TOUGH2-GRS
Version 1

User Manual

GRS - 403

Gesellschaft für Anlagen-
und Reaktorsicherheit
(GRS) gGmbH

TOUGH2-GRS
Version 1

User Manual

Martin Navarro
Jens Eckel

July 2016

Remark:

This study has been funded by the German Federal Ministry for the Environment, Nature Conservation, Building and Nuclear Safety (BMUB) under the support codes UM13A03400.

The work was conducted by Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH.

The authors are responsible for the content of the report.

GRS - 403
ISBN 978-3-944161-84-6

Key Words:
Final disposal, TOUGH2, TOUGH2-GRS, two-phase flow, radioactive waste

Kurzfassung

TOUGH2 ist ein Code zur Simulation mehrphasiger Strömungsvorgänge in porösen Medien, der vom Lawrence Berkeley National Laboratory, Kalifornien, USA, entwickelt wurde. Die GRS verwendet den Code seit 1991 im Rahmen von Prozess- und Langzeitsicherheitsanalysen für tiefe geologische Endlager und hat den Code im Laufe der Jahre um verschiedene endlagerrelevante Prozesse erweitert. Der von der GRS weiterentwickelte TOUGH2-Quellcode wird unter dem Namen TOUGH2-GRS geführt. Der vorliegende Bericht stellt Version 1.1.g des Codes vor, die im Vorhaben UM13 A 03400 des BMUB entwickelt wurde.

Abstract

TOUGH2 is a code for the simulation of multi-phase flow processes in porous media that has been developed by the Lawrence Berkeley National Laboratory, California, USA. Since 1991, GRS has been using the code for process analyses and safety assessments for deep geological repositories and has extended the code by several processes that are relevant for repository systems. The TOUGH2 source code that has been developed further by GRS is referred to as TOUGH2-GRS. The present report presents code version 1.1.g, which was developed in project UM13 A 03400 sponsored by the German Federal Ministry for the Environment, Nature Conservation, Building and Nuclear Safety (BMUB).

Table of Content

1 Introduction

TOUGH2 is a code for the simulation of multi-phase flow in porous media that has been developed at the Lawrence Berkeley National Laboratory, California, USA /PRU 99/. GRS uses the code to simulate multi-phase flow and radionuclide transport in repository systems for radioactive waste.

Since 1992, GRS has been supplementing the code by additional physical processes relevant in connection with repository systems. Early modifications focused on host rocks composed of rock salt. They included the implementation of porosity and permeability changes due to salt creep /JAV 95/, which is a main driver for barrier sealing and fluid flow in rock salt concepts. Non-linear and brine-dependent sorption has been implemented, too, in order to simulate the transport of radionuclides in the far-field of a repository /FIS 01/, /JAV 01/, /JAV 02/. Later on, code developments focused on argillaceous host rocks. From 2006 to 2008, a porosity model for immobile water was introduced /NAV 08/ as well as the dilation of microscopic pathways /NAV 09/. The new pathway dilation process has been applied in the FORGE project to gas injection experiments conducted in the Mont Terri underground research laboratory /NAV 13c/. In the FORGE project, new mechanisms of pathway dilation including kinetic dilation processes have been added to the code.

Until 2009, most TOUGH2 modifications of GRS had been realized independently. Integration and improvement of the previous modifications was attempted between 2009 and 2012 within BMUB project 3609 R 03210, resulting in the TOUGH2-GRS code version 00 /NAV 13b/, which was extensively used in the VSG project, a preliminary safety analysis for the Gorleben site (BMUB project UM10 A 03200, /KOC 12/, /LAR 13/).

The shift to version[1] 1.1.g, which is presented in this manual, was undertaken in project ZIESEL (UM13A03400). This code version has also been subject to extensive quality assurance, particularly by means of the automatic code testing programme SITA

---

[1] The version name takes the form `<major version>.<minor version>.<patch version>`. The major code version is related to specific code development projects. Source codes sharing the same minor version have to be compatible (same intended functions and same input and output parameters). Patch versions introduce bugfixes or minor code changes.

/HOT 16a/, /HOT 16b/, /SEH 16/. The result of these code tests are documented in /HOT 16a/, /HOT 16b/, /SEH 16/.

2 General code features

2.1 Code basis and bugfixes

TOUGH2-GRS is based on version 2.0 of TOUGH2, which has been supplemented by the solver package T2SOLV (files t2cg22.f and t2solv.f) of version 2.1.

The solver package T2SOLV has been improved by GRS to speed up the iterative solvers DSLUBC, DSLUCS, DLUSTB, and DSLUGM. Particularly, the modification improves the transformation from coordinate format (COO) to compressed sparse column format (CSC) by changing the routines DS2Y and QS2I1D. The modified code was presented at the TOUGH-Symposium 2015 /NAV 15/ and has shown to yield a significant speed improvement.

TOUGH2 version 2.0 has been issued with several bugfixes that have been published on the LBNL website. The following of these bugfixes have been included in TOUGH2-GRS (quotations from /LBNL 16b/):

- Bugfix of March 8, 2006 (file *module t2f.f*):

  "**Bug.** Occasionally, severe convergence problems are encountered when using the MINC method to simulate multiphase flow problems in fractured reservoirs. […] **Fix.** The problem occurs because of a conditional switching of relative permeabilities at interfaces in which a nodal distance is zero, as is the case for fracture nodal distances generated by the MINC method. It is recommended to simply comment out (or delete outright) the following two lines in subroutine MULTI, module t2f.f:

  `IF(D1.EQ.0..AND.REL20.NE.0.) REL1=REL2`

  `IF(D2.EQ.0..AND.REL10.NE.0.) REL2=REL1` […]"

- Bugfix of January 8, 2001 file (*module t2f.f*):

  "**Bug.** When using fully-coupled multiphase diffusion (default; MOP(24)=0) with the EWASG fluid property module, diffusive fluxes are erroneously printed as 0. This error affects only the printing of fluxes; the simulation uses proper fluxes, and the results calculated for mass fractions etc. are all o.k. **Fix.** Subroutine OUTDF in file t2f.f has two "PRINT 5071 ..." statements, the first of which is engaged for

MOP(24)=1, while the second is active for MOP(24)=0. In the second statement, change the argument of array `FDIF` from `II(KK,NPH,N)` to `II(KK,2,N)."`

Additionaly, solution 1.2.2 of issue "Too Many Significant Digits" that has been published on LBNL's website /LBNL 16a/ was implemented by replacing the calculation of variables `N10` and `DF` in subroutine `FLOP`, file t2cg22.f, by

```
N10=PRECISION(A)
DF=SQRT(EPSILON(A)/2)
```

2.2 Guidelines for code development

The TOUGH2-GRS code has been developed in compliance with general and specific GRS guidelines for code development (/GRS 13/, /HOT 16b/). Amongst others, these guidelines aim at code maintainability, code legibility, and avoidance of coding errors. Several measures were taken to reach these aims:

- New code parts were implemented in Fortran90/95 with extensive use of Fortran modules.
- The original source code was changed as little as possible. Changes are visible and reversible (original code parts remain in the code).
- New code parts use explicit variable declaration (`IMPLICIT NONE` statement) and avoid `COMMON` blocks, which can cause a storage access violation.
- Separate Fortran modules are used for every new category of physical processes (like e.g. seal corrosion) and every module is stored in a separate file. The rationale for doing so is to increase traceability and to facilitate code coverage analyses.
- In most cases, the accessed module variables are explicitly declared by using the `USE […], ONLY` statement in order to clarify module dependencies.
- Module variables are not changed by external routines in most cases. Therefore, variables, functions, and subroutines of a module are declared as `PRIVATE` by de-

fault[2]. Names of public variables, functions and subroutines incorporate the module name in order to clarify module dependencies.

- It has been attempted to keep variable names self-explanatory but not too long.
- The source code was intensively commented.
- It was checked that new code parts do not produce any compiler warnings (gfortran and Intel Fortran compilers have been used).

2.3 What is new in TOUGH2-GRS version 01?

Version 01 of TOUGH2-GRS has been subject to a major restructuring and now benefits from a highly modular structure. New physical processes (i.e. "process modules") have been added and the implementation of existing modules has partly been changed to a large extent. Module VTEMP (prescription of time-dependent temperatures), which was specific to the VSG project /KOC 12/, has been removed.

As compared to the previous version, all physical process modules (except for the PRLIM and the RN modules), are now included in the implicit Newton-Raphson solution scheme, which gives higher accuracy. Thus, changes of porosity and permeability caused by the COMP, CORRO, CORFL, FISS, and DEGRA modules are calculated within the Newton-Raphson iteration loop.

While some modules of the previous version 00 /NAV 13b/ had to be controlled by code modifications, all modules are now controlled by the standard input file using separate data blocks (i.e. the CORRO, COMP, or CORFL blocks).[3] Input data blocks of new modules may now contain blank lines, which are not interpreted as block end markers anymore but as input data (zeros or blanks). For new modules, the end of a data block is now recognized by the appearance of a new keyword so that the data blocks do not need to close with a blank line anymore.

---

[2] As an exception, all variables and routines of the MISC module are declared as `PUBLIC`.

[3] However, there are still two exceptions to this: the sizes of static TOUGH2 arrays (defined in module_sizes.f90) and the input parameters of the ANALYSE module. The ANALYSE module compiles runtime information that may help to understand convergence problems. This module is still under development and will therefore not be explained in this report.

Some input parameters are checked for consistency and plausibility in order to avoid errors in code application. These checks still do not cover the full parameter space.

A reporting system for messages, warnings and errors has been introduced to the code. A first reporting takes place immediately before the first time step. The code is forced to exit if an error has occurred at this time. A second report is given at the end of the simulation and is placed at the end of the main output so that the user can easily find the report.

2.4 Deviations from TOUGH2 v. 2.0 defaults

The user should be aware that the default behaviour of the code has changed in some aspects from TOUGH2 v 2.0 over TOUGH-GRS v00 to TOUGH-GRS v01. This implies that the same input data might not lead to the same simulation result.

2.4.1 Unlimited number of materials

It is now possible to adjust the maximum number of materials by means of the new parameter `MR`, which is located in file `module_sizes.f90`. Previous versions of TOUGH2-GRS and TOUGH2 version 2.0 use a maximum number of 27. Note that `MR` has to be *larger* than the number of materials in use because of the specific coding of the TOUGH2 input routine.

2.4.2 Unlimited number of time steps

Parameter `MCYC` determines the maximum number of time steps.

The default value `MCYC`=0 forces TOUGH2 v. 2.0 to quit the simulation before the first time step is executed. In contrast, both versions of TOUGH2-GRS use this setting to indicate the absence of a time step limit. For this reason, TOUGH2-GRS input files are not compatible with TOUGH2 v. 2.0 if `MCYC` = 0.

2.4.3 Automatic Leverett scaling disabled

In contrast to TOUGH2 v. 2.0 there is no automatic Leverett scaling of capillary pressures if permeability modifiers (`PM`) are used.

2.4.4 Extended printouts

TOUGH2-GRS v00 and v 01 generate additional printout at time zero and after the last time step. The default value of parameter `MCYPR` (i.e. `MCYPR`=0), which defines the maximum time step difference between two printouts, has been changed from 1 to 10^7.

Version 01 also generates the files ELE_MAIN and CON_MAIN, which hold element and connection specific printouts in the data format of the time series files FOFT, COFT, GOFT, and DOFT (see chapter 16).

In the printout of total component masses in active elements (printout starting with `"MASS IN PLACE"`) a new definition of active elements is now used. Elements with volumes larger than 1E+50 m^3 are now regarded as inactive elements.

2.4.5 Conservation of floating point precision

Input data are read by TOUGH2 v. 2.0 either from the STDIN channel or from separate files like the MESH file. If input data is passed via the STDIN channel, the data will first be written to a separate file and will be reread afterwards. This indirect reading of input data may cause a loss of floating point precision.

The problem is that read and write operations of TOUGH2 v. 2.0 are performed using the format string "E10.4". This causes a loss of precision if the user makes efficient use of the field length. Let us consider the read and write operations for the element volume, which is stored in the ELEME data block. Reading a numerical value of 1.23456E+4 from the STDIN channel, using the format string "E10.4" is not a problem, but writing this value to the MESH with the same format string yields 0.1235E+05 if the *gfortran* 4.9.3 compiler is used. Apparently, the introduction of leading zeros to the significand and to the exponent during the write operation has resulted in a loss of two digits. TOUGH2-GRS v. 01 solves this problem by reading and writing volumes as strings ("A10") and not as floating point values.

2.4.6 Enforced solution of linear equations

If TOUGH2 2.0 converges in the first iteration, primary variables remain unchanged because TOUGH2 exits the iteration before the linear equations are solved (by calling the `LINEQ` routine). A consequence of this is that sources and sinks, which are introduced by adding a term to the right side of the linear equation system, remain inactive. In general, this is not a major problem because the convergence criteria are met. However, the absence of sinks and sources can cause significant problems in connection with the CORFL, COMP and CORRO modules of TOUGH2-GRS. The CORFL module uses sinks and sources to promote the seal corrosion (see chapter 7). The COMP module, on the other hand, uses sinks and sources to change the porosity of compacting backfill (see chapter 5). Again, the CORRO module produces gas and withdraws water by means of sinks and sources (see chapter 6). These processes are stalled if convergence only needs one iteration, which may sum up to significant errors if this happens to take place over many time steps. In the long-term evolution of a repository system, there may be time periods with almost no fluid flow. In these periods, the processes of metal corrosion (CORRO), compaction (COMP), and seal corrosion (CORFL) may still take place and should advance. For this reason TOUGH2-GRS 01 forces the code to perform at least two iterations (`ITER` > 1).

2.5 Modules and physical processes

TOUGH2-GRS v. 01 has a highly modular structure. Most of the new code parts have been embedded in Fortran modules. Only few changes have been made to the TOUGH2 v. 2.0 core and to the solver package T2SOLV.

Tab. 2.1 gives a summary over the Fortran modules of TOUGH2-GRS v. 01. The names of module files comply with the scheme `module_`<module name>`.f90`. Some modules introduce a new input data block. The keyword of the data block corresponds to the module name (i.e. the input data block of module CORRO starts with the keyword `CORRO`; an exception to this rule is module XRANGE, which uses the keyword RANGE). Modules marked in the table with "7(R)" require the equation-of-state module EOS7 or EOS7R and must not be invoked in connection with other EOS modules.

Fortran modules are not allowed to use each other in a circular manner. In order to clarify module dependencies, all modules have been classified in nine levels (see Tab.

2.1), where modules were only allowed to call modules of a lower level. The compilation of modules has to proceed from the lowest to the highest level.

Tab. 2.1 Fortran modules of TOUGH2-GRS

Modules in bold face introduce new data blocks. Data block keywords correspond to the module names (except for module XRANGE which uses the keyword RANGE).

Level	Module name	EOS	QA	Purpose
9	PROGFLOW		B	High-level calls from within the original TOUGH2 code
8	TEST		B	Plausibility tests for input data
	OUTPUT		B/C	Extended output
	ANALYSE		C	Run-time monitoring for the physical interpretation of convergence failures
	NEWBLOCKS		B	Routines to introduce new input data blocks
7	PERM		B	General permeability routines and externalized permeability variables
6	**CNTRL**		C	General controls including time step control
	COMP	7(R)	A	Compaction of crushed salt backfill in rock salt cavities including changes of permeability and capillary pressure
	CORFL		A	Seal corrosion by corroding fluids
	CORRO		A	Gas production and water consumption due to metal corrosion
	DEGRA		A	Time-dependent change of seal permeability
	DOFT		B/C	Time series for domains
	FISS		D	Transport of gas in microscopic pathways including porosity and permeability changes with pressure thresholds, softening mechanisms and kinetic effects
	GCOMP		D	Properties of the main gas component
	PRLIM		D	Pressure limitation caused by the escape of gas from the model domain
	PTIME		C	TIME vector with higher precision
	RN		A	Decay, advection, diffusion and linear sorption for radionuclide chains
	EOSSALTREP		D	*For EOS-Modul SALTREP*
	XRANGE		C	Range control for primary variables
5	PORO		B	General routines for porosity change and externalized porosity variables
4	ALARM		B	Extended warning system
3	MISC		B	General constants and routines
2	TOUGH		B	Standard TOUGH2 variables (COMMON blocks) used in new modules (explicit type declaration)
1	SIZES		B	Static array sizes of TOUGH2

Some modules introduce groups of physical processes or properties. These are the modules

- COMP (compaction of crushed salt backfill in rock salt cavities),
- CORFL (seal corrosion by corroding fluids),
- CORRO (gas production and water consumption due to metal corrosion),
- DEGRA (time-dependent change of seal permeability),
- FISS (dilation of microscopic gas pathways),
- GCOMP (properties of the main gas component),
- PRLIM (pressure limitation caused by the escape of gas from the model domain), and
- RN (decay, advection, diffusion and linear sorption for radionuclide chains).

We will refer to these modules as "process modules". All process modules are level 6 modules (see Tab. 2.1).

Several process modules do not only change porosity but also require information on the current porosity estimation of the other process modules. For this reason, porosity change variables have been shifted from the process modules to a separate module PORO, which contains common porosity change routines and variables. As a level 5 module, it can be accessed by every process module.

Some process modules change intrinsic permeability, but there is no module that requires the permeability estimation of other process modules. For this reason, a general permeability change module PERM has been introduced at level 7 that collects the permeability changes of all process modules and passes it to the TOUGH2 core.

There are some other modules, which do not introduce any physical processes:

- Module PROGFLOW acts as a main linker for the TOUGH2 core (i.e. the original source code of TOUGH2 2.0).
- Module NEWBLOCKS holds general routines for reading new input data blocks.
- Module SIZES holds the static array sizes of the TOUGH2 core.

- Module TOUGH gives access to the common blocks of the TOUGH2 core.
- Module MISC holds general routines and variables used by most of the modules.
- The modules CNTRL and XRANGE control the simulation.
- Modules OUTPUT and DOFT control the extended printout.
- Module TEST checks the input data for plausibility in order to reduce the likelihood of erroneous simulation input.
- A common reporting system for messages, warnings, and errors has been introduced with version 01 of TOUGH2-GRS. The respective routines can be found in the module ALARM.
- Module ANALYSE provides a runtime monitoring for the interpretation of convergence failures. The module is still under development. We will therefore not address it any further in this report.
- Module EOSSALTREP is no genuine part of TOUGH2-GRS. It is only an interface to the EOS module SALTREP /NAV 14/, which is not described in this manual.

The modules listed in Tab. 2.1 show different degrees of quality assurance. Tab. 2.1 identifies four levels of quality assurance:

- level A: Verification tests have been designed and performed (see test documentation in /HOT 16a/)
- level B: Modules have been tested indirectly by the above tests
- level C: Modules have been frequently used by GRS but no systematic testing has been performed
- level D: Quality assurance pending

2.6 General remark on restart functionalities

Modules that introduce new input data blocks (modules in Tab. 2.1 marked in bold face) have a restart functionality. During the simulation, each of these modules creates a file named `SAVE_<block name>` holding restart data. In order to restore the status of a module, the respective files have to be renamed to `INCON_<block name>`. If a restart file is present, the corresponding input data passed to the code via the STDIN channel is ignored.

The restart functionality is not fully tested yet and we recommend **not** using it for the time being.

3 New flow and transport features

3.1 Check valve boundary (IRP=9 with upstream weighting mobilities)

Flow boundaries behaving like one-way outlets (check valve behaviour) can be introduced by means of inactive elements with zero permeability. However, this approach only works if intrinsic permeabilities are upstream-weighted. If harmonic weighting is used instead, a vanishing permeability would completely suppress advective flow.

We have therefore introduced a new permeability function (`IRP` = 9), which sets the relative permeability for both gas and liquid to zero. If mobilities are upstream-weighted (`MOP(11)` = 2), this yields the desired check valve behaviour even if intrinsic permeabilities are different from zero.

3.2 Simulated vertical pressure gradient (ICP=9 & IRP=1)

If the host rock of a repository is composed of impermeable rock salt, it usually does not have to be part of the modelling grid. Modelling grids for such repositories usually concentrate on the mining cavities, the backfill, and technical barriers.

For large-scale repository models, acceptable calculation times can often be achieved by using a coarse spatial discretization. One possibility to do so is to model horizontal drifts as one-dimensional structures without vertical discretization. Basically, this approach has to be used with caution since it may introduce the wrong physics. Without vertical discretization there will also be no vertical pressure gradient. As a consequence, a gas phase cannot rise to the top of the drift but will remain homogeneously distributed in the vertical. The hydraulic behaviour of such a system can be completely different from a system that allows a vertical separation of phases. Let us consider the case of a gas bubble that has been trapped in front of a seal with high gas entry pressure. If vertical phase separation takes place, the gas bubble would rise to the top of the drift, allowing the liquid phase to bypass it. Without vertical phase separation, the gas bubble might block the entire liquid flow.

If a phase rises to the top of a drift, it will spread in horizontal direction due to the gravitational forces that are acting on the gas and liquid phase. These forces are absent if vertical discretization is omitted. However, in this case they can be substituted by capillary forces.

TOUGH2-GRS introduces a new capillary pressure function (`ICP` = 9), which adds a correction term to the van Genuchten capillary pressure function. The purpose of the correction term is to simulate the gravitational forces which are missing if there is no vertical discretization.

In order to derive the correction term we consider the two flow systems shown in Fig. 3.1. Both systems are closed systems with constant initial gas pressure and no capillary forces. System A is composed of two grid elements with gas saturation S_{g1} and S_{g2}. System B is derived from system A by subdividing the elements in vertical direction. The mean gas saturation of domain 1 and 2 in system B shall be equal to the gas saturation of element 1 and 2 in system A, respectively.

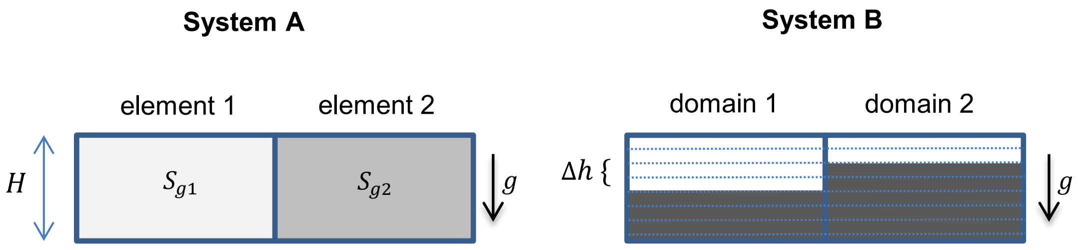

Fig. 3.1 Models for drift sections with (A) and without (B) vertical discretisation

g: gravity, H: drift height, Δh: height difference of water columns, S_g: gas saturation.

System B establishes a liquid flow from domain 2 to domain 1 due to a pressure difference

$$\Delta p_{2\rightarrow 1} = -\varrho g \Delta h \left(S_{g2} - S_{g1}\right), \tag{3.1}$$

resulting from the height difference of both water columns (note that S_{g1} and S_{g2} refer to a gas, not to a liquid saturation). Here, ϱ is the liquid's density, g the gravitational acceleration and Δh the height difference between the water columns. In contrast, system A will not establish a liquid flow because the pressure is equal at both element centres. In order to initiate the same flux in system A, a capillary function

$$p_{\text{grav}}(S_g) := -\varrho g H S_g \tag{3.2}$$

is introduced so that

$$\Delta p_{\text{grav},2\to 1} = -\varrho g \Delta h(S_{\text{g}2} - S_{\text{g}1})\,. \tag{3.3}$$

With a given capillary pressure function $p_{\text{cap}}(S_g)$, a corrected capillary pressure function $\hat{p}_{\text{cap}}$ is defined as (note that TOUGH2 uses $P_{\text{cap}} < 0$)

$$\hat{p}_{\text{cap}}(S_g) = p_{\text{cap}}(S_g) + p_{\text{grav}}(S_g). \tag{3.4}$$

Moreover, a linear relationship is derived for the relative permeability functions (`IRP` = 1) in order to capture the effect of the vertical flow cross section, see /NAV 16/.

In TOUGH2-GRS, $\hat{p}_{\text{cap}}$ is selected by means of `ICP` = 9 with $p_{\text{cap}}(S_g)$ being the van Genuchten function /GEN 80/, choosen by `ICP` = 7. The input parameters `CP(1)` to `CP(5)` are the same as for `ICP` = 7 and the additional parameter `CP(6)` holds the term $\varrho g H$ to simulate a vertical pressure gradient without vertical discretization.

4 Module interactions

4.1 Porosity

Porosity changes due to matrix compression and thermal expansion are basic parts of TOUGH2. TOUGH2-GRS allows porosity changes also due to backfill compaction (module COMP, see chapter 4) and the opening of micro fissures (module FISS, see chapter 11).

In each Newton-Raphson iteration TOUGH2-GRS calculates the components of porosity change. These are the porosity change $\Delta\phi_{\mathrm{ce}}$ from compaction and thermal expansion (from the `MULTI` routine), the porosity change $\Delta\phi_{\mathrm{comp}}$ from backfill compaction (module COMP), and the porosity change $\Delta\phi_{\mathrm{fiss}}$ from the opening of micro-fissures (module FISS). The sum

$$\Delta\phi = \Delta\phi_{\mathrm{ce}} + \Delta\phi_{\mathrm{comp}} + \Delta\phi_{\mathrm{fiss}} \tag{4.1}$$

is used to correct the estimation of porosity for the new time step.

All porosity changes mentioned above refer to the TOUGH2 porosity ϕ_{tough} (variable `PHI`), which is the porosity of the initial volume of the element V_{tough} (variable `EVOL`). However, porosity changes also change the volume of the porous medium. TOUGH2-GRS therefore introduces a true element volume $V_{\mathrm{cavity}}(t)$ and a corresponding physical porosity ϕ_{phys} (see chapter 5.1.1 for further explanation). Both porosity variables are connected by the assumption of pore space conservation

$$V_{\mathrm{cavity}}(t) \cdot \phi_{\mathrm{phys}}(t) = V_{\mathrm{tough}} \cdot \phi_{\mathrm{tough}} \tag{4.2}$$

During a Newton-Raphson iteration, $V_{\mathrm{cavity}}(t)$ and $\phi_{\mathrm{tough}}(t)$ are calculated from the values at the last time step t_{old} and from the porosity changes

$$V_{\mathrm{cavity}}(t) = V_{\mathrm{cavity}}(t_{\mathrm{old}}) + V_{\mathrm{tough}} \cdot \Delta\phi_{\mathrm{comp}} \tag{4.3}$$

$$\phi_{\mathrm{tough}}(t) = \phi_{\mathrm{tough}}(t_{\mathrm{old}}) + \Delta\phi_{\mathrm{ce}} + \Delta\phi_{\mathrm{comp}} + \Delta\phi_{\mathrm{fiss}}. \tag{4.4}$$

By definition, fissure porosity ϕ_{fiss} is only accessible to gas (see chapter 11). Capillary pressures therefore should be zero if the liquid-accessible porosity $\phi_{\mathrm{phys}} - \phi_{\mathrm{fiss}}$ is saturated. TOUGH2-GRS therefore provides a liquid saturation $\hat{S}_{\mathrm{liq}}$ for the liquid-accessible porosity, which is determined by the relation

$$S_{\mathrm{liq}} \cdot \phi_{\mathrm{phys}} \cdot V_{\mathrm{cavity}} = \hat{S}_{\mathrm{liq}} \cdot \left(\phi_{\mathrm{phys}} - \phi_{\mathrm{fiss}}\right) \cdot V_{\mathrm{cavity}} \tag{4.5}$$

which gives

$$\hat{S}_{\mathrm{liq}} = S_{\mathrm{liq}} \cdot \left(1 - \frac{\phi_{\mathrm{fiss}}}{\phi_{\mathrm{phys}}}\right)^{-1} . \tag{4.6}$$

In many cases, $\phi_{\mathrm{fiss}} \ll \phi_{\mathrm{phys}}$, so that $\hat{S}_{\mathrm{liq}} \approx S_{\mathrm{liq}}$.

4.2 Intrinsic permeability

TOUGH2-GRS allows the intrinsic permeability $\vec{k}$ of grid elements to change with time. $\vec{k}$ is either calculated on the basis of the anisotropic permeability $\vec{k}_{\mathrm{ROCKS}}$, which is defined in data block ROCKS, or on the basis of the isotropic porosity-permeability relationship $k_{\mathrm{COMP}}(\phi) \cdot \mathbf{I}$ that is defined by the COMP module (backfill compaction):

$$\vec{k} = \begin{cases} k_{\mathrm{COMP}}(\phi)\, \mathbf{I}\, |f_{\mathrm{PM}}|\, f_{\mathrm{DEGRA}}\, f_{\mathrm{CORFL}} & \text{if } k_{\mathrm{COMP}}(\phi) \text{ defined and } f_{\mathrm{PM}} > 0 \\ \vec{k}_{\mathrm{ROCKS}}\, |f_{\mathrm{PM}}|\, f_{\mathrm{DEGRA}}\, f_{\mathrm{CORFL}} & \text{otherwise.} \end{cases} \tag{4.7}$$

f_{DEGRA} and f_{CORFL} are permeability modifiers supplied by the modules DEGRA and CORFL, respectively. They equal 1 if no permeability change is intended. This is always the case if the respective module is inactive. f_{PM} is the standard permeability modifier `PM`, which is part of the grid data (data block `ELEME`). We use the absolute value of f_{PM} in equation (4.7) because the COMP module uses negative values of f_{PM} to turn off the porosity-permeability relationship $k_{\mathrm{COMP}}(\phi)$ for single elements.

Module FISS introduces a new permeability term to the flow equation for the gas phase (see equation (12.14) on p. 56). This means that intrinsic permeability $\vec{k}$ is not changed.

5 COMP: Convergence and compaction

Rock salt is subject to creep deformation. Mining cavities in rock salt therefore close with time. This process is often called "convergence" because the walls of a spherical cavity tend to converge against one point. The rate of convergence depends on the support of the backfill, if present, but also on the pore pressures inside the cavity. During the process of convergence, the backfill suffers compression as well as a decrease in porosity and permeability. Capillary pressure and relative permeabilities may change, too.

The module COMP implements the combined process of convergence and backfill compaction. The mathematical model will only be described briefly in this report. Further details on the origin and purpose of the model can be found in /NAV 13a/.

5.1 Physical models

5.1.1 Change of cavity volume and porosity

According to /STE 85/ the convergence of a cavity of volume $V(t)$ can be described by the differential equation

$$\dot{V}(t) = -K \cdot V(t) \tag{5.1}$$

Factor $K > 0$ is called the rate of convergence and depends on the local pressure, porosity, and temperature. Volume V is the volume of an entire cavity, not of grid elements that represent subsections of a cavity. However, with the assumption of a homogeneous distribution of K inside the cavity's backfill, equation (5.1) may be used to describe the compaction of subsection i of the cavity

$$\dot{V}_i = -K \cdot V_i \quad \text{with} \quad \sum_i V_i = V. \tag{5.2}$$

TOUGH2-GRS applies equation (5.2) to grid elements and since equation (5.2) describes the compaction of the backfill (not the convergence of the cavity) we will call factor K a *compaction rate* in the context of this equation.

TOUGH2-GRS does not change the volume of grid elements V_{tough}, which is stored in the variable `EVOL`. If grid elements are supposed to compact, V_{tough} only stands for the initial volume, not for the volume after compaction has taken place. The TOUGH2-internal porosity variable ϕ_{tough} (`PHI`) refers to the initial element volume V_{tough} with $V_{\mathrm{p}} = V_{\mathrm{tough}} \cdot \phi_{\mathrm{tough}}$ being the pore volume.

TOUGH2-GRS introduces an additional variable V_{cavity} (`VCAVITY`) which holds the current volume of the compacting grid element. This variable is updated in every time step. Further on, the new porosity function ϕ_{phys}(`PHIACT`) is introduced, which is calculated using an equation for the conservation of pore space, see equation (4.2).

V_{cavity} and ϕ_{phys} are the volume and porosity variables which are used by most process modules because they stand for the cavity volume and the material porosity, respectively. Since it is unknown which kind of material enters the initial grid volume during convergence, there is no other physical interpretation of V_{tough} and ϕ_{tough} except for that they represent the initial values. However, both $V_{\mathrm{tough}} \cdot \phi_{\mathrm{tough}}$ and $V_{\mathrm{cavity}} \cdot \phi_{\mathrm{phys}}$ may be used to determine the pore volume and thus the component masses.

5.1.2 Compaction rates

As mentioned before, the convergence rate is taken for the compaction rate of grid elements. TOUGH2-GRS uses a slightly modified approach of /NOS 05/ to calculate the convergence rate. In /NAV 13a/, this approach is called the "extended approach of Stelte".

According to this approach, the convergence rate is derived from a *reference convergence rate* K_{ref} by applying several factors which are introduced to isolate functional dependencies of the convergence rate on location, pressure, backfill porosity, time and temperature

$$K = \begin{cases} K_{\mathrm{ref}} \cdot f_{\mathrm{loc}} \cdot f_p \cdot f_\phi \cdot f_t \cdot f_T & \text{for} \quad \phi > \phi_{\mathrm{lim}} \\ 0 & \text{for} \quad \phi \le \phi_{\mathrm{lim}} \end{cases} \tag{5.3}$$

Variable ϕ_{lim} is a porosity limit for convergence, which is not part of the extended approach of Stelte. The single factors of the term belonging to $\phi > \phi_{\mathrm{lim}}$ are described in the following.

- **Reference convergence rate.** It has been assumed by Stelte /NAV 13a/ that the convergence rate of an open cavity converges against a constant value, the *stationary convergence rate*. The reference convergence rate K_{ref} is the stationary convergence rate of an open cavity at reference conditions $p_{\mathrm{ref}} = 1$ bar (reference gas pressure inside the cavity) and T_{ref} (temperature of the surrounding rock). The reference convergence rate can be estimated from in-situ measurements or from geomechanical simulations.

- **Local dependency.** f_{loc} captures the dependency of the stationary convergence rate on the location. Usually, the stationary convergence rate varies with depth, temperature and with the geometry of the cavity.

- **Pressure dependency.** The convergence rate of a cavity decreases with increasing pore pressure. Assuming isotropic stress p_G in the surrounding rock, the dependency on the mean pore pressure $\hat{p}$ is captured by a factor

  $$f_p = \left(1 - \frac{\hat{p} - p_{\mathrm{ref}}}{p_G - p_{\mathrm{ref}}}\right)^{m_p} \quad \text{for} \quad p \leq p_{\mathrm{G}}, \quad \text{otherwise } f_p = 0. \tag{5.4}$$

  p_{G} is the isotropic lithostatic pressure that would have been present at the reference location in the absence of a cavity. The mean pore pressure $\hat{p}$ is calculated as the arithmetic mean of gas pressure p_{gas} and liquid pressure p_{liq}, weighted by the liquid saturation S_{liq}

  $$\hat{p} = p_{\mathrm{gas}}\left(1 - S_{\mathrm{liq}}\right) + p_{\mathrm{liq}} S_{\mathrm{liq}} = p_{\mathrm{gas}} + p_{\mathrm{cap}} S_{\mathrm{liq}} \ . \tag{5.5}$$

  Note that the variation of p_G with depth is not covered by this approach.

- **Backfill support.** The supporting pressure on the cavity walls caused by the backfill increases with increasing compaction. The factor

  $$f_\phi(\phi, \phi_{\mathrm{r}}) = \left[1 + \frac{h(\phi, \phi_{\mathrm{r}})}{\left(\phi \cdot g(\phi, \phi_{\mathrm{r}})\right)^{1/m_\phi}}\right]^{-m_\phi} \quad \text{for} \quad \phi < \phi_{\mathrm{r}}, \text{ otherwise } f_\phi = 1, \tag{5.6}$$

  describes the effect of the supporting pressure on the convergence rate, where ϕ_{r} is the porosity prevailing at the onset of the supporting pressure. Functions $h(\phi, \phi_{\mathrm{r}})$ and $g(\phi, \phi_{\mathrm{r}})$ are determined by

$$h(\phi,\phi_r) = h_0 + h_1 \frac{\phi}{\phi_\mathrm{r}} + h_2 \left(\frac{\phi}{\phi_\mathrm{r}}\right)^2 + h_3 \left(\frac{\phi}{\phi_\mathrm{r}}\right)^3 \text{ and} \tag{5.7}$$

$$g(\phi,\phi_\mathrm{r}) = g_0 + g_1 \frac{\phi}{\phi_\mathrm{r}} + g_2 \left(\frac{\phi}{\phi_\mathrm{r}}\right)^2 \tag{5.8}$$

with $h_0 = 1$, $h_2 = -(3 + 2h_1)$, $h_3 = h_1 + 2$ and $g_0 = 1$. Parameters ϕ_0, h_1, g_1, and g_2 are input parameters. Presently, we set $m_\phi = m_p$.

- **Transient processes.** After excavation, the convergence rate of an open cavity converges against a stationary convergence rate, i.e. against the reference convergence rate K_ref. The gradual transition to the stationary convergence rate is expressed by the factor f_t

$$f_t = 1 + \frac{1}{1 + {\lambda_s}^{-1} \ln(V_0/V)} \left(\frac{K_0}{K_\mathrm{ref} \cdot f_\phi(\phi_0)} - 1\right). \tag{5.9}$$

  K_0 and ϕ_0 are the initial convergence rate and initial porosity, respectively, at the start of the simulation. Parameter λ_s mainly depends on the time and has to be determined by means of geomechanical calculations. More details can be found in /NOS 05/ and /NAV 13a/.

- **Temperature dependency.** Temperature has a strong influence on salt creep and therefore also on the convergence rate. The temperature dependency is captured by the factor f_T, which is defined as

$$f_T(T) = \frac{1}{1+a} \cdot \exp\left(\frac{Q_1}{R} \cdot \left(\frac{1}{T_\mathrm{ref}} - \frac{1}{T}\right)\right) + \frac{a}{1+a} \cdot \exp\left(\frac{Q_2}{R} \cdot \left(\frac{1}{T_\mathrm{ref}} - \frac{1}{T}\right)\right). \tag{5.10}$$

  This formulation considers two creep processes with activation energies of Q_1 and Q_2 and a weighting parameter a. T_ref is the reference temperature, i.e. the temperature of the rock at the time when the reference convergence rate was measured. T is the present rock temperature and R the universal gas constant. The equation is valid for homogeneous temperature distributions.

5.1.3 Porosity-permeability relationships

Experimental studies have noticed power law relationships between permeability and porosity of compacted salt grit for delimited porosity ranges /KRÖ 09/. The COMP module introduces piecewise-defined porosity-permeability relationships with up to five power subfunctions

$$k = a_i \, \phi^{b_i}. \tag{5.11}$$

Here, k is the intrinsic permeability and ϕ the porosity. Parameters a_i and b_i $(1 \leq i \leq 5)$ are fitting parameters.

The porosity domain of each subfunction is determined by the intersections of adjacent subfunctions i and $i + 1$. In the input data block COMP, subfunctions have to be defined in the order of descending porosity, i.e. subfunction i refers to higher porosities than subfunction $j > i$.

The user-defined piecewise function is overridden by

$$k = 0 \quad \text{for} \quad \phi \leq \phi_{k=0} \tag{5.12}$$

if input parameter $\phi_{k=0}$ is set to a positive non-zero value. Note that porosity-permeability relationships of the COMP module are only appropriate to define isotropic permeabilities.

5.1.4 Negative permeability modifiers

If a porosity-permeability relationship is specified by the COMP module for a certain material and if permeability modifiers (parameter `PMX` of data block ELEME) are non-negative, the corresponding permeability entries of data block ROCKS are ignored (see chapter 4.2 for an explanation). This means that positive permeability modifiers are applied to the porosity-permeability relationship of the COMP module, whereas negative values are applied to the anisotropic permeabilities of data block ROCKS.

5.1.5 Cross-section corrections

Rock convergence changes the volume of a mining cavity and its associated flow cross-section. The relation between the physical volume of a cavity and its cross-section depends on the geometry of the cavity. It is therefore assumed that the deformation of grid elements follows the deformation of a cylindrical drift with negligible compaction perpendicular to the drift axis.

The change of the flow cross-sections suggests a change in the interface area of element connections in the discretized model. In TOUGH2, the physical volume is an element-specific entity and therefore requires a weighting and averaging procedure if it is applied to element connections. Therefore, the interface areas remain unchanged but local permeabilities, diffusivities, and thermal conductivity are multiplied by a correction factor, thereby using the weighing and averaging scheme of these parameters. The correction factor is

$$\frac{A_t}{A_0} \tag{5.13}$$

with A_0 being the interface area of the connection and A_t the true interface area at time t. Note that A_t is unknown, but due to the above assumption we can substitute this term by

$$\frac{V_t}{V_0}\,, \tag{5.14}$$

where V_0 is the initial element volume and V_t is the true volume at time t. This term equals A_t/A_0 for cylindrical drifts with negligible compaction perpendicular to the drift axis.

5.1.6 Change of capillary pressures

Compaction changes the pore structure of the backfill and thus influences capillary pressures. The COMP module offers the possibility to alter the capillary pressures according to Leverett's J-function /LEV 41/

$$p_{\mathrm{c}}(S) = p_{\mathrm{c0}}(S)\left(\frac{k_0\phi}{\phi_0 k}\right)^{\alpha}, \tag{5.15}$$

where p_c is the capillary pressure at liquid saturation S, porosity ϕ, and permeability k. $p_{\mathrm{c}0}(S)$ is the unscaled capillary pressure function for the initial porosity ϕ_0 and initial permeability k_0 as defined in data block ROCKS. If the user chooses $p_{\mathrm{c}0}$ to be the van Genuchten function (`ICP` = 7), the maximum capillary pressure cut-off `CP(4)` is also applied to the scaled function.

5.2 Note on the compaction of saturated media

The compaction process may cause numerical problems if a fully liquid-saturated medium is compacted and if liquid flow is restricted. Pressures might then rise quickly and lead to a strong decrease of the time step width. The assumption of a compressible material (`COM` > 0) can significantly lengthen the time scale of the pressure rise with beneficial effects on automatic time stepping.

5.3 Input data

The input parameters for the module are passed via data block COMP (see Tab. 5.1). The compaction behaviour as well as the porosity-permeability relationship and the Leverett scaling of capillary pressures are defined for each material separately. Note that the restart file `SAVE_COMP` also stores porosity values which override the porosity values stored in the `INCON` file.

Tab. 5.1 Input format of data block COMP

Line	Column	Content	Format	Unit	Description	Variable
1	1-5	`"COMP"`	A5		Keyword	
2	1-5	Material name	A5		Name of compacting material (as defined in ROCKS) In the following, `M` is the material index.	
3	1-10	K_ref	G10.4	$[\mathrm{sec}^{-1}]$	Reference convergence rate	`COMP_KREF(M)`
	11-20	f_loc	G10.4	[1]	Local modification of the reference convergence rate	`COMP_FLOC(M)`
	21-30	p_G	G10.4	[Pa]	Lithostatic pressure at the location if no cavity would exist	`COMP_PG(M)`
	31-40	$m_p = m_\phi$	G10.4	[1]	Exponent for the calculation of f_p	`COMP_M(M)`
	41-50	ϕ_r	G10.4	[1]	Porosity at which backfill support commences	`COMP_PHIref(M)`
	51-60	h_1	G10.4	[1]	Parameter for the calculation of f_ϕ	`COMP_H1(M)`
	61-70	g_1	G10.4	[1]	Parameter for the calculation of f_ϕ	`COMP_G1(M)`

Line	Column	Content	Format	Unit	Description	Variable
	71-80	g_2	G10.4	[1]	Parameter for the calculation of f_ϕ	COMP_G2(M)
4	1-10	λ_s	G10.4	[1]	Parameter for the calculation of f_T	COMP_LAMBDAS(M)
	11-20	K_0	G10.4	[sec^{-1}]	Initial convergence rate	COMP_K0(M)
	21-30	a	G10.4	[1]	Weighting factor for creep processes	COMP_A(M)
	31-40	Q_1	G10.4	[J/mol]	Activation energy for creep process 1	COMP_Q1(M)
	41-50	Q_2	G10.4	[J/mol]	Activation energy for creep process 2	COMP_Q2(M)
	51-60	T_{ref}	G10.4	[°C]	Temperature of the rock at the reference location	COMP_TGREF(M)
	61-70	ϕ_{lim}	G10.4	[1]	Lower porosity limit for the convergence process (may be 0)	COMP_PHIMIN(M)
5	1-10	0 or 1	I10	[1]	Enable Leverett scaling of capillary pressures? (0 = no, 1 = yes)	COMP_LEVQ(M)
	11-20	k_0	G10.4	[m^2]	Reference permeability for Leverett scaling	COMP_LEVPERREF(M)
	21-30	ϕ_0	G10.4	[1]	Reference porosity for Leverett scaling	COMP_LEVPHIREF(M)
	31-40	α	G10.4	[1]	Exponent of the Leverett scaling	COMP_LEVEXP(M)
6	1-10	$\phi_{k=0}$	G10.4	[1]	Permeability will be 0 for porosities $\leq \phi_{k=0}$	COMP_CUTOFF(M)
7	1-10	n_{sec}	I10	[1]	Number of subsections of the porosity-permeability relation	COMP_NBRANCH(M)
8	1-10	a_i	G10.4	[m^2]	Factor of power law i	COMP_PERA(I,M)
	11-20	b_i	G10.4	[m^2]	Exponent of power law i	COMP_PERB(I,M)
Repeat line 8 for all n_{sec} sections						
Repeat line 2 to 7+n_{sec} for all compacting materials						

5.4 Amendments to version 00

Although the input parameters and input format have not changed since version 00, the implementation of the compaction process has been completely revised. Changes of porosity, permeability and capillary pressure are now calculated implicitly, i.e. inside the Newton-Raphson iteration loop, which gives higher accuracy.

6 CORRO: Gas production due to iron corrosion

In repositories for radioactive waste, the corrosion of iron is an important source for gas production. Iron corrosion requires the presence of water so that the availability of water can become a limiting factor for gas production. Within the CORRO module, iron corrosion is modelled by introducing gas sources and liquid sinks.

The user may define an arbitrary number of gas sources. Every gas source is specified by a gas generation rate (mole per second) and a maximum amount of gas that can be produced. Gas sources can be either placed inside a grid element or distributed over a material domain while assuming the same gas generation rate per element volume within this domain.

The CORRO module allows coupling the gas source to a water sink in order to reduce gas production if the availability of water decreases. The water consumption rate is coupled to the gas production rate by the input parameter $r_{\mathrm{w/g}}$ (`CORRO_moleWaterPerMoleGas`) which describes what amount of water is consumed for every mole of gas that is produced. The metals and chemical reactions involved in the corrosion process therefore do not have to be specified explicitly.

A gas source becomes inactive when the maximum amount of gas has been produced. The *degree of corrosion* indicates the fraction of the maximum amount that has already been produced. If the gas source is coupled to a water sink, gas production is halted if no water is available.

6.1 Water consumption

The COMP module uses three possible sources for water consumption. These are

1. the water contained in wastes and waste packages,
2. the water component of the liquid phase, and
3. the water that is implicitly contained in the brine component of the liquid phase.

The wastes and waste packages are no TOUGH2 entities, but their water content is explicitly specified as an input parameter m_{can} of the CORRO module. This water does not participate in the flow system. It is just amenable to the corrosion process and will be consumed first, that is before the consumption of pore water.

If the entire canister water has been consumed, the CORRO module starts to consume the water of the liquid phase. The water component will be consumed first, followed by the brine component. Usually, the consumption rate for the brine component is higher than for the water component since brine is not completely composed of water. The mass of water that is implicitly contained in brine is determined by the input parameter $r_{\mathrm{w/b}}$ (`CORRO_waterMassPerBrineMass`).

Water consumption is enabled by setting `CORRO_consumeWater=1` and is disabled by default. The canister water is always available for the corrosion process if water consumption has been enabled. In contrast, pore water consumption has to be enabled by setting `CORRO_consumePoreWater=1`. Pore water is declared as unavailable for the corrosion process if liquid saturation is lower than the saturation limit $S_{\mathrm{empty}}^{\mathrm{CORRO}}$ (`CORRO_Sliqempty`).

6.2 Gas component

The CORRO module only operates with the equation-of-state modules EOS7 and EOS7R, which introduce a single non-condensible gas component. It is this gas component that is generated by the CORRO module.

For reasons of simplicity, the CORRO module does not distinguish whether gas is produced inside or outside the waste canisters. The physical image behind this simplification is that the gas produced in the interior of a canister is able to escape from the canister immediately, e.g. due to pitting corrosion. The user should be aware that this assumption might be an inadequate model for some problems.

The main gas component of chemical iron corrosion is hydrogen but the non-condensable gas that is produced in connection with EOS7 and EOS7R is air. If hydrogen is the main gas component in the simulation, the properties of the gas component can be changed by means of the GCOMP module (see chapter 10). However, this may not be sufficient if the composition of the main gas component depends on time or location. Consequently, it is necessary to reflect on possible errors resulting from the choice of a gas component that does not match the gas component of the real system.

Firstly, an erroneous viscosity of the vapour-gas mixture will cause errors in the gas flow. This should be corrected by choosing an appropriate viscosity function using the GCOMP module. Secondly, an erroneous atomic weight leads to errors in gas density as well as in the mass and mass fraction of the gas component. Obviously, this has an impact on mass transport. However, an erroneous atomic weight does not necessarily lead to an erroneous gas transport in terms of the amount of substance if gas production is specified in terms of the amount of substance too:

- For a given amount of substance of a gas component, gas pressure is independent of the atomic weight because EOS7 and EO7R implement the ideal gas law.
- There is no major impact of the atomic weight on the density of the liquid phase if the mass fraction of dissolved gas is small (which usually is the case).
- According to Henry's law, the partial pressure of the main gas component is proportional to the mole fraction of the dissolved component. For this reason the amount of gas dissolved in liquid is independent of the gas component's atomic weight.
- The effect of the atomic weight on the diffusion of gas in the liquid phase is negligible. Let κ be the gas component of TOUGH2-GRS and $\hat{\kappa}$ the real gas component. If the dissolved amount of gas is equal for both gas components (i. e. $n^{\kappa} = n^{\hat{\kappa}}$) and if we set the diffusion coefficient of component κ equal to that of component $\hat{\kappa}$,

  $$d_{\mathrm{liq}}^{\kappa} \coloneqq d_{\mathrm{liq}}^{\hat{\kappa}}, \tag{6.1}$$

  then the diffusive flux of component κ in the liquid phase will be equal to that of the real component $\hat{\kappa}$ in terms of the amount of substance

  $$\frac{f_{\mathrm{liq}}^{\hat{\kappa}}}{\mu_{\hat{\kappa}}} = \frac{f_{\mathrm{liq}}^{\kappa}}{\mu_{\kappa}} \tag{6.2}$$

  with f_{β}^{κ} being the diffusive mass flow in phase β and μ_{κ} the atomic weight of component κ. Equation (6.2) is derived by substituting the diffusive mass flows with the corresponding TOUGH2 equation

  $$f_{\beta}^{\kappa} = -\phi \tau_0 \tau_{\beta} \rho_{\beta} d_{\beta}^{\kappa} \nabla X_{\beta}^{\kappa} \,. \tag{6.3}$$

  This gives

  $$m_{\mathrm{liq}} d_{\mathrm{liq}}^{\kappa} \nabla \frac{n^{\kappa}}{m_{\mathrm{liq}}} = \widehat{m}_{\mathrm{liq}} d_{\mathrm{liq}}^{\hat{\kappa}} \nabla \frac{n^{\hat{\kappa}}}{\widehat{m}_{\mathrm{liq}}}$$

which again reduces to equation (6.1) under the approximation that the dissolved gases have a minor influence on liquid density and that liquid density is constant in space.

- The same operations for the diffusion in the gas phase yield

  $$m_{\mathrm{gas}} d_{\mathrm{gas}}^{\kappa} \nabla \frac{n^{\kappa}}{m_{\mathrm{gas}}} = \widehat{m}_{\mathrm{gas}} d_{\mathrm{gas}}^{\widehat{\kappa}} \nabla \frac{n^{\widehat{\kappa}}}{\widehat{m}_{\mathrm{gas}}}. \tag{6.4}$$

  The gas masses m_{gas} and $\widehat{m}_{\mathrm{gas}}$ depend on the atomic weights so that equation (6.3) only reduces to

  $$d_{\mathrm{gas}}^{\kappa} = d_{\mathrm{gas}}^{\widehat{\kappa}} \tag{6.5}$$

  if either vapour or the main gas component is absent. Diffusion in the gas phase may therefore be affected by errors in atomic weight.

The user should therefore specify gas sources in terms of the amount of substance. The interpretation of the simulated gas transport should focus on the amount of substance too.

6.3 Gas generation rates

For each gas source the maximum amount of producible gas N_{tot} (`CORRO_moleGas`) is specified as well as the gas generation rate Q. Corrosion experiments usually consider the rate of gas generation as a function of time t. We therefore use time-dependent rate functions to specify gas generation rates.

Corrosion at full liquid saturation

The CORRO module offers an exponential and a power law relationship to describe the corrosion rates for full liquid saturation (i. e. the entire gas source is in contact with water). The exponential function

$$Q_{\text{e}}(t) = A_{\text{liq}} e^{Bt} \quad , B < 0 \tag{6.6}$$

with fitting parameters A_{liq} and B is selected by setting `usePowerLaw=0`. The power law relationship

$$Q_{\text{p}}(t) = a\left(\frac{t-t_0}{t_{\text{ref}}}\right)^n \quad , n < 0 \quad \text{and} \quad t - t_0 \neq 0 \tag{6.7}$$

is selected by setting `usePowerLaw` = 1. Parameters a, t_0, t_{ref}, and n are fitting parameters.

The amount of gas produced until time t is

$$N(t) = \int_{\hat{t}=0}^{t} Q(\hat{t}) d\hat{t}. \tag{6.8}$$

Using equations (6.6) and (6.7) for Q leads to

$$N(t) = \frac{A_{\text{liq}}}{B}(e^{Bt} - 1) \tag{6.9}$$

for the exponential function and

$$N(t) = \begin{cases} \frac{a\, t_{\mathrm{ref}}}{n+1}\left[\left(\frac{t - t_0}{t_{\mathrm{ref}}}\right)^{n+1} - \left(\frac{t_0}{t_{\mathrm{ref}}}\right)^{n+1}\right] & \text{for} \quad n \neq -1 \\ a\, t_{\mathrm{ref}} \ln\left(1 + \frac{t}{t_0}\right) & \text{for} \quad\ = -1 \end{cases} \tag{6.10}$$

for the power law function.

Equations (6.6) and (6.7) are not implemented directly because time-discretization of production rates would lead to a rough approximation of the cumulative amount of produced gas according to equation (6.8). Instead, a mean production rate $\bar{Q}$ during a time step Δt is calculated as

$$\bar{Q}_{t_{\mathrm{old}} \to t_{\mathrm{old}} + \Delta t} \coloneqq \frac{N(t_{\mathrm{old}} + \Delta t) - N(t_{\mathrm{old}})}{\Delta t}\ , \tag{6.11}$$

where time t_{old} is the starting time. Note that $\bar{Q}$ is the production rate under full liquid saturation, and the production rate under limited water availability $\tilde{Q}$ will have to be derived from $\bar{Q}$ later on.

In order to calculate $\bar{Q}$, the amount of produced gas for the last time step $N(t_{\mathrm{old}})$ is needed. TOUGH2-GRS does not store $N(t_{\mathrm{old}})$ but the normalized value of $N(t_{\mathrm{old}})$, which we call the *degree of corrosion*

$$g(t) \coloneqq \begin{cases} \frac{N(t)}{N_{\mathrm{tot}}} & \text{if less than 1} \\ 1 & \text{otherwise} \end{cases}. \tag{6.12}$$

Here, N_{tot} is the total amount of gas that can be produced by the source. Using the degree of corrosion, equation (6.11) is rewritten as

$$\bar{Q}_{t_{\mathrm{old}} \to t_{\mathrm{old}} + \Delta t} \coloneqq \frac{N(t_{\mathrm{old}} + \Delta t) - g(t_{\mathrm{old}})\, N_{\mathrm{tot}}}{\Delta t}. \tag{6.13}$$

Corrosion with limited water availability

Up to now, Q and N have been functions of time. In order to allow for a temporal stagnancy of corrosion, equation (6.13) is generalized by describing the time variable of the rate function Q as a function of the degree of corrosion

$$\theta : g \mapsto t \tag{6.14}$$

yielding

$$\bar{Q}_{t_{\mathrm{old}} \to t_{\mathrm{old}} + \Delta t} := \frac{N\big(\theta\big(g(t_{\mathrm{old}})\big) + \Delta t\big) - g(t_{\mathrm{old}})\, N_{\mathrm{tot}}}{\Delta t}. \tag{6.15}$$

The time function θ is derived by solving equation (6.12) using equation (6.9) or (6.10), which gives

$$\theta(g) \ = \frac{1}{B} ln \left(\frac{g N_{\mathrm{tot}} B}{A_{\mathrm{liq}}} + 1 \right) \tag{6.16}$$

for the exponential function and

$$\theta(g) = \begin{cases} \left[\dfrac{g N_{\mathrm{tot}} (n+1)}{t_{\mathrm{ref}}\, a} - \left(\dfrac{t_0}{t_{\mathrm{ref}}} \right)^{n+1} \right]^{\frac{1}{n+1}} t_{\mathrm{ref}} - t_0 & \text{for} \quad n \neq -1 \\ t_0 \left(\exp \dfrac{g N_{\mathrm{tot}}}{t_{\mathrm{ref}}\, a} - 1 \right) & \text{for} \quad\ = -1 \end{cases} \tag{6.17}$$

for the power law function. Note that equations (6.16) and (6.17) will only be used if `dependsOnTime=0`. Otherwise, $\theta(g) = t_{\mathrm{old}}$ will be used, which means that the corrosion rate decreases with time even if the corrosion process is stalled. The user should be aware that this is an unphysical behaviour in most cases.

The production rate for limited water availability $\tilde{Q}$ is taken for the production rate for unlimited water supply $\bar{Q}$ unless water consumption is enabled (`CORRO_consumeWater=1`) and corrosion rates depend on the degree of corrosion (`corro_dependsOnTime=0`). In what follows, $\tilde{Q} \neq \bar{Q}$ is considered.

If water consumption is enabled and corrosion rates depend on the degree of corrosion, the derivation of the production rate for limited water availability $\tilde{Q}$ from the production rate for unlimited water supply $\bar{Q}$ is controlled by the switch `CORRO_AssumeWaterTable`. Basically this switch decides which fraction of the gas source is in contact with water under unsaturated conditions. If the capillary fringe is small with regard to the height of a grid element, we can usually assume the formation of a horizontal water table. In this case, liquid saturation equals the fraction α of the element that is under full liquid saturation. As an approximation, α is used to define the fraction of the gas source that is under full liquid saturation. Choosing `CORRO_AssumeWaterTable=1` sets $\alpha = S_{\mathrm{liq}}$, otherwise $\alpha = 1$.

The exponential function is then defined as

$$\tilde{Q}_{t_{\mathrm{old}} \to t_{\mathrm{old}} + \Delta t} \coloneqq \frac{\alpha A_{\mathrm{liq}} e^{Bt} + (1-\alpha) A_{\mathrm{vap}}}{A_{\mathrm{liq}}} \, \bar{Q}_{t_{\mathrm{old}} \to t_{\mathrm{old}} + \Delta t} \tag{6.18}$$

and the power law function is defined as

$$\tilde{Q}_{t_{\mathrm{old}} \to t_{\mathrm{old}} + \Delta t} \coloneqq \alpha \, \bar{Q}_{t_{\mathrm{old}} \to t_{\mathrm{old}} + \Delta t}. \tag{6.19}$$

6.4 Input data

The input parameters for the corrosion process are defined in data block CORRO. The order of input parameters reflects the history of the CORRO module and will probably be revised in the future.

Tab. 6.1 Input format of data block CORRO

Line	Column	Content	Format	Unit	Description	Variable
1	1-5	`"CORRO"`	A5		Keyword	
2	1-10	$R_{\mathrm{w/b}}$	G10.4	–	Water mass per brine mass Default: 0.46	`CORRO_waterMassPerBrineMass`
2	11-20	$S_{\mathrm{empty}}^{\mathrm{CORRO}}$	G10.4	–	Pore water consumption only if $S_{\mathrm{liq}} > S_{\mathrm{empty}}^{\mathrm{CORRO}}$	`CORRO_Sliqempty`
3	1-5	`"E","e","M","m"`	A5		Location type. Gas source is either placed in an element ("E", "e") or homogeneously distributed over a material ("M", "m").	`CORRO_loctype(s)` `s` is the source index
3	6-10	Location	A5		Element or material name	`CORRO_loc(s)`
3	11-15	Switch: water consumption?	I5	1	Stop gas generation in absence of water? 0: no, 1: yes	`CORRO_consumeWater(s)`
3	16-20	Switch: pore water consumption?	I5	1	Stop gas generation in absence of pore water? 0: no, 1: yes	`CORRO_consumePoreWater(s)`
3	21-25	Switch: Assume water table?	I5	1	Recognize liquid saturation as relative height of a water table inside an element? 0: no, 1: yes	`CORRO_assumeWaterTable(s)`
3	26-30	Switch: Use time-dependent rate function?	I5	1	Calculate generation rates using the time instead of the degree of corrosion? 0: no, 1: yes	`CORRO_dependsOnTime(s)`
3	31-40	A_{liq}	G10.4	mol/s	Fitting parameter of exponential rate function for water contact	`CORRO_Aliquid(s)`
3	41-50	A_{vap}	G10.4	mol/s	Fitting parameter of exponential rate function for vapour contact	`CORRO_Avapour(s)`
3	51-60	B	G10.4	1/s	Fitting parameter of exponential rate function. 0: constant rate, <0: decreasing rate	`CORRO_B(s)`
3	61-70	–	–	–	INACTIVE	–
3	71-80	Switch: Use power function?	G10.4		Use power function instead of exponential function? 0: no, 1: yes	`CORRO_useFunction2(s)`
4	1-10	g_0	G10.4	1	Initial degree of corrosion	`CORRO_initialDegreeOfCorrosion(s)`
4	11-20	N_{tot}	G10.4	mol	Maximum amount of gas that can be produced	`CORRO_moleGas(s)`
4	21-30	m_{can}	G10.4	kg	Canister water	`CORRO_canisterWaterMass(s)`
4	31-40	$r_{\mathrm{w/g}}$	G10.4	1	Mole of water consumed for every mole of gas that is produced	`CORRO_moleWaterPerMoleGas(s)`
4	41-50	a	G10.4	mol/s	Fitting parameter of power law rate function	`CORRO_f2_a(s)`
4	51-60	n	G10.4	1	Fitting parameter of power law rate function	`CORRO_f2_n(s)`
4	61-70	t_0	G10.4	s	Fitting parameter of power law rate function	`CORRO_f2_t0(s)`
4	71-80	t_{ref}	G10.4	s	Fitting parameter of power law rate function	`CORRO_f2_tref(s)`
Repeat lines 3 and 4 for all gas sources						

6.5 Amendments to version 00

Module CORRO has been subject to major changes. Gas sources can now be placed either in a material or in a single element. It is also possible to superimpose several gas sources. The definition of gas sources has been simplified too. There is no need to specify the type of metal or stoichiometry of the chemical reaction anymore. The rate functions have been completely revised and the power law relationship has been added.

With version 01, the use of the CORRO module has been restricted to EOS7 and EOS7R. Gas production and water consumption are now calculated implicitly, which gives higher accuracy. The internal handling of gas sources has been designed for efficient loop parallelisation.

7 CORFL

The CORFL module models the permeability change of technical barriers (seals) due to the percolation of corrosive solutions. Although originally designed to simulate the corrosion of salt concrete by magnesium bearing solutions, the module does not specify the chemistry of the seal material and of the percolate.

7.1 Implementation

The CORFL module uses the primary variable *brine mass fraction* as an information carrier for the corrosion capacity of the fluid. The brine mass fraction of 'normal corrosive brine' (this term will be defined below) and non-corrosive brine is specified by the input parameters $\tilde{X}_\mathrm{b}$ and $\hat{X}_\mathrm{b}$. For the following $\tilde{X}_\mathrm{b} < \hat{X}_\mathrm{b}$ is assumed, which implies that the brine mass fraction increases during the corrosion process, but in principle, the CORFL module also accepts $\tilde{X}_\mathrm{b} > \hat{X}_\mathrm{b}$.

If a corrosive fluid with a brine mass fraction X_b enters a corrodible grid element, the CORFL module changes the composition of the brine to that of non-corrosive brine (i. e. the brine mass fraction equals $\hat{X}_\mathrm{b}$). This means that non-corrosive brine leaves the grid element and grid elements on the downstream side are not corroded anymore. The physical image behind this model is that of a quick corrosion process which causes a sharp corrosion front inside the grid element.

In order to raise the brine mass fraction inside a corroding element, the CORFL module places a brine source and a water sink inside the grid element. The source and the sink ensure that the change of brine composition is an isovolumetric process. This is necessary to keep constant pore pressures.

The CORFL module uses the cumulative brine mass that has already been released from the brine source to determine the grid element's degree of corrosion. The degree of corrosion is zero for non-corroded materials and one for full corrosion. The permeability of a material changes in response to its degree of corrosion.

7.2 Source and sink calculation

The brine sources and water sinks that are calculated by the CORFL module usually have to be corrected at every Newton-Raphson iteration. If the brine inside a corroding grid element has a brine mass fraction of $X_{\mathrm{b}} \neq \hat{X}_{\mathrm{b}}$, the brine source and water sink at that location have to be adapted in order to achieve $X_{\mathrm{b}} = \hat{X}_{\mathrm{b}}$. Basically, the method for doing so is first to calculate the brine and water mass that should be present at $X_{\mathrm{b}} = \hat{X}_{\mathrm{b}}$ and then to add or subtract the missing or excessive masses. That is, if m_b is the current brine mass and $\hat{m}_b$ the brine mass required for $X_{\mathrm{b}} = \hat{X}_{\mathrm{b}}$, the brine source has to be corrected by adding the term

$$\frac{\hat{m}_b - m_b}{\mathrm{V}\,\Delta t} \tag{7.1}$$

where V is the element volume and Δt is the time step. In equation (7.1) the current brine mass m_b can be calculated from the secondary variables (`PAR` array) of TOUGH2 by

$$m_b = V \cdot \phi \cdot S_{\mathrm{l}} \cdot \rho_{\mathrm{l}}(X_{\mathrm{b}}) \cdot X_{\mathrm{b}} \tag{7.2}$$

with porosity ϕ, liquid saturation S_{l}, liquid densityρ_{l}, and the current brine mass fraction X_{b}.

Since the change of brine mass fraction must not affect the liquid volume, the product $V \cdot \phi \cdot S_{\mathrm{l}}$ has to remain the same. Hence, the intended brine mass is given by

$$\hat{m}_b = V \cdot \phi \cdot S_{\mathrm{l}} \cdot \rho_{\mathrm{l}}\left(\hat{X}_{\mathrm{b}}\right) \cdot \hat{X}_{\mathrm{b}} \tag{7.3}$$

with $\rho_{\mathrm{l}}\left(\hat{X}_{\mathrm{b}}\right)$ being the liquid density at $X_{\mathrm{b}} = \hat{X}_{\mathrm{b}}$. The density $\rho_{\mathrm{l}}\left(\hat{X}_{\mathrm{b}}\right)$ can be calculated using the formula implemented in the equation-of-state modules EOS7 and EOS7R, which refers to the density of pure water ρ_{w}:

$$\rho_{\mathrm{l}}(X_{\mathrm{b}}) = \frac{\rho_{\mathrm{w}} d}{d(1 - X_{\mathrm{b}}) + X_{\mathrm{b}}} \tag{7.4}$$

with

$$d = \frac{\rho_{\mathrm{b}}(p_0, T_0)}{\rho_{\mathrm{w}}(p_0, T_0)} \tag{7.5}$$

being the ratio between the density of the reference brine ρ_b and the density of pure water ρ_w at reference conditions p_0 and T_0.

In order to avoid calculating the pressure- and temperature-dependent density of pure water, ρ_w is substituted by applying equation (7.4) to the brine densities $\hat{X}_\mathrm{b}$ and X_b

$$\rho_\mathrm{l}(\hat{X}_\mathrm{b}) = \rho_\mathrm{l}(X_\mathrm{b}) \frac{1 - X_\mathrm{b} + \frac{X_\mathrm{b}}{d}}{1 - \hat{X}_\mathrm{b} + \frac{\hat{X}_\mathrm{b}}{d}}. \tag{7.6}$$

Combining this equation with equation (7.3) allows to calculate the intended brine mass from the given primary and secondary variables of TOUGH2:

$$\hat{m}_b = V\phi S_\mathrm{l} \rho_\mathrm{liq}(X_\mathrm{b}) \frac{1 - X_\mathrm{b} + \frac{X_\mathrm{b}}{d}}{1 - \hat{X}_\mathrm{b} + \frac{\hat{X}_\mathrm{b}}{d}} \hat{X}_\mathrm{b}. \tag{7.7}$$

The same procedure is used to calculated the intended water mass

$$\hat{m}_\mathrm{w} = V \cdot \phi \cdot S_\mathrm{l} \cdot \rho_\mathrm{l}(\hat{X}_\mathrm{b}) \cdot (1 - \hat{X}_\mathrm{b}), \tag{7.8}$$

which leads to

$$\hat{m}_\mathrm{w} = V\phi S_\mathrm{l} \rho_\mathrm{liq}(X_\mathrm{b}) \frac{1 - X_\mathrm{b} + \frac{X_\mathrm{b}}{d}}{1 - \hat{X}_\mathrm{b} + \frac{\hat{X}_\mathrm{b}}{d}} (1 - \hat{X}_\mathrm{b}). \tag{7.9}$$

7.3 Degree of corrosion

The *corrosion potential* γ is defined as the ratio between the volume of corroded solid and the liquid volume necessary for its complete corrosion. In the model, the reference liquid for the definition of the corrosion potential is a liquid characterized by a brine mass fraction of $\tilde{X}_\mathrm{b}$.

If V is the volume of a grid element and ϕ its porosity, $V(1 - \phi)$ is the solid volume and $V\phi$ the pore volume. If full liquid saturation is considered, the number of times the pore fluid has to be exchanged for a complete corrosion is

$$n = \frac{V(1-\phi)}{V\phi\ \gamma}\ . \tag{7.10}$$

If the pore water is exchanged by a corrosive liquid with brine mass fraction $\tilde{X}_\mathrm{b}$ and brine mass m_b, corrosion takes place. In order to raise the brine mass fraction to $\hat{X}_\mathrm{b}$ with brine mass $\hat{m}_b$ the brine source has to inject the brine mass of $\hat{m}_b - m_b$. Using equation (7.9), a complete corrosion of the solid would require a brine mass injection of

$$m_{\mathrm{b,max}} = n(\hat{m}_b - m_b) = nV\phi S_\mathrm{l}\rho_\mathrm{l}\big(\tilde{X}_\mathrm{b}\big)\left(\frac{1-\tilde{X}_\mathrm{b}+\frac{\tilde{X}_\mathrm{b}}{d}}{1-\hat{X}_\mathrm{b}+\frac{\hat{X}_\mathrm{b}}{d}}\hat{X}_\mathrm{b}\ -\tilde{X}_\mathrm{b}\right). \tag{7.11}$$

Let $m_{\mathrm{b,sum}}(t)$ be the brine mass that has been released by a brine source until time t. The degree of corrosion $g(t)$ of the grid element can now be calculated as

$$g(t) = \frac{m_{\mathrm{b,sum}}}{m_{\mathrm{b,max}}}. \tag{7.12}$$

7.4 Permeability change

We adopt the physical model of /BFS 09/ (p. 154 ff) for the permeability change of corroding seals. The model assumes a sharp, planar corrosion front, which is migrating through the grid element due to the homogenous percolation of the seal. The corrosion front separates fully corroded material from material without any corrosion. Kinetic effects which would smooth out the corrosion front are neglected. Please note that this model might not capture the true physics of seal corrosion for all types of seals, especially seal percolation and corrosion is highly localized.

Due to the assumption of a sharp corrosion front, the degree of corrosion g stands for the corroded fraction of a grid element. We leave the permeability of the element unchanged for $g = 0$. If $g = 1$, it is multiplied by a factor of $10^{\varepsilon_\mathrm{k}}$. For partly corroded materials we assume a series connection of corroded and non-corroded material, which gives us

$$f_{\mathrm{CORFL}} = \frac{1}{1-g(1-10^{-\varepsilon_\mathrm{k}})} \tag{7.13}$$

for the permeability modifier f_CORFL (see chapter 4.2).

7.5 Input data

Tab. 7.1 summarizes the input parameters of the CORFL module. All corrosion parameters are material-specific.

Tab. 7.1 Input format of data block CORFL

Line	Column	Content	Format	Unit	Description	Variable
1	1-5	`"CORFL"`	A5		Keyword	
2	1-5	Material name	A5		Name of compacting material (as defined in ROCKS) In the following, `M` is the material index.	
3	1-10	γ	G10.4	[1]	Corrosion potential (volume of solid corroded per solute volume)	`CORFL_UK(M)`
	11-20	*unused*				
	21-30	$\tilde{X}_\mathrm{b}$	G10.4	[1]	Brine mass fraction of corrosive brine (choose brine used in corrosion experiment in order to determine corrosion potential)	`CORFL_XB1(M)`
	31-40	$\hat{X}_\mathrm{b}$	G10.4	[1]	Brine mass fraction of non-corrosive brine (choose $\hat{X}_\mathrm{b} \neq \tilde{X}_\mathrm{b}$)	`CORFL_XB2(M)`
	41-50	*unused*				
	51-60	ε_k	G10.4	[1]	Permeability factor of fully corroded element is $10^{\varepsilon_\mathrm{k}}$	`CORFL_EK(M)`
Repeat line 2 to 3 for all corroding materials						

8 DEGRA: Time-dependent permeability changes

The major function of technical seals is to prevent fluid flow. There are many possible causes for seal degradation involving different physical or chemical processes. It is therefore difficult to define a general-purpose degradation module. For the sake of simplicity and generality, the DEGRA module simply models seal degradation as a time-dependent change of permeability. The functionality of the DEGRA module has not changed since TOUGH2-GRS version 00.

The DEGRA module influences the intrinsic permeability by introducing the permeability modifier f_{DEGRA} (see chapter 4.2). For each material under consideration, a time window $[t_1, t_2]$ for the degradation process has to be specified. f_{DEGRA} increases linearly from 1 to a target value F in this time window

$$f_{\mathrm{DEGRA}} = \begin{cases} 1 & \text{for } \ t \leq t_1 \\ 1 + \left(\dfrac{t - t_1}{t_2 - t_1}\right)(F - 1) & \text{for } \ t_1 < t < t_2 \\ F & \text{for } \ t \geq t_2 \end{cases} \tag{8.1}$$

Tab. 8.1 describes the data block of the DEGRA module.

Tab. 8.1 Input format of data block DEGRA

Line	Column	Content	Format	Unit	Description	Variable
1	1-5	`"DEGRA"`	A5		Keyword	
2	1-5	Material name	A5		Name of degrading material (as defined in ROCKS). In the following, `M` is the material index.	
2	6-15	t_1	G10.4	[sec]	Start time of permeability change t_1	`DEGRA_T1(M)`
2	16-25	t_2	G10.4	[sec]	End time of permeability change t_2	`DEGRA_T2(M)`
2	26-35	F	G10.4	[1]	Factor F for the permeability at the time t_2	`DEGRA_F(M)`
Repeat line 2 for all material areas concerned.						

9 PRLIM: Gas pressure limitation

Fluid flow models for repositories in rock salt often do not consider the host rock, which is impermeable, but restrict the model domain to the excavated cavities and technical barriers. However, salt rock may become permeable to gas flow if gas pressures approach or exceed the minimal principal stress. Gas pressure opens microscopic or macroscopic pathways through the rock above a certain threshold pressure. The overall effect is a pressure limitation approximately around this threshold pressure.

The PRLIM module introduces a limiting pressure in order to simulate this process. If gas pressures exceed this pressure, they are clipped to the limiting value. The accumulated gas mass that is lost by this procedure is registered and accumulated. The PRLIM module keeps track of the cumulated gas mass that is lost by this procedure. The module's functionality has not changed since TOUGH2-GRS version 00a.

The PRLIM module introduces the minimal principal stress field $\sigma_{\min}$, which varies linearly with depth z

$$\sigma_{\min}(z) = \sigma_{\min,\mathrm{ref}} + (z - z_{\mathrm{ref}})\frac{\mathrm{d}\sigma_{\min}}{\mathrm{d}z}. \tag{9.1}$$

$\sigma_{\min,\mathrm{ref}}$ is the minimal principal stress at the reference depth z_{ref} and $\mathrm{d}\sigma_{\min}/\mathrm{d}z > 0$ is the vertical gradient of the minimal principal stress. Neglecting that stresses may change in the vicinity of mining cavities, $\sigma_{\min}(z)$ is taken as a reference pressure in order to define the threshold pressure for gas entry into the rock p_{entry}. From a physical point of view, the entry pressure does not need to equal the minimal principal stress because the opening of microscopic pathways may already take place at lower pressures. In order to define the entry pressure, a pressure offset Δ_{dilat} is introduced

$$p_{\mathrm{entry}} = \sigma_{\min} - \Delta_{\mathrm{dilat}}. \tag{9.2}$$

Parameter Δ_{dilat} can be defined separately for every material. Tab. 9.1 shows the input data block of module PRLIM.

Tab. 9.1 Input format of data block PRLIM

Line	Column	Content	Format	Unit	Description	Variable
1	1-5	`"PRLIM"`	A5		Keyword	
2	1-10	z_{ref}	G10.4	[m]		`PRLIM_ZREF`
2	11-20	$\sigma_{\mathrm{min,\,ref}}$	G10.4	[Pa]		`PRLIM_P`
2	21-30	$\frac{d\sigma_{\mathrm{min}}}{dz}$	G10.4	[Pa/m]		`PRLIM_DPDZ`
3	1-5	Material name	A5		Activate PRLIM for this material. `M` ist he material index in the following	
3	6-15	Δ_{dilat}	G10.4	[Pa]	Δ_{dilat}	`PRLIM_POFFSET(M)`
Repeat line 3 for all materials that introduce pressure limitation						

10 GCOMP: Primary non-condensable gas component

EOS7 and EOS7R use air as the primary non-condensable gas component, which usually is an appropriate choice shortly after the closure of a repository. However, due to gas-producing processes, other gas components may evolve and dominate the air component. For this reason, the GCOMP module allows to change the main characteristics of the primary non-condensable gas component in terms of absolute molecular weight M_{gas}, specific enthalpy h_{gas}, the inverse Henry constant H_{inv} , and the viscosity of the gas-vapour mixture. Input parameters of the GCOMP module are summarized in Tab. 10.1 and will be explained in the following.

Tab. 10.1 Input format of data block GCOMP

Line	Column	Vontent	Format	Unit	Description	Variable
1	1-5	`"GCOMP"`	A5		Keyword	
2	1-10	M_{ncg}	G10.4	[kg/mol]	Absolute molecular weight	GCOMP_AM
2	11-20	h_{gas}	G10.4	[J/mol]	Specific enthalpy	GCOMP_CV
2	21-30	a_1	G10.4	[Pa^{-1}]	Coefficient 1 for the calculation of the inverse Henry constant	GCOMP_HC1
2	31-40	a_2	G10.4	[Pa^{-1} °C^{-1}]	Coefficient 2 for the calculation of the inverse Henry constant	GCOMP_HC2
2	41-50	a_3	G10.4	[Pa^{-1} °C^{-2}]	Coefficient 3 for the calculation of the inverse Henry constant	GCOMP_HC3
2	51-60	$H_{\mathrm{inv,min}}$	G10.4	[Pa^{-1}]	Minimum value for the inverse Henry constant	GCOMP_HCMIN
2	61-70	`"Air-H2O"`, `"H2-H2O"` or `"H2"`	A10		Viscosity function selector	GCOMP_VISFUN

10.1.1 Absolute molecular weight

The absolute molecular weight of the non-condensable gas can be changed by means of the input parameter M_{ncg} (`GCOMP_AM`), which has to be defined in units of kg/mol (the absolute molecular weight variables `AMA` of EOS7 and EOS7R are defined in units of g/mol). Since the composition of the main gas component may change with time and place, M_{ncg} might not be the correct molecular weight for the entire simulation. However, many processes are not sensitive to errors in molecular weight if the amount of substance is correct. This applies to

- partial pressures (because EOS7 and EOS7R assume ideal gasses),
- the Henry law (which only depends on the amount of substance),
- advective flow of the gas component (because the mass of dissolved gas usually is too small to influence fluid density), and
- diffusive flow of the gas component (which only depends on the amount of substance).

Simulations may therefore use erroneous molecular weights of the primary non-condensable gas component. The user merely has to place the correct amount of substance in the model and has to interpret the output of gas mass in terms of amount of substance.

10.1.2 Specific enthalpy

By setting $h_{\text{gas}} > 0$ (`GCOMP_CV`), the specific enthalpy will be changed to h_{gas}. If $h_{\text{gas}} \leq 0$, the default value of the respective EOS module is used, which is 733 J/mol for EOS7 and EOS7R (air), and 10,170 J/mol for EOS5 (hydrogen).

10.1.3 Inverse Henry constant

The GCOMP module calculates a temperature-dependent inverse Henry constant using cubic polynomial with a lower cut-off $H_{\text{inv,min}}$

$$H_{\text{inv}} = \max\left(H_{\text{inv,min}}, a_1 + a_2\vartheta + a_3\vartheta^2\right), \tag{10.1}$$

where ϑ is the temperature in degree Celsius. The equation is applied if $H_{\text{inv}} \neq 0$. Otherwise, the inverse Henry constant implemented in the respective EOS module is used.

10.1.4 Gas phase viscosity

The GCOMP module offers three viscosity functions for vapour-bearing gases of the following composition:

1. **Hydrogen only (no vapour)**

   `GCOMP_VISFUN` = "H2" selects the viscosity function of the EOS5 module.

2. **Air and vapour**

   `GCOMP_VISFUN` = "Air-H2O" selects the viscosity function of the EOS7 and EOS7R module.

3. **Hydrogen and vapour**

   This viscosity function is selected by `GCOMP_VISFUN` = "H2-H2O".

Viscosity function 3 is based on a temperature-dependent viscosity function of /LAN 69/ for a $\mathrm{H_2}$-vapour mixture, which is extended by a pressure-dependent correction $h(p)$

$$\eta_{\mathrm{H_2}}\left(T,p,x_{\mathrm{H_2}}\right) = f(T)\,\cdot g\left(x_{\mathrm{H_2}}\right)\,\cdot h(p)\,\cdot 191.8\,\cdot\,10^{-7}\,\mathrm{Pa\,s}. \tag{10.2}$$

T is the temperature in Kelvin, p is the gas pressure, and $x_{\mathrm{H_2}}$ the mass fraction of $\mathrm{H_2}$. The function $f(T)$ is given by

$$f(T) = \left(\frac{T}{273.16\ \mathrm{K}}\right)^{n} \tag{10.3}$$

with

$$n = x_{\mathrm{H_2}}\left(0.674 - 0.034\cdot 10^{-4}\frac{\vartheta}{1\ °C}\right) + x_{\mathrm{H_2O}}\left(1.082 - 0.25\cdot 10^{-4}\frac{\vartheta}{1\ °C}\right). \tag{10.4}$$

Function $g\left(x_{\mathrm{H_2}}\right)$ is defined by

$$g\left(x_{\mathrm{H_2}},x_{\mathrm{H_2O}}\right) = \left(1+\sqrt{2\left(2.016\,x_{\mathrm{H_2}} + 18.016\,x_{\mathrm{H_2O}}\right)}\right)\frac{0.6355\,x_{\mathrm{H_2}} + 0.439\,x_{\mathrm{H_2O}}}{4.36\,x_{\mathrm{H_2}} + 6.31\,x_{\mathrm{H_2O}}} \tag{10.5}$$

with $x_{\mathrm{H_2O}} = 1 - x_{\mathrm{H_2}}$.

The pressure-dependency of the viscosity

$$h(p) = \left(\frac{p}{101325\ \mathrm{Pa}}\right)^{0.005} \tag{10.6}$$

has been derived from the one implemented in the viscosity routine of EOS5 for pure hydrogen and therefore has to be regarded as an approximation without direct experimental basis.

11 PTIME: High-precision printout times

Module PTIME has the same function as data block TIMES except for the higher precision of time values and the extended limit of 10000 time entries. If a data block PTIME has been defined, it will replace an existing TIMES block. Tab. 11.1 shows the structure of the PTIME data block, which is very similar to that of the data block TIMES.

Tab. 11.1 Input format of data block PTIME

Line	Column	Content	Format	Unit	Description	Variable
1	1-5	`"PTIME"`	A5		Keyword	
2	1-5	Number of times in table	I5	[1]	Must not be larger than 100	ITI
2	6-10	Number of times to be used	I5	[1]	Must be larger or equal to ITI. If larger than ITI, printout times will be distributed according to TINTER.	ITE
2	11-20	Maximum time step	G10.4	[s]	Upper limit for the time step size after first printout time. Inactive if zero.	DELAF
2	21-30	Time increment for printouts ITI to ITE	G10.4	[s]	Only active if ITE > ITI	TINTER
≥3	1-20, 21-40, 41-60, 61-80	Table of printout times (ascending order)	4G20.14	[s]		TIS(I)
Repeat						

12 FISS: Gas pathway dilation

Gas transport in host rock with very low permeability, like clay stone or rock salt, becomes more efficient as gas pressure rises. The increase of transport efficiency is not a gradual process but one that is controlled by pressure thresholds for the opening of microscopic pathways, a process often referred to as “pathway dilation”. At even higher pressures, macroscopic fracturing may occur.

The FISS module introduces several models for a threshold-driven advective transport of the gas phase. It offers several models for the change of pressure thresholds (softening), porosity, and gas permeability, which can be combined.

The FISS module uses the porosity change mechanism that has been explained in chapter 4.1. Changes of gas permeability do not affect the intrinsic permeability but are implemented by introducing a new permeability term to the flow equation for the gas phase. This will be explained in chapter 12.3.

12.1 Pressure thresholds and softening models

The pressure p_{thr} is defined as the threshold pressure for gas flow in dilating microfissures. If the threshold pressure decreases with increasing gas pressure, this is termed as softening process. The decrease of the pressure threshold Δp_{thr} between successive time steps is controlled by several softening models, which are selected via the switches `ifsoft` and `fsoft(3)` (see Tab. 12.1). Input parameter `ifsoft` selects the model describing the equilibrium between gas pressure and pressure threshold. Input parameter `ifsoft(3)` introduces an optional time-dependency of the softening process. First of all, the required pressure change for complete equilibration $\widetilde{\Delta p}_{\mathrm{thr}}$ is calculated and forms the basis for the actual pressure change Δp_{thr}, which may be time-dependent.

Models describing the equilibrium

- `ifsoft=0`: constant threshold, no softening.

$$\widetilde{\Delta p}_{\mathrm{thr}} = 0 \tag{12.1}$$

- `ifsoft=1`: reversible linear softening. The threshold pressure p_{thr} decreases linearly from $p_{\mathrm{thr,ini}}$ to $p_{\mathrm{thr,ini}} - a_2$ as p increases from $p_{\mathrm{thr,ini}}$ to $p_{\mathrm{thr,ini}} + a_1$.

$$\widetilde{\Delta p}_{\mathrm{thr}} = \begin{cases} -p_{\mathrm{thr}} + p_{\mathrm{thr,ini}} & \text{for} \quad x_{\mathrm{thr}} < 0 \\ -p_{\mathrm{thr}} - x_{\mathrm{thr}} \cdot a_2 + p_{\mathrm{thr,ini}} & \text{for} \quad 0 \le x_{\mathrm{thr}} \le 1 \\ -p_{\mathrm{thr}} - a_2 + p_{\mathrm{thr,ini}} & \text{for} \quad x_{\mathrm{thr}} > 1 \end{cases} \tag{12.2}$$

with

$$x_{\mathrm{thr}} = \frac{p - p_{\mathrm{thr,ini}}}{a_1}. \tag{12.3}$$

- `ifsoft=2`: irreversible linear softening. Same as `ifsoft=2` but positive values of $\widetilde{\Delta p}_{\mathrm{thr}}$ are clipped to zero.

Model introducing time-dependency

- `ifsoft(3)=0`: no time-dependency,

$$\Delta p_{\mathrm{thr}} = \widetilde{\Delta p}_{\mathrm{thr}}. \tag{12.4}$$

- `ifsoft(3)=1`: positive rate limit a_4,

$$\Delta p_{\mathrm{thr}} = \min\left(\left|\widetilde{\Delta p}_{\mathrm{thr}}\right|, a_4\, \Delta t\right)\ \mathrm{sgn}\left(\widetilde{\Delta p}_{\mathrm{thr}}\right). \tag{12.5}$$

- `ifsoft(3)=2`: rate decrease according to a decay function with half-life a_4,

$$\Delta p_{\mathrm{thr}} = \widetilde{\Delta p}_{\mathrm{thr}} \left(1 - 2^{-\frac{\Delta t}{a_4}}\right). \tag{12.6}$$

12.2 Fissure porosity models

A similar modelling approach is used for the relationship between porosity and pressure. Models for the state of equilibrium are selected by means of parameter `ifpor`. They determine the porosity change between successive time steps $\widetilde{\Delta\phi}_{\mathrm{fiss}}$, which would be required for equilibration. The actual porosity change $\Delta\phi_{\mathrm{fiss}}$ is derived from $\widetilde{\Delta\phi}_{\mathrm{fiss}}$ and may include time-dependent effects. Models for time-dependency are selected by means of parameter `fpor(matno,5)`.

Models describing equilibrium

- `ifpor=0`: No porosity change.

$$\widetilde{\Delta\phi}_{\mathrm{fiss}} = 0 \tag{12.7}$$

- `ifpor=1`: Porosity switches between two values. Should be used in combination with a kinetic model in order to avoid oscillations,

$$\widetilde{\Delta\phi}_{\mathrm{fiss}} = \begin{cases} b_1 - \phi_{\mathrm{fiss}} & \text{for} \quad p > p_{\mathrm{thr}} \\ -\phi_{\mathrm{fiss}} & \text{for} \quad p \leq p_{\mathrm{thr}} \end{cases}. \tag{12.8}$$

- `ifpor=2`: Power-law behaviour with optional cut-off

$$\widetilde{\Delta\phi}_{\mathrm{fiss}} = \begin{cases} -\phi_{\mathrm{fiss}} + b_2 & \text{for} \quad p > p_{\mathrm{thr}} \wedge b_4 = 1 \\ -\phi_{\mathrm{fiss}} + b_2 (x_{\mathrm{por}})^{b_3} & \text{for} \quad p > p_{\mathrm{thr}} \wedge b_4 \neq 1, \\ -\phi_{\mathrm{fiss}} & \text{for} \quad p \leq p_{\mathrm{thr}} \end{cases} \tag{12.9}$$

with

$$x_{\mathrm{por}} = \frac{p - p_{\mathrm{thr}}}{b_1}. \tag{12.10}$$

Model introducing time-dependency

- `fpor(matno,5)=0`: no time-dependency,

$$\Delta\phi_{\mathrm{fiss}} = \widetilde{\Delta\phi}_{\mathrm{fiss}}. \tag{12.11}$$

- `fpor(matno,5)=1`: positive rate limit b_6,

$$\Delta\phi_{\mathrm{fiss}} = \min\left(\left|\widetilde{\Delta\phi}_{\mathrm{fiss}}\right|, b_6\,\Delta t\right)\ \mathrm{sgn}\left(\widetilde{\Delta\phi}_{\mathrm{fiss}}\right). \tag{12.12}$$

- `fpor(matno,5)=2`: rate decrease according to an exponential decay function with half-life b_6,

$$\Delta\phi_{\mathrm{fiss}} = \widetilde{\Delta\phi}_{\mathrm{fiss}}\left(1 - 2^{-\frac{\Delta t}{b_6}}\right). \tag{12.13}$$

12.3 Permeability models

The flow of gas through micro fractures is controlled by an additional term $k_{\mathrm{dil}}\boldsymbol{K}_{\mathrm{dil}}$ in the flow equation for the gas phase

$$F_{\mathrm{gas}} = -\frac{\rho_{\mathrm{gas}}}{\mu_{\mathrm{gas}}}\left(k\, k_{\mathrm{r,gas}}\boldsymbol{I} + \underbrace{k_{\mathrm{dil}}\boldsymbol{K}_{\mathrm{dil}}}_{\substack{\text{micro crack}\\ \text{permeability}}} \right)\left(\nabla p_{\mathrm{gas}} - \rho_{\mathrm{gas}} g\right). \tag{12.14}$$

This means that the FISS module requires a modification of the TOUGH2 core (`MULTI` routine). In the above equation, k is the intrinsic and $k_{\mathrm{r,gas}}$ the relative gas permeability, ρ_{gas} is the density, μ_{gas} the dynamic viscosity of the gas phase, and g the vector of gravitational acceleration. The new variable k_{dil} is the pressure-dependent gas permeability of the fissure network, and the tensor $\mathbf{K}_{\mathrm{dil},i}$ introduces anisotropy. It is assumed that micro cracks which are opened by an intruding gas phase remain filled with air, and for this reason no relative gas permeability is introduced for the micro crack network.

The two terms in the large brackets of equation (12.14) are the permeability of the original pore space and of the secondary pore space gained by pathway dilation. In a space-discretised version of equation (12.14), these two terms are weighted separately, with $k_{\mathrm{dil}}\boldsymbol{K}_{\mathrm{dil}}$ always being upstream-weighted.

k_{dil} is defined as a power function with a lower and an optional upper cut-off

$$k_{\mathrm{dil}} = \begin{cases} c_3 & \text{for} \quad y < 0 \\ y^{c_5}\cdot(c_4 - c_3) + c_3 & \text{for} \quad (0 \le y \le 1) \vee (y > 1 \wedge c_6 = 0), \\ c_4 & \text{for} \quad y > 1 \wedge c_6 \neq 0 \end{cases} \tag{12.15}$$

with

$$y = \frac{x - c_1}{c_2 - c_1}. \tag{12.16}$$

Parameters $c_1, \dots, c_6$ refer to the internal variables `fper(1)`, …, `fper (6)`. Parameter c_6 enables the upper cutoff of k_{dil}. The meaning of x, c_1, and c_2 depends on the choice of `ifper`:

- `ifper=0`: $x = c_1$. Constant permeability $k_{\mathrm{dil}} = c_3$.
- `ifper=1`: $x = \phi_{\mathrm{fiss}}$ (k_{dil} is a function of fissure porosity)
- `ifper=2`: $x = p - p_{\mathrm{thr}}$ (k_{dil} is a function of gas pressure)

- `ifper=3`: $x = p_{\mathrm{thr}}$ (k_{dil} is a function of the pressure threshold)

12.4 Input data

Tab. 7.1 shows the input parameters of the FISS module.

Tab. 12.1 Input format of data block FISS

Line	Column	Content	Format	Unit	Description	Variable
1	1-5	`"FISS"`	A5		Keyword	
2	1-5	Material name	A5		Name of compacting material (as defined in ROCKS) In the following, `M` is the material index.	
3	1-10		G10.4	[Pa]	Initial pressure threshold	`ptini(M)`
4	1-5		I5		Softening model	`ifsoft(M)`
	11-20		G10.4		Parameter 1 of softening model	`fsoft(M,1)`
…						
	71-80		G10.4		Parameter 7 of softening model	`fsoft(M,7)`
5	1-5		I5		Porosity model	`ifpor(M)`
	11-20		G10.4		Parameter 1 of porosity model	`fpor(M,1)`
…						
	71-80		G10.4		Parameter 7 of porosity model	`fpor(M,7)`
6	1-5		I5		Permeability model	`ifper(M)`
	11-20		G10.4		Parameter 1 of permeability model	`fper(M,1)`
…						
	71-80		G10.4		Parameter 7 of permeability model	`fper(M,7)`
7	31-40		G10.4	[1]	Permeability factor for ISO=1	`fiss_tensor(M,1)`
	31-40		G10.4	[1]	Permeability factor for ISO=2	`fiss_tensor(M,2)`
	31-40		G10.4	[1]	Permeability factor for ISO=3	`fiss_tensor(M,3)`
Repeat line 2 to 7 for all dilating materials						

12.5 Amendments to version 00

The FISS module has been subject to major changes. A user interface has been implemented and the models for softening, porosity change and permeability have been completely revised. Porosity change is now part of the implicit solution of the differential equations calculated in order to achieve higher accuracy.

13 CNTRL: Simulation controls

The CNTRL module controls the simulation with regard to abortion criteria, automatic time stepping, and printout. The structure of data block CNTRL is shown in Tab. 13.1.

Tab. 13.1 Input format of data block CNTRL

Line	Column	Content	Format	Unit	Description	Variables (Long names! Ignore line feeds.)
1	1-5	`"CNTRL"`	A5		Keyword	
w2	1-10	Do not stop after 10 iterations with ITER=1?	I10		0: no 1: yes	`cntrl_ignore10TimesIterEq1`
2	11-20	Do not stop after convergence failure subsequent to two iterations with ITER=1?	I10		0: no 1: yes	`cntrl_ignoreConvFailureAfter2TimesIterEq1`
3	1-10	Stop execution if time step width remains `N` times below `Tmin`?	G10.4	sec	`Tmin` (inactive if Tmin=0)	`minStepSize`
3	11-20		I10		`N`	`stopAfter`
4	1-10	Factor for the increase of time step width if `ITER` < `MOP(16)`	G10.4		TOUGH2-standard: 2 recommended: 1.2	`cntrl_dtfacQuickIter`
4	11-20	Factor for the increase of time step width if `ITER` >= `MOP(16)`	G10.4		TOUGH2-standard: 1 recommended: 1.1	`cntrl_dtfacSlowIter`
5	1-10	Number of time steps that avoid last time step width with convergence failure	I10		0: inactive	`cntrl_avoiddt`
6	1-10	Adapt time stepping in order to avoid small time step widths before printout times	I10		0: no 1: yes	`cntrl_softlanding`
7	1-10	Maximum relative change of the generation rate if linear interpolated tables are used (rate change per rate)	G10.4		0: inactive	`cntrl_MaxRelGenRateChange`
8	1-10	Decimal places of real numbers	I10		0: switches to 12, <0: binary output	`cntrl_decimalplaces`
9	1-10	Stride for time series printouts (FOFT, COFT etc.)	I10		0: switches to 1 (=print every time step)	`cntrl_ioft`
10	1-10	Minimum time between two time series printouts divided by time	G10.4		0: inactive	`cntrl_MinRelDur`
11	1-10	Printout run-time information to STDOUT	I10		0: no 1: yes	`cntrl_monitoring`

13.1 Abortion controls

13.1.1 Do not exit on equilibrium

TOUGH2 v. 2.0 uses two criteria to detect static or dynamic equilibria: the code aborts if either ten iterations with `ITER`=1 have occurred or if there is a convergence failure after two subsequent iterations with `ITER`=1. Since version 00 of TOUGH2-GRS, the CNTRL module allows to disable these criteria by means of the two parameters `cntrl_ignore10TimesIterEq1` and `cntrl_ignoreConvFailureAfter2TimesIterEq1`, for example if temporary equilibria are expected during the simulated time frame. However, this feature has become obsolete due to the enforcement of at least two iterations (see chapter 2.4.6).

13.1.2 Minimum time step width

Some process modules of TOUGH2-GRS may induce fast temporal changes of the flow field. The CORFL module, for instance, may trigger rapid flow processes after seals have failed. Fast processes usually lead to a drastic reduction of time step width up to a degree where simulations become inefficient and should be terminated. In order to avoid stagnant simulations with very small time step widths, the parameters `minStepSize` and `stopAfter` can be used to define an additional abortion criterion. TOUGH2-GRS will terminate the simulation if the time step width falls below the value of `minStepSize` exactly `stopAfter` times in succession.

13.2 Time step controls

13.2.1 Step width incrementation

TOUGH2 increases the time step width if the number of Newton-Raphson iterations `ITER` is smaller than or equal to `MOP(16)`. The standard increment is a factor of 2. This factor can be adjusted by the CNTRL module by means of parameter `dtfacQuickIter`.

The automatic time stepping control of TOUGH2 works well if the bandwidth of time step widths that meet the criterion `NOITE` ≥ `ITER` > `MOP(16)` is of moderate size. If it is too large, time steps will not increase, although they could without causing convergence failures. In this case, the incrementing factor `dtfacSlowIter` can be used to approximate the upper step width limit (`ITER` > `NOITE`) slowly (see above). It is mandatory to set 1 ≤ `dtfacSlowIter` < `dtfacQuickIter`.

13.2.2 Avoiding convergence failures

If the time step bandwidth of the criterion `NOITE` ≥ `ITER` > `MOP(16)` is too small or even zero, the simulation will suffer from many convergence failures because it will often run into the state `ITER` > `NOITE`. In order to avoid this, TOUGH2-GRS can be instructed to avoid the last time step width that has produced a convergence failure for at least `cntrl_avoiddt` time steps.

13.2.3 Avoiding small time steps before printout times

Printout times specified in data block TIMES can lead to a sharp reduction of time step width. The reason is that time steps which are close to printout times are clipped in order to reach the specified time.

An automatic step width adjustment has been introduced to the CNTRL module to alleviate the effects of this clipping. It is switched via parameter `cntrl_softlanding`. If enabled, the CNTRL module checks whether the next printout time is nearer than a quadruple of the current time step size. If so, the time step width is slightly reduced in order to divide the time to the next printout into almost equal parts.

13.2.4 Avoiding large changes in gas generation

If sinks and sources are introduced using linear interpolated tabular data (`MOP(12)`=0), large time steps can lead to inaccurate mass injection or production. Setting parameter `cntrl_MaxRelGenRateChange` to a positive value will reduce the time step width in order to limit the absolute relative change of the generation rate to `cntrl_MaxRelGenRateChange` (rate change per rate).

13.3 Output controls

13.3.1 Precision of time series printouts

Data blocks FOFT, COFT and GOFT introduce time series printouts for specified elements, connections, and sources, respectively. TOUGH2-GRS also introduces domain-specific printouts by means of data block DOFT (see chapter 16).

Time series printouts can grow very large if there are a lot of time steps. The size of these printouts can be controlled by parameter `cntrl_decimalplaces` of the CNTRL module, which sets the number of decimal places. Note that zero values will always be printed as "0." no matter how many decimal places have been prescribed. If `cntrl_decimalplaces` is set to a negative value, a binary printout is produced using the file names `FOFT.bin`, `COFT.bin` and so on. The binary output can be converted to ASCII format by means of the tool `translate.f90`.

13.3.2 Reducing the amount of time series printouts

An effective way to decrease the size of time series printouts is to skip time steps. Parameter `cntrl_ioft` defines a minimum time step difference between time series printouts. Parameter `cntrl_MinRelDur` multiplied by the current time introduces a minimum time between time series printouts.

13.3.3 Additional printouts

Parameter `cntrl_monitoring` prints additional run-time information to STDOUT. The printing routine can be found in file `module_test.f90` in routine `test_diagnostics`. The user is encouraged to adapt the routine according to his needs.

14 RANGE: Limiting primary variables

Some EOS modules run into problems when phases appear or disappear and the set of primary variables is changed. A possible workaround is to avoid changes in the number of phases. Basically this means to restrict the gas saturation, which is a primary variable, to a two-phase state.

Another problem is that brine mass fractions can run out of bounds if water saturation is very low. The reason is that errors in the brine mass fraction do not violate the absolute convergence criterion if there is only very little liquid in the pore space. Exploding brine mass fractions need not be a numerical problem, but they clearly diminish the value of a brine mass fraction printout. One might therefor want to restrict the brine mass fraction to reasonable values.

The RANGE module introduces several ways to restrict the range of gas saturation and brine mass fraction. Some of the measures involve clipping of primary variables with violation of mass conservation. For this reason the physical system parameter should be chosen in such a way that clipping is avoided as much as possible.

The parameters of the RANGE module are listed in Tab. 14.1. The following chapters will elaborate on their use.

Tab. 14.1 Input format of data block RANGE

Line	Column	Content	Format	Unit	Variables
1	1-5	Keyword `"RANGE"`	A5		
2	1-10	**Limit brine mass fraction?** 0: No. 1: Yes. Restrict to interval $[X_{\mathrm{b,min}},\ X_{\mathrm{b,max}}]$.	G10.4	[1]	`RANGE_restrictXb`
2	11-20	$\boldsymbol{X}_{\mathbf{b,min}}$	G10.4	[1]	`RANGE_XBmin`
2	21-30	$\boldsymbol{X}_{\mathbf{b,max}}$	G10.4	[1]	`RANGE_XBmax`
3	1-10	**Constant brine mass fraction?** 0: No 1: Yes.	G10.4	[1]	`RANGE_equilibrateXb`
3	11-20	$\boldsymbol{X}_{\mathbf{b,eq}}$: Brine mass fraction in equilibrium.	G10.4	[1]	`RANGE_EQ_Xbr`
3	21-30	$\boldsymbol{R}_{\mathbf{b/w}}$: Brine mass per mass of contained water.	G10.4	[1]	`RANGE_EQ_` `BrineMassPerWaterMass`
4	1-10	**Limit gas saturation?** 0: No. 1: Yes. Restrict to interval $[S_{\mathrm{g,min}},\ S_{\mathrm{g,max}}]$.	G10.4	[1]	`RANGE_restrictSg`

4	11-20	**$S_{\mathrm{g,min}}$**	G10.4	[1]	`RANGE_SGmin`
4	21-30	**$S_{\mathrm{g,max}}$** with $S_{\mathrm{g,min}} < S_{\mathrm{g,max}}$	G10.4	[1]	`RANGE_SGmax`
5	1-10	**Dim saturation vapour pressure?** 0: No. 1: Yes. Reduce saturation vapour pressure linearly as gas saturation rises from $S_{\mathrm{g,dim}}$ to $S_{\mathrm{g,zero}}$. $S_{\mathrm{g,dim}} < S_{\mathrm{g,zero}} \leq 1$	G10.4	[1]	`RANGE_reducePs`
5	11-20	**$S_{\mathrm{g,dim}}$**	G10.4	[1]	`RANGE_PS_SGmin`
5	21-30	**$S_{\mathrm{g,zero}}$**	G10.4	[1]	`RANGE_PS_SGmax`

14.1 Limiting brine mass fraction

Brines are defined as a mixture of pure water and a reference brine by the equation-of-state modules EOS7 and EOS7R. The mass ratio between reference brine and liquid defines the brine mass fraction. Following this concept, the brine mass fraction cannot be larger than 1 without sacrificing the idea of mixing two liquids. However, if the specified reference brine is regarded as an intermediate product of a mixing process involving pure water and an unspecified brine, brine mass fractions beyond 1 can be physically plausible and TOUGH2 does not object to such values.

In order to obtain the primary variables of the next time step, TOUGH2 has to estimate the brine mass fraction too. An inaccurate estimation does not necessarily lead to a violation of the convergence criteria since these focus on mass conservation, not on mass fraction conservation. If liquid saturation is low, the absolute errors of the brine mass will be small too, which implies that TOUGH2 tolerates large errors in brine mass fraction if grid elements are almost dry. Consequently, the estimated brine mass fraction may by far exceed the value of 1. This is not a problem in terms of mass conservation but may contradict the idea of mixing two types of liquid. The presence of grid elements with very high brine mass fractions may also complicate the interpretation of the simulation output.

The RANGE module allows restricting the brine mass fraction to the interval $[X_{\mathrm{b,min}}, X_{\mathrm{b,max}}]$ by clipping the values of the corresponding primary variable. This should not lead to a significant violation of mass conservation if the reasons for mass fractions larger than 1 are those which have been describe above (very low liquid saturation). Setting $X_{b,min} = 0$ and $X_{b,min} = 1$ should be appropriate in many cases unless other process modules like CORFL demand that mass fractions larger than 1 should be tolerated.

14.2 Constant brine mass fraction

The RANGE module introduces the parameters `RANGE_equilibrateXb`, `RANGE_EQ_Xbr`, and `RANGE_EQ_BrineMassPerWaterMass` in order to keep the brine mass fraction at a constant value. The intention is to simulate a chemical equilibrium of the solution with the salt rock, which would result in a constant brine mass fraction.

The RANGE module maintains preserves a constant brine mass fraction while conserving the total water mass (including the water mass implicitly contained in the reference brine). The physical model has been described in /NAV 13b/. However, it is recommend not to use this functionality anymore. The adjustment of the brine mass fraction to the target value of `RANGE_EQ_Xbr` is not part of the implicit calculation of TOUGH2 at the cost of accuracy. Further on, the constant brine mass functionality has only been tested for two phase states.

Also, keeping the brine mass fraction constant is incompatible with the CORFL module, which uses the brine mass fraction as a marker for the fluid's capability to corrode the seal. The interference with the CORRO module, which is able to change the brine mass fraction, has not yet been well tested.

14.3 Clipping gas saturation and reducing vapour pressure

Setting `RANGE_restrictSg` = 1 will induce the code to clip the primary variable *gas saturation* to the interval `RANGE_SGmin` to `RANGE_SGmax` after every time step in order to maintain a two-phase state. This clipping operation violates mass conservation and should therefore be avoided as much as possible by an appropriate design of the flow system. The system should be designed such that it remains in a two-phase state by itself, for example by introducing residual saturations to the relative permeability function. Capillary pressure may also prevent desaturation of elements. Yet, the disappearance of a phase cannot be entirely prevented by these measures for the following reasons:

- Even an immobile gas phase can be displaced by gas dissolution and advective transport of dissolved gas in the liquid phase.
- Even an immobile liquid phase can be displaced by evaporation and advective transport of vapour in the gas phase.
- Backfill compaction, induced by the COMP module, compresses the gas phase and increases gas dissolution. If liquid saturation is near to one, the gas phase can disappear.

In order to prevent evaporation of the entire liquid phase, the RANGE module can reduce the saturation vapour pressure. By setting `RANGE_reducePs` = 1 the vapour pressure is reduced linearly to zero as gas saturation increasees from `RANGE_PS_SGmin` to `RANGE_PS_SGmax`. The user should set `RANGE_PS_SGmax` < `RANGE_SGmax` in order to prevent the RANGE module from clipping the gas saturation (see Fig. 14.1).

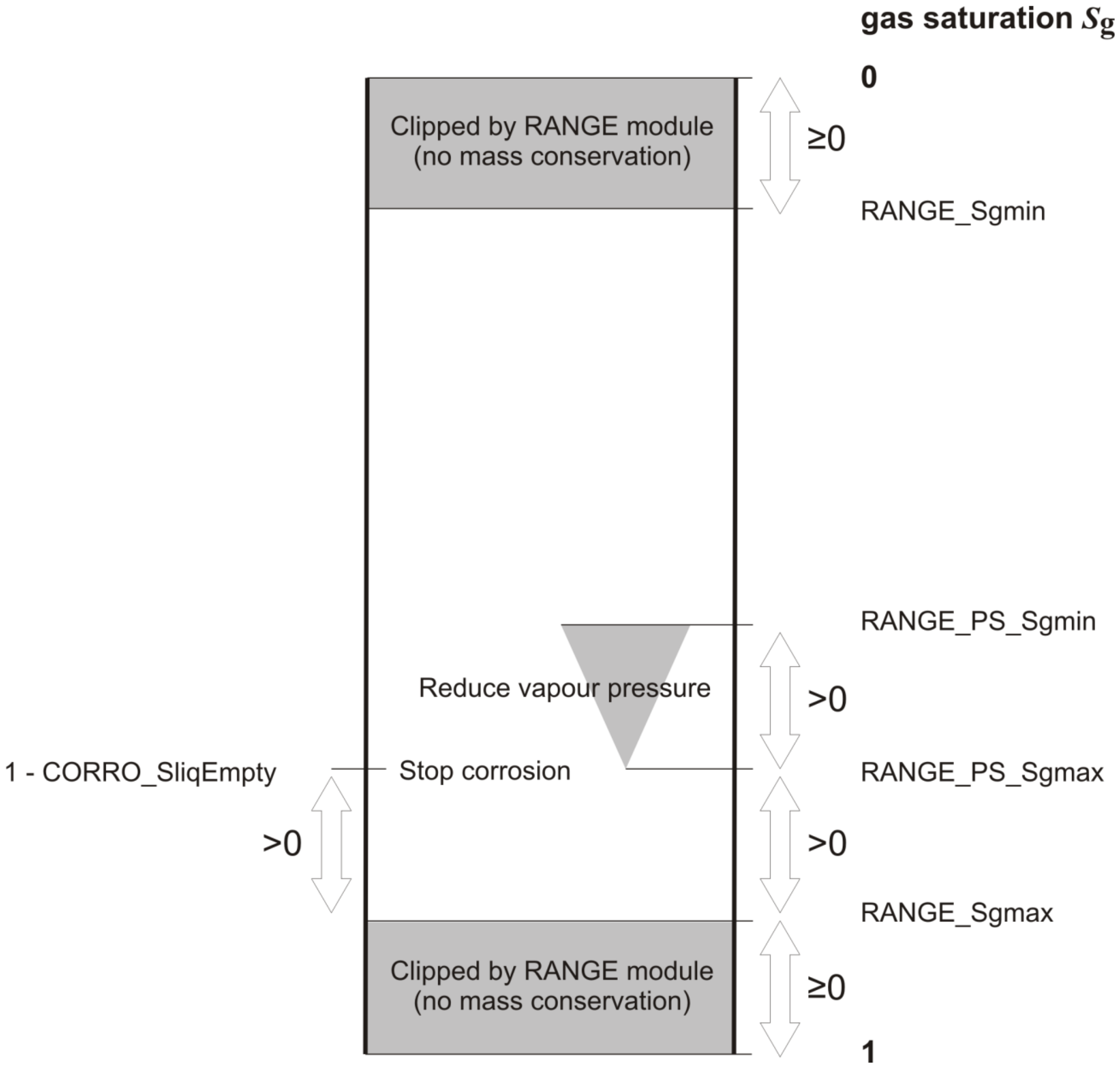

Fig. 14.1 Relationship between saturation limits of the RANGE and CORRO modules used to maintain two-phase states.

15 RN

The development of the radionuclide module (termed RN module in the following) was initiated by the conception that transported radionuclides in conceivable contaminant transport conditions do not influence fluid flow significantly because of their low concentrations. In this context, the approach of EOS7R to solve the mass balance equations for the air, water and brine components together with those for the radionuclides is probably not optimal because both sets of balance equations can be solved successively. Also, the standard TOUGH2 tolerance limits for residua are too large for radionuclide traces so that additional tolerance limits for radionuclides have to be introduced. However, a very low tolerance for radionuclide residua can have a negative effect on code performance if the mass estimation for main components continues to converge during the Newton iteration loop, thereby disturbing the estimation of radionuclide mass fractions.

The RN module therefore introduces radionuclides independently from the EOS module. Radionuclides are transported by diffusion and advection using the phase flows and phase distributions previously calculated by TOUGH2. The RN module can be used in connection with any EOS module. Transport and linear adsorption of radionuclides are calculated by time integration using the Bulirsch-Stoer method /STO 80/. With this approach TOUGH2-GRS follows the approach implemented in the code MARNIE /MAR 02/, which is a flow and transport simulator designed for repositories in salt rock formations.

The RN module uses the same equations for advection and diffusion as TOUGH2/EOS7R. However, in contrast to TOUGH2/EOS7R, it has a more accurate implementation of radioactive decay because it assumes exponential decay during a time step instead of linear decay.

15.1 Decay and Adsorption of Nuclides

The radioactive decay of the RN module follows an exponential decay law for each radionuclide with

$$N(t+\Delta t) = N(t)e^{-\lambda \Delta t} \tag{15.1}$$

where N is the amount of the radionuclide, Δt is the length of the actual timestep in TOUGH2 and λ is the radioactive decay constant with

$$\lambda = \frac{\ln 2}{T_{1/2}}, \tag{15.2}$$

and $T_{1/2}$ is the half-life of the radionuclide. The radioactive decay is calculated explicitly before the TOUGH2 iteration step, i.e. outside the solvers for the TOUGH2 differential equations. As a direct consequence, the maximally allowed timestep Δt_{max} for the TOUGH2 solver must fulfill

$$\Delta t_{max} \ll T_{1/2}. \tag{15.3}$$

The default is $\Delta t_{max} \rightarrow \infty$ and Δt_{max} has to be modified in the input. In /HOT 16a/ an example calculation for the default Δt_{max} is given which nicely illustrates that the solution for the transport equations of the RNmod fails if Δt_{max} does not comply with equation (15.3).

In contrast to the EOS7R module, adsorption of radionuclides in RNmod is realized such that the adsorbed radionuclide mass is taken from the initially specified radionuclide mass and is not added to the initial mass like in EOS7R.

15.2 Input data

Tab. 15.1 shows the input parameters of the RN module.

Tab. 15.1 Input format of data block RN

Line	Column	Content	Format	Unit	Description	Variable
1	1-5	Number of radionuclides	A5		Total number of radionuclides	`RN_N`
Repeat the following lines for each radionuclide (RN) i						
2+(i-1)	1-10	Radionuclide name	A10		Name of radionuclide (RN)	`rn_NAME`
	11-21	$T_{1/2}$	G10.4	[s]	Half-life of RN	`rn_HALF`
	21-30	RN Daughter name	A10		Name of daughter RN	`rn_daughtername`
	31-40	Molecular weight	G10.4	[kg/mol]	Molecular weight	`rn_M`

Line	Co-lumn	Content	For-mat	Unit	Description	Variable
					of RN	
	41-50	Henry constant or inverse Henry constant	G10.4	[Pa] or [1/Pa]	Absolute value will be used. <0: use Henry constant >0: use inverse Henry constant	`rn_H`
	51-60	Solubility limit	G10.4	[kg/m³]	Solubility limit	`rn_Ceq`
	61-70	Diffusivity in gas	G10.4	[m²/s]	Diffusion constant in gas phase	`rn_Dg`
	71-80	Diffusivity in liquid	G10.4	[m²/s]	Diffusion constant in liquid phase	`rn_Dl`
The following line may be repeated for each material defined in ROCKS where a RN should be placed or retarded						
RN_N+2	1-10	”`PLACE`” or “`RETARD`”	A10		`PLACE` command places a RN mass into domain, `RETARD` command sets Kd value	
	11-20	Name of RN	A20		RN name. Blanks will select all RN	
	21-25	Material name	A5		Name of material as defined by card ROCKS	
	26-40	Initial mass or Kd value	G15.4	mass: [kg] concentration: [kg/m³] K_d: [m³/kg]	Place command: initial RN mass fpr active elements and initial RN concentration for inactive elements Retard command: K_d value	
	41-50	Time of release of RN	G10.4	[sec]	Place: time of release of RN retard: blank	

16 Output

TOUGH2 produces two principal types of output. The first one is a printout to the STDOUT channel, which can be redirected to a file by the user. The second one is a printout to the files FOFT, COFT, and GOFT (the DOFT file of TOUGH2-GRS also belongs to this category). The difference is that the printout to the STDOUT channel is a printout for all grid locations but only for a few times (which are specified via data block `TIMES` and parameter `MCYPR` of the `PARAM` card) whereas the printout to the *OFT files is a printout for all times but only for few elements or connections. Basically, the printout to the STDOUT channel is a snapshot of the entire system state whereas the FOFT, COFT, GOFT files are created for time series plots. See Tab. 16.1 for the FOFT file, Tab. 16.2 for the COFT file, and Tab. 16.3 for the GOFT file.

TOUGH2-GRS introduces a third type of output: the ELE_MAIN and CON_MAIN files (see Tab. 16.4 and Tab. 16.5). These files hold the information of the element- and connection-specific printout that is printed to the STDOUT channel but use the format of the *OFT files. The files ELE_MAIN and CON_MAIN have been introduced in order to simplify the postprocessing of the output data.

TOUGH2-GRS has added an additional time series file DOFT for domain specific printout. The DOFT file contains variables for selected materials (that have been defined in `ROCKS`). It is created if a data block `DOFT` is added to the input, which is followed by a list of materials (one per line), see Tab. 16.6. The content of the DOFT file is shown in Tab. 16.7. Please note that the first entry refers to the entire model (i. e. all materials) and the second entry to the first material, and so on.

With TOUGH2-GRS v01 the input parameter `KDATA` has received an extended functionality. Setting `KDATA` = 4 generates extended output with additional output parameters for the time series COFT and DOFT (see Tab. 16.2 and Tab. 16.7).

The realization of new process modules yields a significant growth in the number of output parameters. To reduce the resulting amount of data, the module CNTRL allows to print time series only for each *n*th timestep ($n \geq 1$), see chapter 13.3.2.

Time series files, which can require a large amount of disk space, may also be generated in binary format. Whether the time series output uses text or binary format is controlled by the CNTRL module (see chapter 13).

Tab. 16.1 Element-specific parameters in `FOFT`

No	Module (0 printed if inactive)	Parameter	Description	Unit
1		P_{gas}	gas pressure	[Pa]
2		$P_{\text{liq}} = P_{\text{gas}} + P_{\text{cap}}$	liquid pressure	[Pa]
3		S_{liq}	gas saturation	[1]
4		S_{gas}	liquid saturation	[1]
5		X_{brine}	brine concentration	[1]
6		T	temperature	[°C]
7		$-P_{\text{cap}}$	capillary pressure (in TOUGH2 P_{cap} is negative thus the output is positive)	[Pa]
8		$P_{\text{pot}} = P_{\text{liq}} - zg\rho_{\text{liq}}$	potential pressure (with constatt density this is equivalent to a potential)	[Pa]
9		ρ_{liq}	density of the gas phase	[kg/m^3]
10		ρ_{gas}	density of the liquid phase	[kg/m^3]
11		μ_{gas}	viscosity of the gas phase	[Pa s]
12		μ_{liq}	viscosity of the liquid phase	[Pa s]
13		k	intrinsic permeability (for ISO=1 multiplied by the area reduction factor $V_{\text{cavity}}/V_{\text{tough}}$)	[m^2]
14		V_{tough}	volume of the TOUGH grid element	[m^3]
15	COMP otherwise 1	$V_{\text{cavity}}/V_{\text{tough}}$	area reduction factor $V_{\text{cavity}}/V_{\text{tough}}$	[1]
16	COMP otherwise ϕ_{tough}	ϕ_{phys}	porosity	[1]
17		ϕ_{tough}	internal porosity of TOUGH	[1]
18	FISS, otherwise 0	ϕ_{fiss}	porosity of a fissure	[1]
19	FISS	p_{thr}	pressure threshold	[Pa]
20	COMP	K	rate of convergency	[1/s]
21	CORRO	g	grade of corrosion (weighted average over sources)	[1]
22	CORRO	Q	rate of gas generation	[mol/s]
23	CORRO	m_{can}	canister water mass	[kg]
24	CORFL	g	grade of corrosion	[1]
25	EOS7R	$X_{\text{RN1}}^{\text{gas}}$	mass fraction of RN 1 in the gas phase	[1]
26	EOS7R	$X_{\text{RN1}}^{\text{liq}}$	mass fraction of RN 1 in the liquid phase	[1]
27	EOS7R	$X_{\text{RN2}}^{\text{gas}}$	mass fraction of RN 2 in the gas phase	[1]
28	EOS7R	$X_{\text{RN2}}^{\text{liq}}$	mass fraction of RN 2 in the liquid phase	[1]
29		$m_{H_2O}^{\text{gas}}$	mass of water in the gas phase	[kg]

30		$m^{\mathrm{liq}}_{H_2O}$	mass of water in the liquid phase	[kg]
31		$m^{\mathrm{gas}}_{\mathrm{brine}}$	mass of brine in the gas phase	[kg]
32		$m^{\mathrm{liq}}_{\mathrm{brine}}$	mass of brine in the liquid phase	[kg]
33		$m^{\mathrm{gas}}_{\mathrm{air}}$	mass of air in the gas phase	[kg]
34		$m^{\mathrm{liq}}_{\mathrm{air}}$	mass of air in the liquid phase	[kg]
35	EOS7R	$m^{\mathrm{gas}}_{\mathrm{RN1}}$	mass of RN1 in the gas phase	[kg]
36	EOS7R	m^{liq}_{RN1}	mass of RN1 in the liquid phase	[kg]
37	EOS7R	$m^{\mathrm{gas}}_{\mathrm{RN2}}$	mass of RN2 in the gas phase	[kg]
38	EOS7R	m^{liq}_{RN2}	mass of RN2 in the liquid phase	[kg]
39	EOS7R	m^{ads}_{RN1}	adsorbed mass of RN1	[kg]
40	EOS7R	m^{ads}_{RN2}	adsorbed mass of RN2	[kg]
The following lines only appear if the RN module is active, for each radionuclide i.				
41+6*(i-1)		$m^{\mathrm{tot}}_{\mathrm{RN\ i}}$	mass of RN i (0 for inactive elements)	[kg]
42+6*(i-1)		$X^{\mathrm{gas}}_{\mathrm{RN\ i}}$	mobile concentration of. RN i in the gas phase	[1]
43+6*(i-1)		$X^{\mathrm{liq}}_{\mathrm{RN\ i}}$	mobile concentration of. RN i in the liquid phase	[1]
44+6*(i-1)		$m^{\mathrm{gas}}_{\mathrm{RN\ i}}$	mobile mass of RN i in the gas phase(0 for inactive elements)	[kg]
45+6*(i-1)		$m^{\mathrm{liq}}_{\mathrm{RN\ i}}$	mobile mass of RN i in the liquid phase(0 for inactive elements)	[kg]
46+6*(i-1)		$m^{\mathrm{ads}}_{\mathrm{RN\ i}}$	adsorbed mass of RN I (0 for inactive elements)	[kg]

Tab. 16.2 Connection-specific parameters in `COFT`

The gas part of the diffusive flux is set to 0 for `MOPR(24)=0` (default). The fluid component then contains the full multiphase diffusive flux. For `MOPR(24)=1` the diffusive flow is calculated for each phase separately.

No (KDATA ≠ 4)	No (KDATA= 4)	Module (0 printed if inactive)	Parameter	Description	Unit
1	1		Q^{gas}	mass flow in the gas phase	[kg/s]
2	2		Q^{liq}	mass flow in the liquid phase	[kg/s]
3	3		Q^{heat}	heat flow	[J/s]
	4	EOS7R	$Q^{\mathrm{adv,gas}}_{\mathrm{RN1}}$	advective flow of RN1 (gas phase)	[kg/s]
	5	EOS7R	$Q^{\mathrm{adv,liq}}_{\mathrm{RN1}}$	advective flow of RN1 (liquid phase)	[kg/s]
	6	EOS7R	$Q^{\mathrm{adv,tot}}_{\mathrm{RN1}}$	advective flow of RN1	[kg/s]
	7	EOS7R	$Q^{\mathrm{adv,gas}}_{\mathrm{RN2}}$	advective flow of RN2 (gas phase)	[kg/s]
	8	EOS7R	$Q^{\mathrm{adv,liq}}_{\mathrm{RN2}}$	advective flow of RN2 (liquid phase)	[kg/s]
	9	EOS7R	$Q^{\mathrm{adv,tot}}_{\mathrm{RN2}}$	advective flow of RN	[kg/s]
	10	EOS7R	$Q^{\mathrm{dif,gas}}_{\mathrm{RN1}}$	diffusive flow of RN1 (gas phase)	[kg/s]
	11	EOS7R	$Q^{\mathrm{dif,gas}}_{\mathrm{RN2}}$	diffusive flow of RN2 (gas phase)	[kg/s]

	12	EOS7R	$Q_{\mathrm{RN1}}^{\mathrm{dif,gas}}$	diffusive flow of RN1 (gas phase)	[kg/s]
	13	EOS7R	$Q_{\mathrm{RN2}}^{\mathrm{dif,gas}}$	diffusive flow of RN2 (gas phase)	[kg/s]
4	14	EOS7R	$Q_{\mathrm{RN1}}^{\mathrm{tot}}$	total mass flow RN1	[kg/s]
5	15	EOS7R	$Q_{\mathrm{RN2}}^{\mathrm{tot}}$	total mass flow RN2	[kg/s]
	The following limes appear only if the RN module is active, for each radionuclide i, respectively.				
5+i	16+5*(i-1)		$Q_{\mathrm{RN\,i}}^{\mathrm{tot}}$	total mass flow RN i	[kg/s]
	17+5*(i-1)		$Q_{\mathrm{RN\,i}}^{\mathrm{adv,gas}}$	advective flow of RN i (gas phase)	[kg/s]
	18+5*(i-1)		$Q_{\mathrm{RN\,i}}^{\mathrm{adv,liq}}$	advective flow of RN i (liquid phase)	[kg/s]
	19+5*(i-1)		$Q_{\mathrm{RN\,i}}^{\mathrm{dif,gas}}$	diffusive flow of RN i (gas phase)	[kg/s]
	20+5*(i-1)		$Q_{\mathrm{RN\,i}}^{\mathrm{dif,liq}}$	diffusive flow of RN i (liquid phase)	[kg/s]

Tab. 16.3 Sink- and source-specific parameters in `GOFT`.

No	Parameter	Description	Unit
1	N	Component-generated	[1]
2	$\dot{m}$	mass generation rate	[kg/s]
3	$\dot{Q}$	heat generation rate	[J/s]
4	$\int_o^t \dot{m}\,dt$	cumulative mass at time t	[kg]
5	$\int_o^t \dot{Q}\,dt$	cumulative heat at time t	[J]

Tab. 16.4 Element-specific output parameters of TOUGH2-GRS

All parameters are printed at user-defined printout times for all elements to STDOUT (without the parameters specific for the RN module) and line by line for each printout time to the file `ELE_MAIN`. The format of line in `ELE_MAIN` is `time step number, time, [element index,[variables, …], …].`

No	Module (0 printed if inactive)	Parameter	Description	Unit
1		P_{gas}	gas pressure	[Pa]
2		$P_{\mathrm{liq}} = P_{\mathrm{gas}} + P_{\mathrm{cap}}$	liquid pressure	[Pa]
3		S_{liq}	gas saturation	[1]
4		S_{gas}	liquid saturation	[1]
5		X_{brine}	brine concentration	[1]
6		T	temperature	[°C]
7		$-P_{\mathrm{cap}}$	capillary pressure (in TOUGH2 P_{cap} is negative thus the output is positive)	[Pa]
8		$P_{\mathrm{pot}} = P_{\mathrm{liq}} - zg\rho_{\mathrm{liq}}$	potential pressure (with constatt density this is equivalent to a potential)	[Pa]
9		ρ_{liq}	density of the gas phase	[kg/m³]
10		ρ_{gas}	density of the liquid phase	[kg/m³]
11		μ_{gas}	viscosity of the gas phase	[Pa s]

12		μ_{liq}	viscosity of the liquid phase	[Pa s]
13		k	intrinsic permeability (for ISO=1 multiplied by the area reduction factor $V_{\text{cavity}}/V_{\text{tough}}$)	[m²]
14		V_{tough}	volume of the TOUGH grid element	[m³]
15	COMP otherwise 1	$V_{\text{cavity}}/V_{\text{tough}}$	area reduction factor $V_{\text{cavity}}/V_{\text{tough}}$	[1]
16	COMP otherwise ϕ_{tough}	ϕ_{phys}	porosity	[1]
17		ϕ_{tough}	internal porosity of TOUGH	[1]
18	FISS	ϕ_{fiss}	porosity of a fissure	[1]
19	FISS	p_{thr}	pressure Threshold	[Pa]
20	COMP	K	rate of convergency	[1/s]
21	CORRO	g	grade of corrosion (weighted average over sources)	[1]
22	CORRO	Q	rate of gas generation	[mol/s]
23	CORRO	m_{can}	canister water mass	[kg]
24	CORFL	g	grade of corrosion	[1]
25	EOS7R	$X_{\text{RN1}}^{\text{gas}}$	mass fraction of RN 1 in the gas phase	[1]
26	EOS7R	$X_{\text{RN1}}^{\text{liq}}$	mass fraction of RN 1 in the liquid phase	[1]
27	EOS7R	$X_{\text{RN2}}^{\text{gas}}$	mass fraction of RN 2 in the gas phase	[1]
28	EOS7R	$X_{\text{RN2}}^{\text{liq}}$	mass fraction of RN 2 in the liquid phase	[1]
29	for diagnostics of time-stepping problems	MaxRes	number of maximal residua since last print out	[1]
30		phas	Condition of aggregation (1: gas phase only; 2: two-phase; 3: liquid phase only)	[1]
31		$\frac{dP_{\text{cap}}}{dS_{\text{liq}}}$	dPcap/dSliq (for elements with two-phase condition only; always positive)	[Pa]
The following limes appear only if the RN module is active, for each radionuclide i, respectively.				
32+6*(i-1)		$m_{\text{RN i}}^{\text{tot}}$	mass of RN i (0 for inactive elements)	[kg]
33+6*(i-1)		$X_{\text{RN i}}^{\text{gas}}$	mobile concentration of. RN i in the gas phase	[1]
34+6*(i-1)		$X_{\text{RN i}}^{\text{liq}}$	mobile concentration of. RN i in the liquid phase	[1]
35+6*(i-1)		$m_{\text{RN i}}^{\text{gas}}$	mobile mass of RN i in the gas phase(0 for inactive elements)	[kg]
36+6*(i-1)		$m_{\text{RN i}}^{\text{liq}}$	mobile mass of RN i in the liquid phase(0 for inactive elements)	[kg]
37+6*(i-1)		$m_{\text{RN i}}^{\text{ads}}$	adsorped mass of RN I (0 for inactive elements)	[kg]

Tab. 16.5 Connection-specific parameters in TOUGH2-GRS

All parameters are printed at user-defined printout times for all connections line by line for each printout time to the file `CON_MAIN`. The format of line in `CON_MAIN` is `time step number, time, [element index,[variables, …], …]`. The gas part of the diffusive flux is set to 0 for `MOPR(24)=0` (default). The fluid component then contains the full multiphase diffusive flux. For `MOPR(24)=1` the diffusive flow is calculated for each phase separately. The flow is always positive if the direction of flow is from the second element of the connection to the first one.

No	Module (0 printed if inactive)	Parameter	Description	Unit
1		Q^{heat}	heat flow	[J/s]
2		E_{s}	specific energy	[J/kg]
3		Q^{tot}	full mass flow	[kg/s]
4		Q^{gas}	mass flow in the gas phase	[kg/s]
5		Q^{liq}	mass flow in the liquid phase	[kg/s]
6		$Q^{\mathrm{liq}}_{\mathrm{brine}}$	full mass flow	[kg/s]
7		v_{gas}	pore velocity in the gas phase	[m/s]
8		v_{liq}	pore velocity in the liquid phase	[m/s]
9	EOS7R	$Q^{\mathrm{adv,gas}}_{\mathrm{RN1}}$	advective flow of RN1 (gas phase)	[kg/s]
10	EOS7R	$Q^{\mathrm{adv,gas}}_{\mathrm{RN2}}$	advective flow of RN2 (gas phase)	[kg/s]
11	EOS7R	$Q^{\mathrm{adv,liq}}_{\mathrm{RN1}}$	advective flow of RN1 (liquid phase)	[kg/s]
12	EOS7R	$Q^{\mathrm{adv,liq}}_{\mathrm{RN2}}$	advective flow of RN2 (liquid phase)	[kg/s]
13		$Q^{\mathrm{dif,gas}}_{H_2O}$	diffusive flow of water (gas phase)	[kg/s]
14		$Q^{\mathrm{dif,gas}}_{\mathrm{brine}}$	diffusive flow of brine (gas phase)	[kg/s]
15		$Q^{\mathrm{dif,gas}}_{\mathrm{air}}$	diffusive flow of air (gas phase)	[kg/s]
16	EOS7R	$Q^{\mathrm{dif,gas}}_{\mathrm{RN1}}$	diffusive flow of RN1 (gas phase)	[kg/s]
17	EOS7R	$Q^{\mathrm{dif,gas}}_{\mathrm{RN2}}$	diffusive flow of RN2 (gas phase)	[kg/s]
18		$Q^{\mathrm{dif,liq}}_{H_2O}$	diffusive flow of water (liquid phase)	[kg/s]
19		$Q^{\mathrm{dif,liq}}_{\mathrm{brine}}$	diffusive flow of brine (liquid phase)	[kg/s]
20		$Q^{\mathrm{dif,liq}}_{\mathrm{air}}$	diffusive flow of air (liquid phase)	[kg/s]
21	EOS7R	$Q^{\mathrm{dif,liq}}_{\mathrm{RN1}}$	diffusive flow of RN1 (liquid phase)	[kg/s]
22	EOS7R	$Q^{\mathrm{dif,liq}}_{\mathrm{RN2}}$	diffusive flow of RN2 (liquid phase)	[kg/s]
	The following limes appear only if the RN module is active, for each radionuclide i, respectively.			
23+5*(i-1)		$Q^{\mathrm{tot}}_{\mathrm{RN\,i}}$	mass flow of RN i	[kg/s]
24+5*(i-1)		$Q^{\mathrm{adv,gas}}_{\mathrm{RN\,i}}$	advective flow of RN i (gas phase)	[kg/s]
25+5*(i-1)		$Q^{\mathrm{dif,gas}}_{\mathrm{RN\,i}}$	diffusive flow of RN i (gas phase)	[kg/s]
26+5*(i-1)		$Q^{\mathrm{adv,liq}}_{\mathrm{RN\,i}}$	advective flow of RN i (liquid phase)	[kg/s]
27+5*(i-1)		$Q^{\mathrm{dif,liq}}_{\mathrm{RN\,i}}$	diffusive flow of RN i (liquid phase)	[kg/s]

Tab. 16.6 Input format of data block DOFT

Line	Column	Content	Format	Unit	Variable
1	1-5	Keyword `"DOFT"`	A5		
2	1-5	Material	A5	[1]	`edoft`

Repeat line 2 for all required materials. Material 1 will be the second material index in the DOFT file, because the first index is reserved for the entire model domain.

Tab. 16.7 Material-specific parameters in `DOFT`

For each time step the first entry corresponds to the full TOUGH grid (cumulative quantities) and the *i*th entry corresponds to material number *i*-1.

No (`KDATA ≠ 4`)	**No** (`KDATA= 4`)	**Module** (if not active, set to 0)	**Parameter**	**Description**	**Unit**
1	1	CORRO	N^{gas}	amount of gas produced by corrosion	[mol]
2	2	CORRO	m_{can}	canister water mass	[kg]
3	3	PRLIM	V_{esc}	escaped gas (cumulative only)	[m^3]
4	4		$P_{\mathrm{gas,max}}$	maximal gas pressure	[Pa]
5	5		$N_{\mathrm{gas,max}}$	material index where $P_{\mathrm{gas,max}}$ occurs	
	6	EOS7R	m_{RN1}/V_i	mass of RN1 per volume of material i	[kg/m^3]
	7	EOS7R	m_{RN2}/V_i	mass of RN2 per volume of material i	[kg/m^3]
	8	EOS7R	$R_{\mathrm{max,RN1}}$	maximal residuum of RN1	
	9	EOS7R	$R_{\mathrm{max,RN2}}$	maximal residuum of RN2	
	10	EOS7R	$\rho_{\mathrm{max,RN1}}$	density of RN1 for material with maximal residuum	[kg/m^3]
	11	EOS7R	$\rho_{\mathrm{max,RN2}}$	density of RN2 for material with maximal residuum	[kg/m^3]
	12 – 15			*obsolete*	
6	16		$m_{H_2O}^{\mathrm{gas}}$	mass of water in the gas phase	[kg]
7	17		$m_{H_2O}^{\mathrm{liq}}$	mass of water in the liquid phase	[kg]
8	18		$m_{\mathrm{brine}}^{\mathrm{gas}}$	mass of brine in the gas phase	[kg]
9	19		$m_{\mathrm{brine}}^{\mathrm{liq}}$	mass of brine in the liquid phase	[kg]

10	20		$m_{\mathrm{air}}^{\mathrm{gas}}$	mass of air in the gas phase	[kg]
11	21		$m_{\mathrm{air}}^{\mathrm{liq}}$	mass of air in the liquid phase	[kg]
12	22	EOS7R	$m_{\mathrm{RN1}}^{\mathrm{gas}}$	mass of RN1 in the gas phase	[kg]
13	23	EOS7R	m_{RN1}^{liq}	mass of RN1 in the liquid phase	[kg]
14	24	EOS7R	$m_{\mathrm{RN2}}^{\mathrm{gas}}$	mass of RN2 in the gas phase	[kg]
15	25	EOS7R	m_{RN2}^{liq}	mass of RN2 in the liquid phase	[kg]
	The following limes appear only if the RN module is active, for each radionuclide i, respectively.				
15+i	26+3*(i-1)		$m_{\mathrm{RN\ i}}^{\mathrm{tot}}$	mass of RN i (0 for inactive elements)	[kg]
	27+3*(i-1)		$m_{\mathrm{RN\ i}}^{\mathrm{gas}}$	mobile mass of RN i in the gas phase	[kg]
	28+3*(i-1)		$m_{\mathrm{RN\ i}}^{\mathrm{liq}}$	mobile mass of RN i in the liquid phase	[kg]

References

The GRS-A Reports listed below were prepared as part of research projects that were sponsored by the BMUB (formerly BMU). Quoting from these reports, reproducing them in whole or in part or making them accessible to third parties therefore requires the prior consent of the BMUB.

/BFS 09/ Bundesamt für Strahlenschutz (BfS): Sicherheitsanalyse für das verfüllte und verschlossene Endlager mit dem Programmpaket EMOS. BfS-KZL: 9M/23210051/EG/RB/0056/00, 27. March 2009.

/FIS 01/ Fischer-Appelt, K., Fischer, H., Javeri, V., Martens, K.-H., Röhlig, K.-J., Eugen, S.: Implementierung von nichtlinearen Sorptionsansätzen in die Transportrechenprogramme MARNIE, NAMMU und TOUGH2 und deren Überprüfung. GRS-A-2897, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH, 1. January 2001.

/GEN 80/ van Genuchten, M.T.: A Closed-form Equation for Predicting the Hydraulic Conductivity of Unsaturated Soils. Soil Science Society of America Journal, Vol. 44, No. 5, pp. 892–898, DOI 10.2136/sssaj1980.03615995004400050002x, 1980.

/GRS 13/ Software Management Group: Maßbahmen zur Qualitätssicherung bei der Erstellung von Computerprogrammen in der GRS (QM-richtlinien Programmentwicklung). QM-Handbuch, Teil 3: FA 03 "Fachliche Qualitätssicherung von Arbeitsergebnissen", Anlage IV. 8 pp., Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH: Köln, 21. November 2013.

/HOT 16a/ Hotzel, S., Eckel, J., Fischer, H., Navarro, M.: Test Handbook for the Code TOUGH2-GRS. GRS-402, ISBN 978-3-944161-83-9, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) gGmbH, 2016.

/HOT 16b/ Hotzel, S., Navarro, M., Seher, H.: QS-Handbuch für den Programmcode TOUGH2-GRS. GRS-401, ISBN 978-3-944161-82-2, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) gGmbH, 2016.

/JAV 01/ Javeri, V.: Dreidimensionale Analysen zum Nuklidtransport bei salzanteilabhängiger Adsorption in einem porösen Medium mit dem Rechenprogramm TOUGH2. Ed.: Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH, GRS-A-2864, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH: Köln, 2001.

/JAV 02/ Javeri, V.: Analysen zum Nuklidtransport bei variabler Salinität und nichtlinearer Adsorption in der stark heterogenen Geosphäre der Gorlebener Rinne. Ed.: Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH, GRS-A-3038, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH: Köln, 2002.

/JAV 95/ Javeri, V.: Orientierende Analysen zum Nuklidtransport durch Naturkonvektion; Gesteinskonvergenz und Dispersion in porösen Medien mit dem Rechenprogramm TOUGH2. Ed.: Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH, GRS-A-2240: Köln, 1995.

/KOC 12/ Kock, I., Eickemeier, R., Frieling, G., Heusermann, S., Knauth, M., Minkley, W., Navarro, M., Nipp, H.-K., Vogel, P.: Integritätsanalyse der geologischen Barriere. Bericht zum Arbeitspaket 9.1, Vorläufige Sicherheitsanalyse für den Standort Gorleben. GRS-286, ISBN 978-3-939355-62-5, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH: Köln, 2012.

/KRÖ 09/ Kröhn, K.-P., Stührenberg, D., Herklotz, M., Heemann, U., Lerch, C., Xie, M.: Restporosität und -permeabilität von kompaktierendem Salzgrus-Versatz in einem HAW-Endlager - Phase 1. GRS-254, 266 pp.ISBN 978-3-939355-29-8, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH, 2009.

/LAR 13/ Larue, J., Baltes, B., Fischer, H., Frieling, G., Kock, I., Navarro, M., Seher, H.: Radiologische Konsequenzenanalyse. Bericht zum Arbeitspaket 10, Vorläufige Sicherheitsanalyse für den Standort Gorleben. GRS-289, ISBN 978-3-939355-65-6, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH: Köln, 2013.

/LBNL 16a/ Lawrence Berkeley National Laboratory (LBNL): TOUGH Issues & Solutions. as at 8 October 2015, Earth Sciences Division, cited on 4 May 2016.

/LBNL 16b/ Lawrence Berkeley National Laboratory (LBNL): TOUGH2 V2.0: Bugs and Fixes. Lawrence Berkeley National Laboratory (LBNL), cited on 9 February 2016.

/LEV 41/ Leverett, M.C.: Capillary behaviour in porous solids. Transactions of the AIME, Vol. 142, pp. 159–172, 1941.

/MAR 02/ Martens, K.-H., Fischer, H., Romstedt, P.: Beschreibung des Rechenprogrammes MARNIE. GRS-A-3027, 135 pp., Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH: Köln, 1. January 2002.

/NAV 08/ Navarro, M., Baltes, B., Beuth, T., Bracke, G., Fischer, H., Fischer-Appelt, K., Hotzel, S., Javeri, V., Kindt, A., Lambers, L., Larue, J., McStocker, B., Oppermann, U., Schrödl, E.: Verfolgung und Bewertung der Fortentwicklung des Standes von Wissenschaft und Technik beim Nachweis der Langzeitsicherheit von Endlagern. Abschlussbericht zum Vorhaben SR 2548. GRS-A-3418, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH: Köln, 2008.

/NAV 09/ Navarro, M.: Simulating the migration of repository gases through argillaceous rock by implementing the mechanism of pathway dilation into the code TOUGH2 (TOUGH2-PD). PAMINA project, public milestone 3.2.14. Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH, 2009.

/NAV 13a/ Navarro, M.: Die vereinfachte Berechnung der Konvergenzrate salzgrusverfüllter Hohlräume im Steinsalz. GRS-307, ISBN 978-3-939355-86-1, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH: Köln, 2013.

/NAV 13b/ Navarro, M.: Handbuch zum Code TOUGH2-GRS.00a. Erweiterungen des Codes TOUGH2 zur Simulation von Strömungs- und Transportprozessen in Endlagern. GRS-310, ISBN 978-3-939355-89-2, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH: Köln, 2013.

/NAV 13c/ Navarro, M.: Modelling Gas and Water Flow Through Dilating Pathways in Opalinus Clay. The HG-C and HG-D Experiments, A study within the Euratom 7th Framework Programme Project FORGE. GRS-306, ISBN 978-3-

939355-85-4, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH: Cologne, 2013.

/NAV 14/ Navarro, M.: Das EOS-Modul SALTREP für den Code TOUGH2-GRS. GRS-321, 65 pp.ISBN 978-3-944161-01-3, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH, 2014.

/NAV 15/ Navarro, M.: Speeding up the iterative solvers of the T2SOLV solver package. A presentation at the TOUGH-Symposium 2015 at the Lawrence Bereley National Laboratories. as at 2015, cited on 04.05.16.

/NAV 16/ Navarro, M., Fischer, H., Seher, H., Weyand, T.: Ansätze zur Simulation der Zweiphasenströmung in salinaren Endlagern mit dem Code TOUGH2-GRS. Bericht im Vorhaben ZIESEL (Zweiphasenfluss in einem salinaren Endlager am Beispiel des ERAM). GRS-398, ISBN 978-3-944161-79-2, 2016.

/NOS 05/ Noseck, U., Brewitz, W., Becker, D.-A., Buhmann, D., Fahrenholz, C., Fein, E., Hirsekorn, R.-P., Keesmann, S., Kröhn, K.-P., Müller-Lyda, I., Rübel, A., Schneider, A., Storck, R.: Wissenschaftliche Grundlagen zum Nachweis der Langzeitsicherheit von Endlagern. GRS-204, 205 pp.ISBN 3-931995-71-2, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) mbH: Köln, Garching b. München, Berlin, Braunschweig, 2005.

/PRU 99/ Pruess, K., Oldenburg, C., Moridis, G.: TOUGH2 User's Guide, Version 2.0. LBNL-43134, 198 pp., Lawrence Berkeley National Laboratory (LBNL): Berkeley, California, USA, 1. November 1999, revised September 2012.

/SEH 16/ Seher, H., Navarro, M.: SITA, version 0.1.a. A simulation and code testing assistant for TOUGH2 and MARNIE. GRS-400, ISBN 978-3-944161-81-5, Gesellschaft für Anlagen- und Reaktorsicherheit (GRS) gGmbH: Köln, 2016.

/STE 85/ Stelte, N.: Analytische Approximation der Konvergenzrate salzgrusversetzter und unter hydraulischem Druck stehender Hohlräume im Salzgestein. In: Einzeluntersuchungen zur Radionuklidfreisetzung aus einem Modellsalzstock. Fachband 15, Ed.: Projekt Sicherheitsstudien Entsorgung, pp. 108–141, TU Berlin: Berlin, 1985.

/STO 80/ Stoer, J., Bulirsch, R.: Introduction to Numerical Analysis. 1 online resource (ix, 609ISBN 978-1-4757-5594-7, Springer New York: New York, NY, 1980.

List of Figures

List of Tables

Gesellschaft für Anlagen-
und Reaktorsicherheit
(GRS) gGmbH

Schwertnergasse 1
50667 Köln
Telefon +49 221 2068-0
Telefax +49 221 2068-888

Forschungszentrum
85748 Garching b. München
Telefon +49 89 32004-0
Telefax +49 89 32004-300

Kurfürstendamm 200
10719 Berlin
Telefon +49 30 88589-0
Telefax +49 30 88589-111

Theodor-Heuss-Straße 4
38122 Braunschweig
Telefon +49 531 8012-0
Telefax +49 531 8012-200

www.grs.de

ISBN 978-3-944161-84-6