth element is assumed to be cylindrical (*SL* = 'CYL') with an initial radius of R = 0.138 m, a waste form density of 3000 kg m^{-3}, and a surface degradation rate of r = 2 × 10^{-11} kg s^{-1} m^{-2}. The inventory is calculated such that the total amount of initial radionuclide mass in the waste form is 1 kg for an element volume of 0.3 m^3 (note that this volume corresponds to a five-meter long cylinder of radius 0.138 m, i.e., in this example, the element represents a single canister; however, an element may represent more than one or only a fraction of a canister, which is the reason why the inventory is provided in kg m^{-3}; see Footnote ($) in Table 16). The canister fails after 100,000 years, at which time the waste form starts degrading.

The top row of Figure 51 shows the simulation results for the fractional waste degradation model. It confirms that the slope of the congruent radionuclide release rates reflect the combination of the specified fractional degradation rate and radioactive decay rates, and that the total mass at the end of the one million year simulation period approaches the inventory of 1 kg if the isotope is stable, and 10% of the initial inventory if the radionuclide decays according to the specified half-life.

The middle row of Figure 51 shows the behavior for a cylindrical waste form degradation model. As expected, the waste form is fully degraded after $t(q_{WF} = 0) = R \cdot \rho_{WF}/r + t_f = 756{,}000$ years.

The bottom row of Figure 51 shows the behavior for a disk-shape waste form with a thickness of `CYLLEN` = 0.2 m. Since the thickness is less than the diameter of the cylinder, the waste is consumed by axial degradation from the two circular end faces at $t(q_{WF} = 0) = L/2 \cdot \rho_{WF}/r + t_f = 575{,}000$ years, at which time the rate drops sharply to zero. As expected, the cumulative mass of the released component reaches 1 kg for the stable isotope, and 0.1 kg for the decaying radionuclide.

This examples verify the implementation of radionuclide releases from a degrading waste form, assuming that specific degradation rates are constant, i.e., not affected by temperature.

Table 16. Parameters for Waste Degradation

Parameter	Waste Degradation Model	
	Fractional	Cylindrical
ROCKS*		
DROK	-	waste form density, ρ_{WF} [kg m^{-3}]
GENER		
SL	'FDR'	'CYL'
NS	material number of waste form	material number of waste form
GX	fractional degradation rate, ω [s^{-1}]	degradation rate, r [kg m^{-2} s^{-1}]
EX $	inventory, $m_0(1 - IRF)$ [kg m^{-3}]	inventory, $m_0(1 - IRF)$ [kg m^{-3}]
HG	canister failure time, t_f [s]	canister failure time, t_f [s]
GMOD #	-	initial cylinder radius, R [m]
CYLLEN @	-	cylinder length, L [m]
TRACE &		
TRHENRY	rate multiplication factor, Υ [-]	rate multiplication factor, Υ [-]
TRARR(1) !	pre-exponential factor q_0 [s^{-1}]	pre-exponential factor q_0 [kg m^{-2} s^{-1}]
TRARR(2) ^	activation energy, E_a [J mol^{-1}]	activation energy, E_a [J mol^{-1}]
TRMW %	reference temperature, T_{0c} [°C]	reference temperature, T_{0c} [°C]

* Define material name and properties for waste form; only *DROK* affects degradation model.

\# Provide in block GENER.1, Columns 71–80.

@ Provide in block GENER.1, Columns 81–90; if $L = 0$, degradation from the circular end faces of the cylinder is ignored (see Eq. 19-8).

$ The inventory needs to be provided as the mass of radionuclide per volume of waste form as represented in the model. For example, if canisters have length L and radius R, then $m_0 = M_{can}^{RN}/L \cdot A$, where $M_{can}^{RN} = WLF \cdot (L_{can} \cdot \pi \cdot R^2)$ with the waste loading factor defined as $WLF = X_{WF}^{RN} \cdot \rho_{WF}$, and A is the sum of the cross-sectional areas perpendicular to the canister axis of all elements representing the waste form. For example, if canisters are aligned in X direction and have a cross-sectional area of $\pi \cdot R^2 \leq \mathrm{A_{grid}}$, then $A = A_{grid}$ (for a Cartesian grid, $A = A_{grid} = \Delta y \cdot \Delta z$). Note that if the resolution in the YZ plane (and therefore A_{grid}) changes along the canister axis, a new material and new GENER block with different inventory density m_0 must be specified. If the waste form itself is discretized, i.e., $\pi \cdot R^2 > \mathrm{A_{grid}}$, then $A = \pi \cdot R^2$, regardless of the discretization A_{grid}.

& See *Finsterle* [2021]. Provide Arrhenius parameters of waste form degradation in last, additional row of block TRACE for a (non-existent) tracer component *NK*. (Note that rows 1 to *NK* - 1 contain the Arrhenius parameters for the radionuclides.)

! If zero, the rate specified in variable *G* of the GENER block is used

^ A non-zero value is needed to trigger the use of the Arrhenius equation for waste degradation.

% If blank or zero, T_{0c} is set to infinity, and q_0 represents the intrinsic rate; $T_0 = T_{0c} + 273.15$.

```
EOS1nT waste form degradation test cases
MULTI----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8----*----9
    3    4    2    6
ROCKS ----1----*----2---*----3----*----4----*----5----*----6----*----7----*----8----*----9
WAFO1     1    3000.0      0.01 1.000E-30 1.000E-30 1.000E-30      40.0   1.00E50
   1.0E-08   1.0E-05                                    0.001
WAFO2     1    3000.0      0.01 1.000E-30 1.000E-30 1.000E-30      40.0   1.00E50
   1.0E-08   1.0E-05                                    0.001
WAFO3     1    3000.0      0.01 1.000E-30 1.000E-30 1.000E-30      40.0   1.00E50
   1.0E-08   1.0E-05                                    0.001
WAFO4     1    3000.0      0.01 1.000E-30 1.000E-30 1.000E-30      40.0   1.00E50
   1.0E-08   1.0E-05                                    0.001
WAFO5     1    3000.0      0.01 1.000E-30 1.000E-30 1.000E-30      40.0   1.00E50
   1.0E-08   1.0E-05                                    0.001
WAFO6     1    3000.0      0.01 1.000E-30 1.000E-30 1.000E-30      40.0   1.00E50
   1.0E-08   1.0E-05                                    0.001

TRACE---IH----*---KD----*----L----*-ARR1----*-ARR2----*---MW----*--DID----*----8----*----9
    1.0E10     0.000 1.000E-20       0.0       0.0    0.0180        -1    stable
    1.0E10     0.000 7.296E-14       0.0       0.0    0.0360        -1     Cl-36
Delete this line to include temperature dependence using the Arrhenius equation
       1.0                                  60000.0      40.0        -1     GLASS

PARAM----1----*-123456789012345678901234----*----5----*----6----*----7----*----8----*----9
   19999    99991000009000000000300005000
                    1.0E+05
     1.E-6                                                1.E-8
             1.0E+07                40.0                   0.0

MOMOP----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8----*----9
2 1          3           4             8

ELEME----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8----*----9
F    1          WAFO1       0.3
C    1          WAFO2       0.3
D    1          WAFO3       0.3
F    2          WAFO4       0.3
C    2          WAFO5       0.3
D    2          WAFO6       0.3

CONNE----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8----*----9

START----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8----*----9

(1) G:      For FDR: Fractional waste-form degradation rate [1/s]
            For CYL: Surface dissolution rate [kg/m2/s]
(2) EG:     Radionuclide inventory [kg/m^3];
            divide linear inventory mass [kg/m] by cross-section perpendicular to
            canister axis
(3) HG:     Time of canister failure, start of waste degradation
(4) GMOD:   Initial cylinder radius (for CYL only)
(5) CYLLEN: Cylinder length

GENER-T--1----*----2----*----3----*----4      Rate Inventory  Failtime    Radius     Length
F    1FDR 1                        COM2  3.6482-13 3333.3E-3 3.1557E12
C    1CYL 2                        COM2  2.000E-11 3333.3E-3 3.1557E12     0.138
D    1CYL 3                        COM2  2.000E-11 3333.3E-3 3.1557E12     0.138       0.2
F    2FDR 4                        COM3  3.6482-13 3333.3E-3 3.1557E12
C    2CYL 5                        COM3  2.000E-11 3333.3E-3 3.1557E12     0.138
D    2CYL 6                        COM3  2.000E-11 3333.3E-3 3.1557E12     0.138       0.2

ENDCY----1----*----2----*----3----*----4----*----5----*----6----*----7----*----8----*----9
```

Figure 50. TOUGH2 input file for simulating radionuclide release by degrading fractional, cylindrical, and disk-shaped waste forms using constant and time-dependent degradation rates for stable and decaying isotopes.

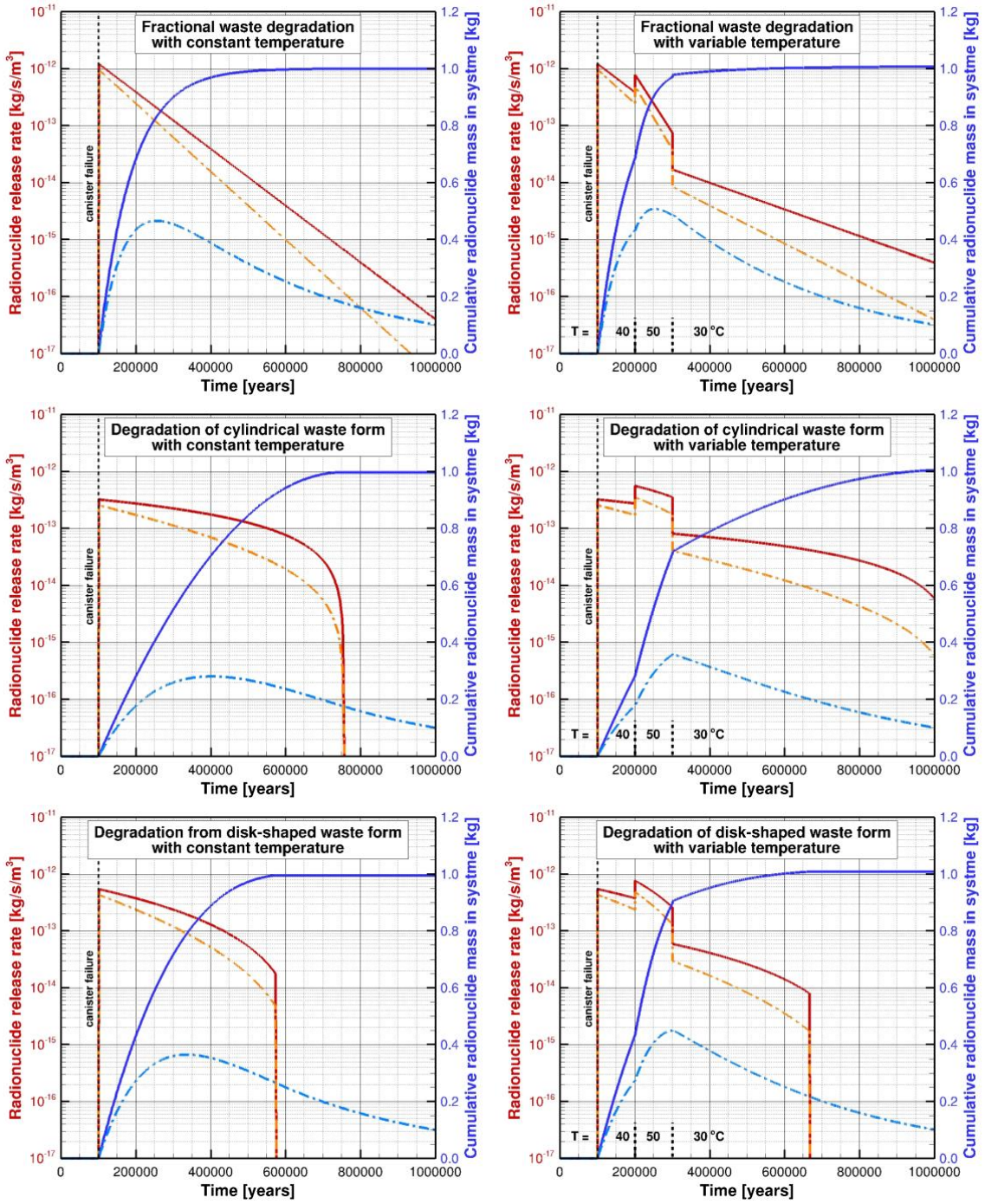

Figure 51. Isotope release rates (warm colors) and mass of released isotopes in the system (cold colors) for a stable isotope (solid lines) and a decaying radionuclide (dash-dotted lines) for a constant degradation rate (left column) and a temperature-dependent degradation rate (right column) using a fractional degradation model top row), a cylindrical waste form (middle row), and a disk-shaped waste form (bottom row).

ACKNOWLEDGMENT

This work was supported, in part, by the Laboratory of Hydraulics, Hydrology, and Glaciology (VAW) of the Swiss Federal Institute of Technology (ETH), Zürich, Switzerland, by the U.S. Dept. of Energy under Contract No. DE-AC02-05CH11231, by the Swiss National Cooperative for the Disposal of Radioactive Waste (Nagra), Wettingen, Switzerland, by Warm Arctic ehf, Kópavogur, Ísland, by Vatnaskil, Reykjavík, Ísland, by Deep Isolation, Berkeley, California, and by Geologica, San Francisco, California.

REFERENCES

Archie, G.E., The electrical resistivity log as an aid in determining some reservoir characteristics, *Trans. AIME, 146*, 54–62, 1942.

Carslaw, H.S. and J.C. Jaeger, *Conduction of Heat in Solids, 2nd edition*. Oxford University Press, Oxford, pp. 336, 1959.

Civan, F., Effective correlation of apparent gas permeability in tight porous media, *Transp. Porous Media*, 82, 375–384, 2010.

Doughty, C., *User's Guide for Hysteretic Capillary Pressure and Relative Permeability Functions in TOUGH2*, Report LBNL-6533E, Lawrence Berkeley National Laboratory, Berkeley, Calif., March 2013.

Deutsch, C.V., and A.G. Journel, *GSLIB, Geostatistical Software Library and User's Guide*, Oxford University Press, New York, New York, 1992.

Ergun, S., Fluid flow through packed column, *Chemical Engineering Progress, 48*(2), 89–94, 1952.

Falta, R. W., K. Pruess, S. Finsterle, and A. Battistelli, *T2VOC User's Guide*, Report LBL-36400, Lawrence Berkeley Laboratory, Berkeley, Calif., 1995.

Finsterle, S., G. J. Moridis, and K. Pruess, *A TOUGH2 Equation-of-State Module for the Simulation of Two-Phase Flow of Air, Water, and a Miscible Gelling Liquid*, Report LBL-36086, Lawrence Berkeley Laboratory, Berkeley, Calif., May 1994.

Finsterle, S., *Parallelization of iTOUGH2 Using PVM*, Report LBNL-42261, Lawrence Berkeley National Laboratory, Berkeley, Calif., 1998.

Finsterle, S., and P.A. Witherspoon, *Implementation of the Forchheimer Equation into iTOUGH2*, Project report, Witherspoon Inc., Berkeley, Calif., 2001.

Finsterle, S., and M.B. Kowalsky, *iTOUGH2-GSLIB User's Guide*, Report LBNL/PUB-3191, Lawrence Berkeley National Laboratory, Berkeley, Calif., 2007.

Finsterle, S., *iTOUGH2-IFC: An Integrated Flow Code in Support of Nagra's Probabilistic Safety Assessment—User's Guide and Model Description*, Report LBNL-1441E, Rev.3, Lawrence Berkeley National Laboratory Report, Berkeley, Calif., 2009.

Finsterle, S., *iTOUGH2 Universal Optimization Using the PEST Protocol*, Report LBNL-3698E, Lawrence Berkeley National Laboratory, Berkeley, Calif., July 2010.

Finsterle, S., *iTOUGH2 User's Guide,* Report LBNL-40401, Lawrence Berkeley National Laboratory, Berkeley, Calif., 2015a.

Finsterle, S., *iTOUGH2 Command Reference*, Report LBNL-40401 (Updated reprint), Lawrence Berkeley National Laboratory, Berkeley, Calif., 2015b.

Finsterle, S., *iTOUGH2 Sample Problems*, Report LBNL-40402 (Updated reprint), Lawrence Berkeley National Laboratory, Berkeley, Calif., 2015c.

Finsterle, S., *iTOUGH2-EOS1nT: A Nonisothermal Two-Phase Flow Simulator for Water and Multiple Tracers—User's Guide*, Report FGC-17-01, Finsterle GeoConsulting, Kensington, Calif., January 2021 (revised).

Florence, F.A., J.A. Rushing, K.E. Newsham, and T.A. Blasingame, Improved permeability prediction relations for low permeability sands. SPE 107954, presented at the 2007 SPE Rocky Mountain Oil and Gas Technology Symposium, Denver, Apr 16–18, 2007.

Forchheimer, P., Wasserbewegung durch Boden, *Zeitschrift Verein Deutscher Ingenieure, 45*, 1782–1788, 1901.

Frederick, D.C., and R. M. Graves, New correlation to predict non-Darcy flow coefficients at immobile and mobile water saturation, Paper SPE 28451, presented at 1994 SPE Annual Technical Conference and Exhibition, pp. 915–921, New Orleans, LA, Sept. 25–28, 1994.

Geertsma, J., Estimating the coefficient of inertial resistance in fluid flow through porous media, *SPEJ*, 445–450, 1974.

Guðmundsdóttir, H., M.Þ. Jónsson, J.-C. Berthet, A. Arnaldsson, and S. Finsterle, *iTOUGH2-FloWell: Simulating Coupled Reservoir and Wellhead Conditions, User's Guide*, Report FGC-17-08 and LBNL-7017E, 2018.

Heid, J.G., J.J. McMahon, R.F. Nielsen, and S.T. Yuster, *Study of the Permeability of Rocks to Homogenous Fluids*, API Drilling & Production Practice, New York, pp. 230–246, 1950.

Hassanizadeh, S.M., M.A. Celia, and H.K. Dahle, Dynamic effect in the capillary pressure-saturation relationship and its impacts on unsaturated flow, *Vadose Zone J*., *1*, 38–57, 2002.

Horseman, S., J. Higgo, J. Alexander, and J. Harrington, *Water, Gas and Solute Movement through Argillaceous Media*, Nuclear Energy Agency Rep. CC-96/1. OECD, Paris, 1996.

Janicek, J. D., and D. L. Katz, Applications in unsteady state gas flow calculations, *Proc.*, Univ. of Michigan Research Conference, June 20, 1955.

Jones, F.O. and W.W. Owens, A laboratory study of low permeability gas sands, SPE 7551, presented at the 1979 SPE Symposium on Low-Permeability Gas Reservoirs, Denver, May 20–22, 1979.

Jones, S. C., Using the inertial coefficient, □, to characterize heterogeneity in reservoir rock, Paper SPE 16949, presented at the 1987 SPE Annual Technical Conference and Exhibition, New Orleans, LA, Oct. 5–8, 1987.

Klinkenberg, L. J., The permeability of porous media to liquids and gases, in *Drilling Production Practice*, pp. 200–213, Am. Pet. Inst., Washington, D.C., 1941.

Letham, E.A., and R.M. Bustin, Klinkenberg gas slippage measurements as a means for shale pore structure characterization, *Geofluids*, 16, 264–278, 2016.

Liu, H.-H., C. Doughty, and G. S. Bodvarsson, An active fracture model for unsaturated flow and transport in fractured rocks, *Water Resour. Res.*, *34*(10), 2633–2646, 1998.

Liu, X., F. Civan, and R. D. Evans, Correlation of the non-Darcy flow coefficient, *JCPT*, *34*(10), 50–54, 1995.

Luckner, L., M. Th. van Genuchten, and D. Nielsen, A consistent set of parametric models for the two-phase flow of immiscible fluids in the subsurface, *Water Resour. Res.*, *25*(10), 2187–2193, 1989.

Magnúsdóttir, L., and S. Finsterle, *iTOUGH2-EOS1sc: Multiphase Reservoir Simulator for Water Under Sub- and Supercritical Conditions, User's Guide*, Report LBNL-7017E, Lawrence Berkeley National Laboratory, Berkeley, Calif., 2015.

Millington, R.J. and J.P. Quirk, Permeability of Porous Solids, *Trans. Faraday Soc.*, 57, 1200–1207, 1961.

Magnúsdóttir, L., and S. Finsterle, *iTOUGH2-EOS1sc: Multiphase Reservoir Simulator for Water under Sub- and Supercritical Conditions*, *User's Guide*, Report LBNL-7017E, Lawrence Berkeley National Laboratory, Berkeley, Calif., 2015.

Moridis, G.J., and K. Pruess, *T2SOLV: An Enhanced Package of Solvers for the TOUGH2 Family of Reservoir Simulation Codes*, Report LBNL-40933, Lawrence Berkeley National Laboratory, Berkeley, Calif., 1997.

Nagra, *Effects of Post-Disposal Gas Generation in a Repository for Spent Fuel, Vitrified High-Level Waste and Long-Lived Intermediate-Level Waste Sited in Opalinus Clay*, Nagra Technical Report NTB 04-06, Nagra, Wettingen, Switzerland, July 2004.

Nuth, M., and L. Laloui, Effective stress concept in unsaturated soils: Clarification and validation of a unified framework, *Int. J. Numer. Anal. Meth. Geomech.*, *32*, 711–801, 2008.

Oldenburg, C.M., G.J. Moridis, N. Spycher, and K. Pruess, *EOS7C Version 1.0: TOUGH2 Module for Carbon Dioxide or Nitrogen in Natural Gas (Methane) Reservoirs*, Report LBNL-56589, Lawrence Berkeley National Laboratory, Berkeley, Calif., 2004.

Parashar, R., and D. Reeves, Computation of flow and transport in fracture networks on a continuum grid, MODFLOW and MORE 2011: Integrated Hydrologic Modeling: Golden, Colorado, 2011.

Parker, J. C., R. J. Lenhard and T. Kuppusamy, A parametric model for constitutive properties governing multiphase flow in porous media, *Water Resour. Res.*, *23*(4), 618–624, 1987.

Pereira, J.-M., O. Coussy, E.E. Alonso, J. Vaunat, and S. Olivella, Is the degree of saturation a good candidate for Bishop's X parameter?, In: *Proc. Fifths Int. Conf. on Unsaturated Soils*, Barcelona, Spain, 913–919, Sep. 2010.

Pritchett, J.W., *STAR User's Manual, Version 9.0*, Technical Report SAIC-02/1055, Science Applications International Corporation, San Diego, Calif., September 2002.

Pruess, K., *TOUGH User's Guide*, Report NUREG/CR-4645, Nuclear Regulatory Commission (also Report LBL-20700, Lawrence Berkeley Laboratory, Berkeley, Calif.), 1987.

Pruess, K., *TOUGH2—A General-Purpose Numerical Simulator for Multiphase Fluid and Heat Flow*, Report LBL-29400, Lawrence Berkeley Laboratory, Berkeley, Calif., 1991.

Pruess, K., C. Oldenburg, and G. Moridis, *TOUGH2 User's Guide, Version 2.0*, Report LBNL-43134, Lawrence Berkeley Laboratory, Berkeley, Calif., 1999.

Pruess, K. and A. Battistelli, *TMVOC, A Numerical Simulator for Three-Phase Non-isothermal Flows of Multicomponent Hydrocarbon Mixtures in Saturated-Unsaturated Heterogeneous Media*, Report LBNL-49375, Lawrence Berkeley National Laboratory, Berkeley, Calif., 2002.

Pruess, K., *ECO2N: A TOUGH2 Fluid Property Module for Mixtures of Water, NaCl, and CO_2*, Report LBNL-57952, Lawrence Berkeley National Laboratory, Berkeley, Calif., 2005.

Pruess, K., *ECO2M: A TOUGH2 Fluid Property Module for Mixtures of Water, NaCl, and CO_2, Including Super- and Sub-Critical Conditions, and Phase Change Between Liquid and Gaseous CO_2*, Report LBNL-4590E, Lawrence Berkeley National Laboratory, Berkeley, Calif., 2011.

Pruess, K., C. Oldenburg, and G. Moridis, *TOUGH2 User's Guide, Version 2.1*, Report LBNL-43134, Lawrence Berkeley Laboratory, Berkeley, Calif., 2012.

Rautman, C.A., *Preliminary Geostatistical Modeling of Thermal Conductivity for a Cross Section of Yucca Mountain, Nevada*, Report SAND94-2283, UC-814, Sandia National Laboratories, Albuquerque, New Mexico, 1995.

Rutqvist, J., Y. Wu, Y. Tsang, and G. Bodvarsson, A modeling approach for analysis of coupled multiphase fluid flow, heat transfer, and deformation in fractured porous rock, *Int. J. Rock Mech. Min. Sci.*, *39*, 429–442, 2002.

Sampath, C.W and K. Keighin, Factors affecting gas slippage in tight sandstones, SPE 9872, *J. Petrol. Technol.*, 34, 2715–2720, 1982.

Stauffer, P.H., K.C. Lewis, J.S. Stein, B.J. Travis, P. Lichtner, and G. Zyvoloski, Joule-Thomson effects on the flow of liquid water, *Transp. Porous Media, 105*(3), 471–485, 2014.

Terzaghi, K., The shearing resistance of saturated soils and the angle between the planes of shear, In: *Proc. 1st Int. Conf. Soil Mech. Found. Engng*, Volume 1, Cambridge, MA, pp. 54–56, 1936.

Thauvin, F., and K.K. Mohanty, Network modeling of non-Darcy flow through porous media, *Transport in Porous Media*, *31*, 19–37, 1998.

Van Genuchten, M.Th., A closed-form equation for predicting the hydraulic conductivity of unsaturated soils, *Soil Sci. Soc. Am. J.*, *44*(5), 892–898, 1980.

Van Loon, L.R., *Effective Diffusion Coefficients and Porosity Values for Argillaceous Rocks and Bentonite: Measured and Estimated Values for the Provisional Safety Analyses for SGT-E2*, Nagra Technical Report NTB 12-03, Nagra, Wettingen, Switzerland, November 2014.

Van Loon, L.R., and J. Mibus, A modified version of Archie's law to estimate effective diffusion coefficients of radionuclides in argillaceous rocks and its application in safety analysis studies, *Appl. Geochem., 59*, 85–94, 2015.

Wainwright, H.M., and S. Finsterle, *Global Sensitivity and Data-Worth Analysis in iTOUGH2, User's Guide*, Report LBNL-1005838, Lawrence Berkeley National Laboratory, Berkeley, Calif., 2015.

Walsh, R., N. Calder, and J. Avis, A simple implementation of 1D hydromechanical coupling in TOUGH2, In: *Proceedings of the TOUGH Symposium 2012.* Report LBNL-5808E. Lawrence Berkeley National Laboratory, Berkeley, Calif., 425–432, 2012.

Weill S., E. Mouche, and J. Patin, A generalized Richards equation for surface/subsurface flow modeling, *J. Hydrol., 366*, 9–20, 2009.

Wu, Y.-S., S. Finsterle, and K. Pruess, *EOS3nn: An iTOUGH2 Module for Non-Newtonian Liquid and Gas Flow*, Report LBNL-51103, Lawrence Berkeley National Laboratory, Berkeley, Calif., 2002.

Xie, M., Wang, W., de Jonge, J. & Kolditz, O. (2007): Numerical modelling of swelling pressure in unsaturated expansive elasto-plastic porous media. *Transp. Porous Media*, 66, 311–339.

Zhang, Y., L. Pan, K. Pruess, and S. Finsterle, A time-convolution approach for modeling heat exchange between a wellbore and surrounding formation, *Geothermics*, *40*(4), 251–266, doi:10.1016/j.geothermics.2011.08.003, 2011.