

HEATSTORE

Interim report on UTES-type/site-specific simulators based on academic/research codes

Prepared by: Sigrún Tómasdóttir & Gunnar Gunnarsson (ed.) (Reykjavik Energy)
Julian Mindel, Thomas Driesner, Daniel Traver Birdsell, Martin O. Saar (ETHZ - Eidgenössische Technische Hochschule Zürich)
Maarten Saaltink & Sebastia Olivella (UPC - Universitat Politècnica de Catalunya)
Stijn Beernink, Martin Bloemendal, Niels Hartog (KWR Watercycle Research Institute)

Checked by: Sigrún Tómasdóttir, Gunnar Gunnarsson (Reykjavik Energy)
Thomas Driesner (ETHZ - Eidgenössische Technische Hochschule Zürich), WP2 Leader

Approved by: Holger Cremer (TNO), HEATSTORE coordinator

Please cite this report as: Tómasdóttir, S. & Gunnarsson, G. (ed) 2019: HEATSTORE – Interim report on UTES-type/site-specific simulators based on academic/research codes. GEOthermica – ERA NET Cofund Geothermal. 55 pp.

This report represents HEATSTORE project deliverable number D2.2.

Summary

This report documents the fundamental details of the academic/research simulators used within work package 2 of the HEATSTORE project. The primary focus of this report is to complement deliverable 2.1 with the more technical aspects of the academic/research simulators used. An updated version of this report is due in Month 35.

The simulators are used to model prospective operations at various sites including the High Temperature - Aquifer Thermal Energy Storage (HT-ATES) sites at Geneva and Bern-Forsthaus in Switzerland as well as Koppert-Cress in the Netherlands. They are also able to model different geothermal conditions, particularly through the modeling of a high enthalpy reservoir in the Azores and scenario modeling for superhigh-temperature resource development in the Hengill area in Iceland.

The academic/research codes discussed in separate chapters within this report are CSMP++/Nexus, MOOSE, Code_Bright, Retraso and SEAWAT. For each simulator the following characteristics are discussed: assumptions and governing equations, numerical discretization method, basic solution algorithm and logic, pre- and post-processing, the sites the code is used for and IP and code availability. An appendix with summarized information on input files for the simulators is provided at the end.



HEATSTORE (170153-4401) is one of nine projects under the GEOthermica – ERA NET Cofund aimed at accelerating the uptake of geothermal energy by 1) advancing and integrating different types of underground thermal energy storage (UTES) in the energy system, 2) providing a means to maximise geothermal heat production and optimise the business case of geothermal heat production doublets, 3) addressing technical, economic, environmental, regulatory and policy aspects that are necessary to support efficient and cost-effective deployment of UTES technologies in Europe.

This project has been subsidized through the ERANET cofund GEOthermica (Project n. 731117), from the European Commission, RVO (the Netherlands), DETEC (Switzerland), FZJ-PtJ (Germany), ADEME (France), EUDP (Denmark), Rannis (Iceland), VEA (Belgium), FRCT (Portugal), and MINECO (Spain).



About HEATSTORE

High Temperature Underground Thermal Energy Storage

The heating and cooling sector is vitally important for the transition to a low-carbon and sustainable energy system. Heating and cooling is responsible for half of all consumed final energy in Europe. The vast majority – 85% - of the demand is fulfilled by fossil fuels, most notably natural gas. Low carbon heat sources (e.g. geothermal, biomass, solar and waste-heat) need to be deployed and heat storage plays a pivotal role in this development. Storage provides the flexibility to manage the variations in supply and demand of heat at different scales, but especially the seasonal dips and peaks in heat demand. Underground Thermal Energy Storage (UTES) technologies need to be further developed and need to become an integral component in the future energy system infrastructure to meet variations in both the availability and demand of energy.

The main objectives of the HEATSTORE project are to lower the cost, reduce risks, improve the performance of high temperature (~25°C to ~90°C) underground thermal energy storage (HT-UTES) technologies and to optimize heat network demand side management (DSM). This is primarily achieved by 6 new demonstration pilots and 8 case studies of existing systems with distinct configurations of heat sources, heat storage and heat utilization. This will advance the commercial viability of HT-UTES technologies and, through an optimized balance between supply, transport, storage and demand, enable that geothermal energy production can reach its maximum deployment potential in the European energy transition.

Furthermore, HEATSTORE also learns from existing UTES facilities and geothermal pilot sites from which the design, operating and monitoring information will be made available to the project by consortium partners.

HEATSTORE is one of nine projects under the GEOTHERMICA – ERA NET Cofund and has the objective of accelerating the uptake of geothermal energy by 1) advancing and integrating different types of underground thermal energy storage (UTES) in the energy system, 2) providing a means to maximize geothermal heat production and optimize the business case of geothermal heat production doublets, 3) addressing technical, economic, environmental, regulatory and policy aspects that are necessary to support efficient and cost-effective deployment of UTES technologies in Europe. The three-year project will stimulate a fast-track market uptake in Europe, promoting development from demonstration phase to commercial deployment within 2 to 5 years, and provide an outlook for utilization potential towards 2030 and 2050.

The 23 contributing partners from 9 countries in HEATSTORE have complementary expertise and roles. The consortium is composed of a mix of scientific research institutes and private companies. The industrial participation is considered a very strong and relevant advantage which is instrumental for success. The combination of leading European research institutes together with small, medium and large industrial enterprises, will ensure that the tested technologies can be brought to market and valorised by the relevant stakeholders.

Document Change Record

This section shows the historical versions, with a short description of the updates.

Version	Short description of change
2019.09.09	Report template sent out to contributors
2019.10.11	Contributions handed in for review
2019.10.18	First Version, reviewed contributions compiled by Sigrún Tómasdóttir
2019.11.07	Last contribution received
2019.11.08	All chapters formatted to attain consistency in overall look
2019.11.14	Final Editing according to inputs of contributors
2019.12.04	Final version

Table of Content

About HEATSTORE	4
1 Introduction.....	8
2 CSMP++/Nexus	9
2.1 Descriptive summary	9
2.2 Assumptions and Governing equations	9
2.3 Numerical discretization method.....	10
2.3.1 Initial and Boundary conditions.....	11
2.3.2 Material discontinuities	12
2.3.3 Wells	13
2.4 Basic Solution algorithm and logic	13
2.5 Pre-processing	14
2.6 Post processing.....	15
2.7 Swiss pilot site Geneva.....	16
2.8 Azores study.....	16
2.9 Icelandic case study	17
2.10 IP and code availability.....	19
2.11 References.....	19
3 MOOSE	20
3.1 Descriptive summary	20
3.2 Assumptions and Governing equations	20
3.3 Numerical discretization method.....	22
3.3.1 Initial and Boundary conditions.....	22
3.3.2 Material discontinuities	23
3.3.3 Wells	23
3.4 Basic Solution algorithm and logic	24
3.5 Pre-processing	25
3.6 Postprocessing.....	25
3.7 Swiss pilot site Geneva.....	25
3.8 Swiss pilot site Bern Forsthaus.....	26
3.9 IP and code availability	26
3.10 References.....	26
4 CODE_BRIGHT and RETRASO	28
4.1 Descriptive summary	28
4.2 Assumptions and Governing equations	28

4.3	Numerical discretization method.....	29
4.3.1	Initial and Boundary conditions.....	30
4.3.2	Material discontinuities.....	31
4.3.3	Wells.....	31
4.4	Basic Solution algorithm and logic.....	31
4.5	Pre- and postprocessing.....	31
4.6	Swiss pilot site Bern Forsthaus.....	31
4.7	IP and code availability.....	31
4.8	References.....	32
5	SEAWAT simulator.....	33
5.1	Descriptive summary.....	33
5.2	Assumptions and Governing equations.....	33
5.3	Numerical discretization method.....	34
5.3.1	Initial and Boundary conditions.....	37
5.3.2	Material discontinuities.....	38
5.3.3	Wells.....	38
5.4	Basic Solution algorithm and logic.....	38
5.5	Preprocessing.....	38
5.6	Postprocessing.....	39
5.7	Dutch pilot site Koppert Cress.....	40
5.8	IP and code availability.....	40
5.9	References.....	40
	Appendix A: input files.....	42
	CSMP++/Nexus.....	42
	CSMP++ regions file.....	42
	ICEMCFD file example.....	42
	CSMP++ configuration file.....	43
	Nexus control file.....	45
	Moose.....	49
	CODE_BRIGHT and RETRASO.....	53
	SEAWAT.....	55

1 Introduction

Some of the HEATSTORE projects encounter complex geological situations which conventional/commercial modelling software cannot accurately simulate. Other projects require combinations of different types of modelling, such as combinations of flow and heat transfer modelling with rock mechanics such as poro- and thermo-elastic effects. To assist in dealing with these challenges, and due to the lack of capabilities in commercial simulators, academic/research codes were used, which have not yet been routinely applied in commercial geothermal projects. The academic/research codes used within HEATSTORE are CSMP++/Nexus, MOOSE, Code_Bright, Retraso and SEAWAT. These codes have been introduced and described briefly in deliverable 2.1 (Driesner, 2019) which covered site descriptions, conceptualization and modelling approaches. This deliverable relates to task 2.2 and is meant to complement deliverable 2.1 with the more technical aspects of each academic simulator used in work package 2 within HEATSTORE. This is the interim report for task 2.2. An updated report is due in month 35.

The codes covered in this report are used to simulate a wide range in geological conditions. The Geneva site represents a geologically challenging site with major thrust and strike-slip faulting intersecting potential aquifers which are inclined themselves due to large scale Alpine deformation. The aim is to seasonally store up to 50 GWh/a of waste heat at the site. Both CSMP++/Nexus and Moose are used to model conditions at the Geneva site. The Bern-Forsthaus site is located within the Swiss Molasse Basin which is a thick tertiary sedimentary body. The purpose of the Forsthaus project is to create an underground thermal energy storage for waste heat from various surface sources that can then be back-produced during wintertime. The conceptual model of the project site is composed of porous and permeable sandstone layers embedded within a low porosity and low permeability matrix. Code_Bright, Retraso and MOOSE are used to model conditions at the Bern site. The aim of the Dutch projects is to analyse how different storage and geohydrological conditions affect recovery efficiency and associated environmental thermal impact. Storage conditions at the Koppert-Cress site will mostly be used as a reference but the site is composed of alternating layers of aquitards and aquifers. The SEAWAT simulator is used to model the Dutch conditions.

Academic codes have the potential to add significant value and innovation to reservoir modelling for a next generation of geothermal reservoir modelling software beyond HT-ATES. With this in mind, task 2.2 also includes the application of the CSMP++ code to two non-heat-storage geothermal projects. These are located in Azores and in Iceland. The Azores site is located in the Caldeiras da Ribeira Grande area which is a volcanic high enthalpy geothermal area. The goal is to use geochemical and geological data to simulate fluid flow in the area. The Iceland site is the Hengill geothermal area located in SW Iceland where two high enthalpy power plants are operated. The Hengill case study within HEATSTORE aims to use CSMP++ to develop a state-of-the-art process model of conditions at greater depths, in connection with plans of deeper drilling in the area. Subsequently, the plan is to incorporate the results from that model into the industrial field scale model of the area which currently excludes heat sources. Furthermore, the project seeks to model different production or injection scenarios for the deep well.

This report is subdivided into chapters that cover the specifics of each academic/research simulator. Each chapter has subsections on assumptions and governing equations, numerical discretization method, basic solution algorithm and logic, pre- and post-processing, the sites the code is used for and IP and code availability. An appendix with summarized information on input files for the simulators is provided at the end.

2 CSMP++/Nexus

2.1 Descriptive summary

Within HEATSTORE's Task 2.2, three modelling case studies invoke the CSMP++ code that has so far mainly been used in academic studies. The CSMP++ consortium describes the code as follows: *"The Complex Systems Modeling Platform (CSMP++) is an API for the simulation of natural processes and their interactions in complex space and time domains. It has been designed for the simulation of complex physics in geometrically complex and scale-variant domains. Its modular structure facilitates the analysis of the emergent properties of the studied system via large-scale mathematical simulations with models incorporating fundamental laws of nature, integrated for scenarios based on observations, problems and possible solutions. Governing PDEs are integrated using FEM and FVM methods and combinations thereof."* (CSMP++ consortium, 2015).

In the context of the Swiss Geneva case study, a simulator application was created by combining CSMP++ and the Nexus.sim application programming interfaces (API). Nexus.sim creates the simulator application and generates its solution algorithm, while CSMP++ was used for its Finite Element and Finite Volume functionality. Nexus (a.k.a. Nexus.sim) is a C++ high level API of integrative nature. It combines various functionalities from other APIs (e.g. CSMP++, VTK, GEMS, etc.) into one, is parallelized, and allows for a streamlined creation of customized simulator applications via algorithms generated from its own classes of simulation steps.

For the two case studies at the Hengill site (Iceland) and Fogo volcano (Azores), CSMP++ is used as described by Weis et al. (2014) and Scott et al. (2015) to simulate boiling and supercritical systems with explicit representation of magma bodies. This advanced functionality is absent from industry standard simulators and shall highlight how utilizing academic tools in an industrial environment has the potential to add significant value and innovation to reservoir modelling for a next generation of geothermal reservoir modelling software beyond HT-UTES.

2.2 Assumptions and Governing equations

The TH conceptual model used in these case studies neglects rock structural-mechanic (M) and fluid/rock chemical processes (C), and thus focuses on providing a first basic understanding of plausible fluid flow (H) and heat transfer (T) conditions. This means that the concept would allow the modelling of injection of a buoyant compressible percolating fluid that is able to exchange heat with its surroundings via advection and conduction processes. It is, however, not aimed at representing effects such as the thermal expansion of the rock surrounding the fluid, fracture mechanical/chemical sealing/opening, and chemical permeability alteration. For the Geneva HT-ATES study, complexity can further be reduced in that the model also assumes that only one single phase fluid (i.e. water) occupies the totality of the void space available in the rock (i.e. pore space). The Icelandic and Azores case studies involve high-enthalpy systems, i.e., the thermodynamics of boiling as well as the strong temperature-pressure dependence of water properties is incorporated. In all three cases, Darcy flow is assumed to be an appropriate approximation as are the instantaneous thermal equilibrium between fluid and rock, and the additive nature of conductive and advective heat transport mechanisms. Advective heat transport is performed by the fluid, while conductive heat transport is performed mainly by the rock material, depending on porosity.

An accurate equation of state for water (IAPS-84) allows capturing buoyancy effects, boiling, and transient pressure effects as accurately as possible in such a fluid-saturated medium. This leads to a set of partial differential equations (PDEs) with non-linear characteristics.

Considering the material properties of the porous rock to be isotropic, locally uniform to a minimum discrete level of a computational cell and constant, the governing equations that represent our

assumptions are dictated by classic (Bear, 1972) conservation laws of mass (1), momentum (2) (i.e. Darcy's Law), and energy (3) + (4), and may thus be expressed by:

$$\frac{\partial(\rho_f \phi)}{\partial t} + \nabla \cdot (\rho_f \mathbf{v}) + q_{\rho,f} = 0 \quad (1)$$

$$\mathbf{v} = -\frac{k}{\mu} (\nabla p - \rho_f \mathbf{g}) \quad (2)$$

$$\frac{\partial(\phi \rho_f h_f)}{\partial t} + \nabla \cdot (\rho_f h_f \mathbf{v}) = 0 \quad (3)$$

$$C_{p,t} \frac{\partial T}{\partial t} - \nabla \cdot (\lambda_t \nabla T) + q_H = 0 \quad (4)$$

where ρ_f is the fluid density, ϕ is the porosity, \mathbf{v} is the Darcy velocity vector, $q_{\rho,f}$ represents fluid mass sources and sinks, p is the fluid pressure, \mathbf{g} is gravitational acceleration, k is the permeability, and μ is the fluid viscosity. Equations (3) and (4) conform the law of conservation of energy, by assuming that heat is transported via advection only by the fluid (3), and by conduction via a combination of the rock and the fluid (4). We purposely neglect kinetic, potential, and viscous dissipation effects, h_f is the specific enthalpy of the fluid, $C_{p,t}$ is the non-specific bulk heat capacity, T is the bulk temperature, λ_t is the bulk thermal conductivity, and q_H represents heat sources and sinks. It is important to note that we also opt at this stage to ignore effects of salinity transport, and that such a feature could be added in the near future if expectations of its effect on the system becomes of greater concern.

For the Icelandic and Azores case, the above equations are augmented to adequate versions for two-phase liquid + vapor transport, the details being laid out by Weis et al. (2014).

2.3 Numerical discretization method

The computational approach for the Geneva site is based on a combined Finite Element (FE) and Finite Volume (FV) (Paluszny, et al., 2007) numerical discretization methodologies of the governing equations presented in Section 0. Equations (1) and (3) are discretized using a hybrid method of FE and FV, while equation (4) and a restructured version of equation (1) are based purely on FE. Equation (2) is deemed a post-process operation of velocity calculation and it can be evaluated after the solution of each timestep using FE shape functions. The combination of FE and FV formulations allow for the use of unstructured meshes, which in turn allows for a greater efficiency¹ when capturing geometrical detail.

The Azores and Icelandic case studies follow a modified version of the above approach that employs the Control Volume Finite Element approach (see Weis et al., 2014). We have found this to be superior in handling supercritical and two-phase conditions where often strong internal source terms occur. For the purpose of these studies, the 2D method of Weis et al. (2014) has been extended to 3D with spatial discretization being based on tetrahedral elements (Lamy-Chappuis et al., unpubl.).

¹ This efficiency is meant in terms of computer memory usage. Every element and node of the mesh occupies a particular amount of memory, and hence the lower the amount of them needed to discretize a particular geometry, the less computational memory is used and therefore more is available for other purposes.

Due to the mixture of formulations and the underlying approximation and continuity/integrability requirements for each equation, certain property “placement” options in the mesh become relevant. We can therefore speak of variables/properties being “placed” at the nodes (i.e. the corner points of the line segments, triangles, and tetrahedrons), or at the elements (i.e. whole individual elements, lines, triangles, or tetrahedrons).

Solution variables are *fluid pressure* and bulk² *temperature* and are placed on the nodes. This means that if after all calculations are finished we would like to identify pressure or temperature values anywhere else away from any element corner point, we will need to interpolate using the corresponding nodal values and approximate them (see **Error! Reference source not found.a**).

On the other hand, placing variables on the elements assumes that all of the element has an equal value for that property. The property may vary from element to element in a piece-wise fashion, but not within (see **Error! Reference source not found.b**). In our formulation, *fluid velocity*, *permeability*, and *thermal conductivity* are examples of properties that are placed on the elements.

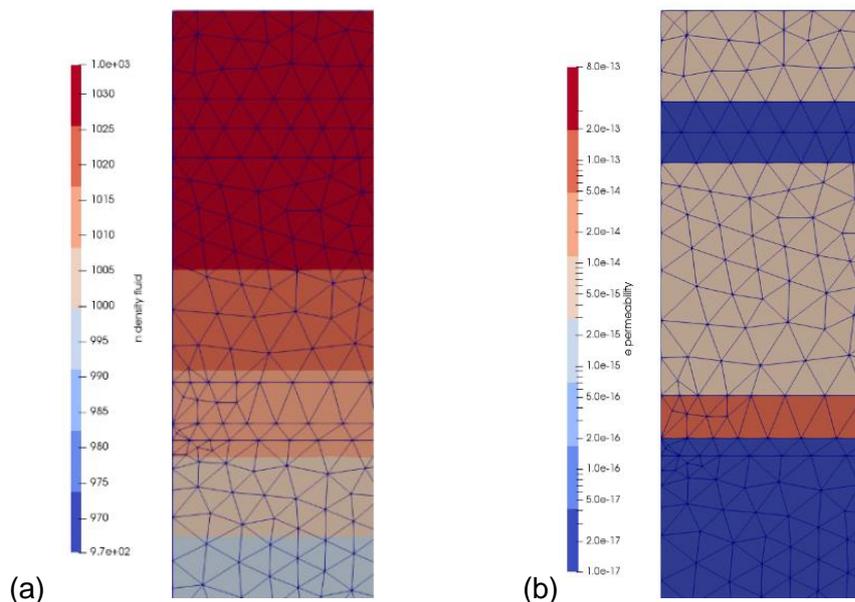


Figure 2.1 (a) Fluid density as a nodal property represented by colors. Color boundaries associated with the legend eventually fall within elements. Such boundaries estimates are possible via interpolation from the known nodal values. (b) Permeability as an element property represented by colors. Color boundaries are perfectly aligned with the element interfaces since a uniform constant value is available for each.

2.3.1 Initial and Boundary conditions

Partial differential equations (1) to (4) form a non-linear, coupled, initial and boundary value problem. To provide *closure* to this problem, boundary and initial conditions need to be supplied to obtain the desired time-sequence of unique solutions. Initial conditions are defined by the values of the main solution variables (i.e. pressure, temperature) and associated function-variables (i.e. density, specific enthalpy, heat capacity, sources and sinks) at the beginning of the simulation ($t = 0$) for the entire simulated domain. In the case of boundary conditions, we have used the *Dirichlet* type to impose the value of a variable at the boundary, and the *Neumann* type to impose a normal gradient or flux of a variable. For both initial and boundary conditions, imposed values may be x , y , z

² Bulk corresponds to the permeable rock including the fluid in its pore-space.

coordinate-dependent, while boundary conditions may also be time-varying. Other conditions similar to the Dirichlet condition are applied within the domain to control sources and sinks, which represent the wells in operation. Adiabatic conditions, also known as *natural* in relation to the applied FE and FV methods, are assumed for all boundaries parallel to the modelled groundwater flow direction.

Table 2.1: Summary of essential conditions applied during initialization and transient simulation. Unless specified, boundary locations not mentioned are assumed to be set to the *Natural* condition for a particular variable.

Variable	Type (location)	Value or function	Applied During	Comment
Pressure	Dirichlet (e.g. top boundary)	(e.g. 101325 [Pa])	Initialization	All other boundaries set to <i>natural</i> conditions (no-flow)
Pressure	Dirichlet (e.g. inlet and outlet boundaries)	Hydrostatic or function	Initialization and Simulation	Groundwater pressure is typically an input at the inlet boundary. In case of the Icelandic and Azores case studies, magma bodies are initialized to pressures higher than hydrostatic using pre-defined functions.
Temperature	Dirichlet (e.g. top boundary)	(e.g. 10 °C)	Initialization and transient simulation	Constant surface temperature condition. Inside domain, temperature distribution may be described by functions representative of the problem of interest.
Temperature	Neumann (e.g. bottom boundary)	Heatflux (e.g. 0.064 [W/m ²])	Initialization and transient simulation	Heat flux is converted to a temperature gradient via the fluid and rock thermal conductivity values adjacent to the boundary.
Enthalpy and Fluid Density	Inflow-Outflow (e.g. inlet and outlet boundaries)	Functions of p and T , depending on flow direction and upwind values	Transient simulation	Allows entry and exit of mass and enthalpy on boundaries of the domain without the need to set a <i>Dirichlet</i> condition for outflux.

To achieve the initial groundwater flow conditions we must solve a steady state problem based on equations (1) and (4). To achieve this we define a reference surface condition of temperature and pressure at depth 0 [m]. Values for lateral hydrostatic boundary conditions are obtained, and an extra pressure based on the groundwater hydraulic head is added only to one of the boundaries (i.e. the inlet) where the negative normal matches the groundwater flow direction. The boundary conditions are then fixed throughout the transient part of the simulation. A summary of the applied conditions is provided in Table 2.1, based on a cubic model with top, bottom, inlet (i.e. for groundwater flow) and outlet boundaries.

2.3.2 Material discontinuities

Material discontinuities are defined by a sharp change in porous medium properties. These are typically permeability, porosity, density (i.e. of the solid), thermal conductivity, and heat capacity. These sharp changes are honoured by the FE formulation via the placement of these properties on the elements (see **Error! Reference source not found.**). This capability will account for the existence of the various rock layers, and reef structures.

Faults and fractures are modelled as two dimensional regions with their own material properties. Essentially, their aperture is considered a very small feature in comparison to other model dimensions. Furthermore, their aperture characteristic or “thickness” can be assumed to be a complete void space (i.e. filled with fluid), or simply a highly permeable fracture or fault “zone”. This allows us to, for example, apply values of permeability to the fault/fracture zone, instead of calculating it as a hydraulic parameter based on the aperture length.

Faults and fractures are modelled via two dimensional surface elements which possess a “thickness” parameter that gives them their pseudo-3D characteristic. These elements are then included into the FE and FV formulations as any other three dimensional element. This capability allows for constructing grids which are computationally cheaper than filling a very thin void space with nearly flat volumetric elements. It must be noted, however that this approach allows for implementation of permeable faults and fractures. If any of these features are meant to be modelled as barriers (i.e. lower-than-matrix permeability), they will need to be meshed using volumetric elements.

2.3.3 Wells

Wells are discretized and included in the mesh in a similar way to faults and fractures, as described in Section 2.3.2. Instead of surface elements, however, line elements are used. The equivalence to a “thickness” is now considered a cross sectional area. A well can be set to a particular flow rate, in which case the total flow rate is evenly distributed among all the well elements. Instead, a fixed pressure may also be set by fixing pressure values along the nodes of each one of the elements of the well.

2.4 Basic Solution algorithm and logic

Each simulation performs a sequence of calculations, transient iterations, or “timesteps”, following an algorithm. The basic/simplified solution algorithm in the TH case can be best described by the following sequence of key steps,

1. From input files, read geometry, boundary conditions, fluid properties, and rock properties.
2. Compute *initial conditions* via the solution of the *steady state versions* of equations (1) and (4). In complex cases, non-linear iterations are required until convergence.
3. Compute initial fluid velocities (\mathbf{v}).
4. Begin transient iterations, and loop over these steps until the goal simulated time is reached:
 - a. Compute temperature diffusion using equation (4).
 - b. Compute advection of h_f , and ρ_f using equations (3) and (1), respectively.
 - c. Correct diffused temperature values using compressibility-related changes caused by the advection of fluid enthalpy via the definition of specific heat capacity. In the case of high-enthalpy systems (Iceland, Azores) this is achieved by an iterative algorithm (see Weis et al., 2014).
 - d. Compute fluid pressure p at the new time level $n + 1$ using a governing equation derived from equation (1).
 - e. Compute new fluid velocities (\mathbf{v}) using equation (2).

The result from such a simulation algorithm is a sequence of fluid pressure, fluid velocity, temperature, and any other dependent variable values at the end of each time-iteration (i.e. also known as “timestep”). The time evolution of these variables has thus been calculated, assuming a linear variation in time of all properties.

The pseudo-code simulation algorithm was transformed into C++ code using the Nexus.sim and CSMP++ frameworks, and compiled (for convenience) into a cross-platform application via a variety of compilers (i.e. Intel, GNU, Clang, etc.). In every supported operating system (Windows, MacOS, Linux), this application is intended to be used in conjunction with input files corresponding to each simulated scenario.

2.5 Pre-processing

TH simulations require a computational grid based on an arbitrarily-detailed geometry. The base geometry can be created in any CAD model, and in the current case it was created using the Petrel software. After trimming surfaces defining the different geological entities to the modelling size, they are then exported and subsequently imported into a 3D meshing software, ICEMCFD (ANSYS, 2010). For the case of the Icelandic Hengill system, the existing rock property distribution of a Voronoy-based grid of Reykjavik Energy's reservoir model was translated to a CAD geometry prior to following the same meshing strategy. As of the writing of this report, no pre-processing has yet been done for the Azores case study.

The geometrical model volume is then tessellated by unstructured/irregular tetrahedrons with reasonable resolution control, thus honouring material interfaces between rock-types, fractures, and well trajectories. Triangles are used to tessellate lower dimensional representations (i.e. interior surfaces) of fractures, and line segments to tessellate well completions. The latter triangles and line segments are often referred to as LDE's (Lower Dimensional Elements), since they possess at least one dimension lower than the maximum dimension used to represent the global domain.

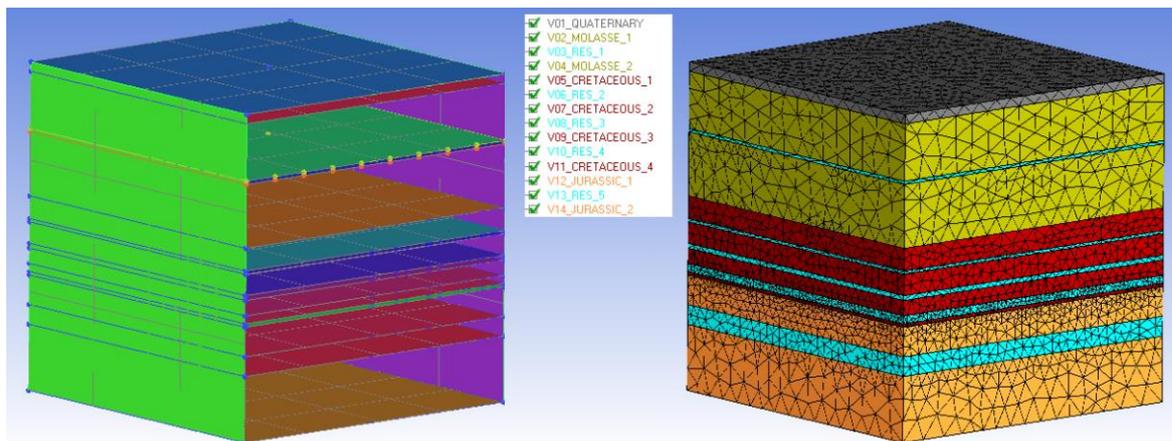


Figure 2.2 Simplified geometry of a simulation model (Left), and resulting tessellation using the ICEMCFD software (Right).

The mesh is output by ICEMCFD via its “CSP” output format into a set of two files with two different extensions. For any single mesh, the “.asc” file contains information about the names of the different groups of elements and their corresponding element numbers (i.e. for complete lines, surfaces, and/or volumes). The “.dat” file is output in binary format to save disk space and contains information regarding nodal coordinates, element connectivity, and overall numbering.

The resulting complete tessellation (i.e. grid or mesh, see Figure 2.2) is populated with all known material properties via the guidance of the simulator application's input files³, honouring the information provided by the studies described in (see (Driesner, 2019), Section 2.3.2). All lines, triangles, and tetrahedrons of the resulting grid are often termed “Elements” or “Cells”. With the grid

³ Input files provide all the information to the simulator when it is first executed. They typically provide values for all needed material properties, as well as initial, and boundary conditions. They also provide simulator control mechanisms such as time intervals, simulation end time, output frames, monitored values, etc. and are often of proprietary format. In our case, all input files other than the mesh are in ASCII (readable text) format.

and the simulator input files corresponding to the conditions set by each simulation scenario, the simulations are ready to be executed.

2.6 Post processing

Post processing of results from the simulator is carried out through two types of files. The first type is a column-based file that outputs a row of values per time-step. The values in the columns can be property ranges, singular values of a function, or the results from region or boundary integrals. These values could also be output yearly and represent, for example, ATES energy efficiency. See Figure 2. for a truncated example of a column based file, and Figure 2.4 for a sample 2D plot created from data from it.

Simulation_Time[s]	Energy_efficiency	Exergy_efficiency	Exergy_thres_efficiency		
31557600	0.667860234308	0.413762827253	0.211972023759	2.27902509224e+14	-1.5
63115200	0.719186435014	0.487822387565	0.304162474564	2.27959121746e+14	-1.6
94672800	0.744731285222	0.526782530731	0.353792595589	2.27969940383e+14	-1.6
126230400	0.761415053844	0.553068452213	0.387711675223	2.27973526226e+14	-1.7
157788000	0.773653951213	0.572791868824	0.413382302772	2.27974803944e+14	-1.7
189345600	0.783239358683	0.588503820341	0.433960352712	2.27975075357e+14	-1.7
220903200	0.791074520069	0.601521564226	0.451092851184	2.27974856197e+14	-1.8
252460800	0.797674701402	0.612610418851	0.465744298839	2.27974375906e+14	-1.8
284018400	0.803359173474	0.622251182088	0.478524207758	2.27973753885e+14	-1.8
315576000	0.808337995235	0.630764057434	0.489840556671	2.27973052022e+14	-1.8
347133600	0.81275659243	0.638373067998	0.499979972602	2.27972309691e+14	-1.8
378691200	0.816719704192	0.645240990709	0.509151441678	2.27971550487e+14	-1.8
410248800	0.820305030908	0.651489514476	0.517511669129	2.27970789316e+14	-1.8
441806400	0.823571935474	0.657212301325	0.525181606273	2.27970033642e+14	-1.8
473364000	0.826566815635	0.662483044914	0.532256658344	2.27969290151e+14	-1.8

Figure 2.3 Sample truncated snapshot of ATES efficiency data presented in columns of text.

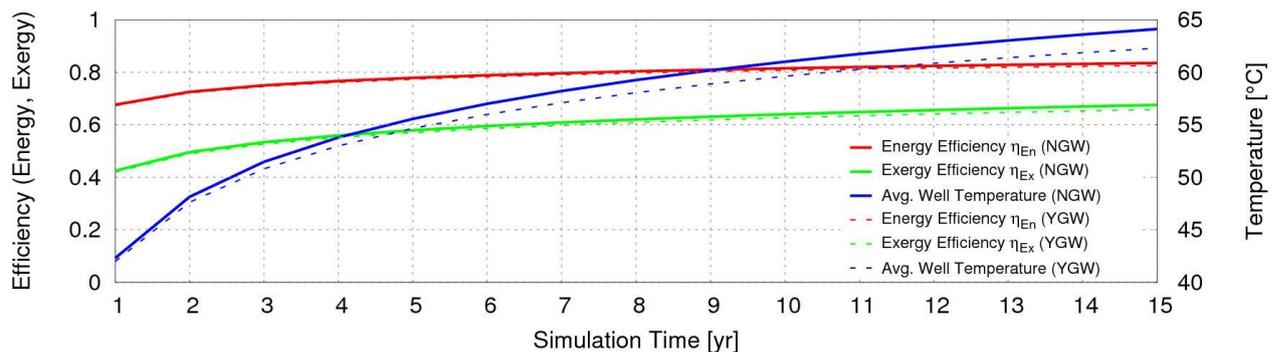


Figure 2.4 Sample plots available from ATES efficiency data of the Geneva site. Energy and Exergy efficiency variation with time for the expected ATES lifetime (e.g. 15 yrs) comparing similar scenario cases with and without groundwater flow. The right-hand axis depicts temperature measurement at well GW_1 for the end of each cycle.

The second type of file is more complex and involves ASCII/binary files in the VTK format (Lorensen, 2010). There are many extensions available for the VTK format, and the extension used by our simulator is specifically *.vtu, which corresponds to output for unstructured grids (i.e. although structured grids can also be output in this format). VTU files contain information regarding the mesh, and the associated values of properties placed on entities such as elements and nodes. These results may be viewed and analysed via external visualizers such as the open-source software

Paraview (www.paraview.org, see Figure 2.5). The simulator input files provide control for what properties are output to the VTU files, thus controlling file size and output frequency. It is up to the user to realize what properties are relevant for a better post-processing analysis.

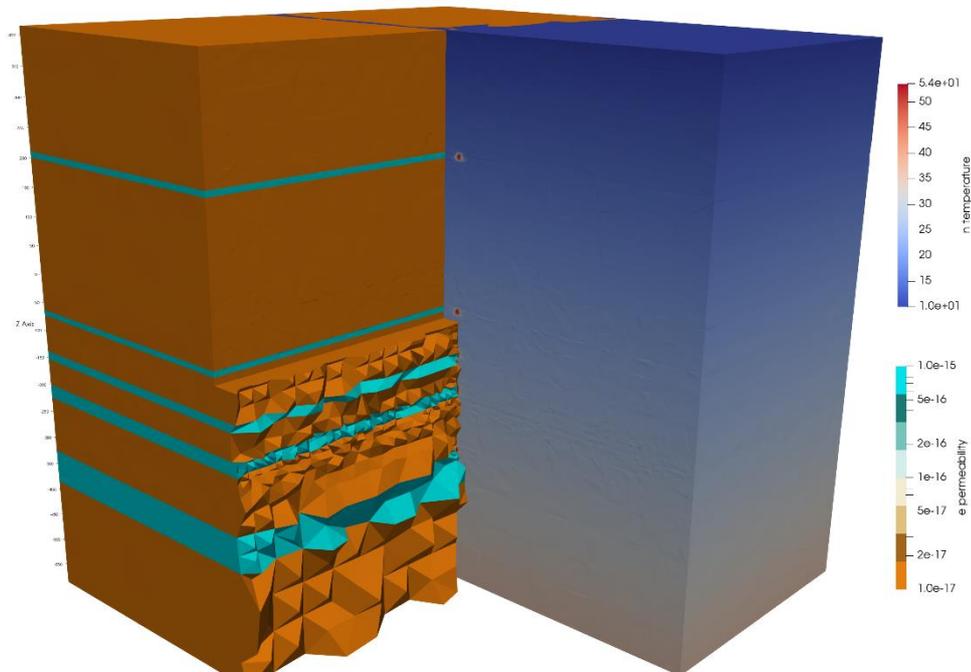


Figure 2.5 Sample Paraview representation of results available as VTU files for the Geneva site, which include the mesh, material, and flow property values.

2.7 Swiss pilot site Geneva

There are no particularities that need special address with the HT-ATES feasibility study carried out in Geneva. The simulations involve a number of scenarios where various configuration parameters are varied to understand their individual and combined impact. The study involves changes to the geometry, the mesh, as well as various material and operational parameters (e.g. injection rates, temperatures, time periods, etc.). All configurations used are defined in the scenario descriptions available in (Driesner, 2019), Section 2.3.7. A sample set of input files for one particular scenario can be seen in Appendix A.

2.8 Azores study

The Azores case study targets to explore the possible driving forces and large-scale flow patterns of the geothermal resources on the northern flanks of Fogo volcano. The existing reservoir model is limited to the drilled extent of the resource. In the context of HEATSTORE Task 2.2 we will use the conceptual model outlined in Figure 2. to explore if the known structure and flow patterns can be explained by a magmatic heat source at depth. Such a larger-scale model may add significant value to future exploration and exploitation strategies.

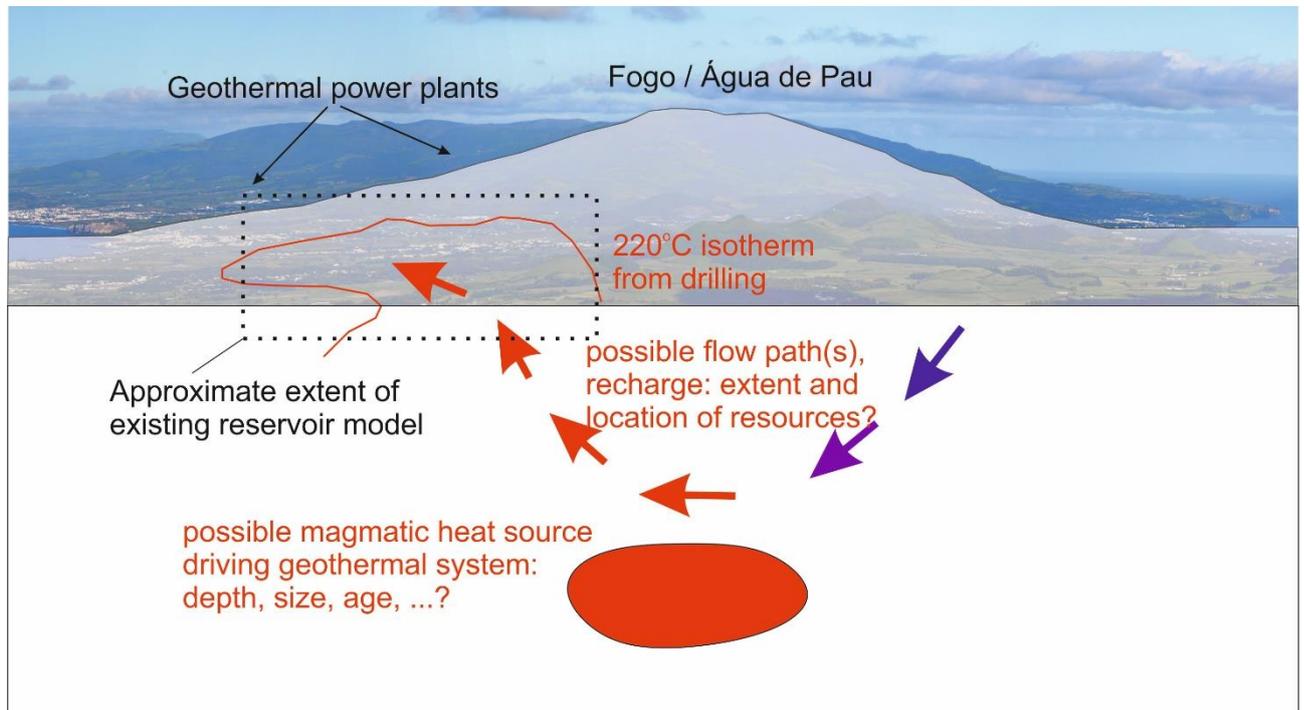


Figure 2.6 Conceptual model of the possible geothermal hydrology of Fogo volcano. The existing TETRAD-based reservoir model (black dotted rectangle) employs artificial boundary conditions to reproduce the observed thermal structure and flow patterns. Within HEATSTORE Task 2.2 we will explore if a larger model with an explicitly represented magmatic heat source is able to provide a more comprehensive view of the system to provide a framework for future exploration and resource utilization.

2.9 Icelandic case study

The Icelandic case study within HEATSTORE follows a similar strategy as the Azores one. It aims to use CSMP++ to develop a state-of-the-art process model of conditions at greater depths and subsequently incorporate the results from that model into the existing industrial TOUGH2-based field scale reservoir model of the area (Gunnarsson et al., 2011). It is critical to develop modeling tools that can simulate deep conditions and proximity to heat sources (magmatic intrusions) before the drilling of a >4 km deep well commences in the Hengill area but such drilling is planned within the next few years as part of the Iceland Deep Drilling Project. The current plan is to simulate different intrusion emplacement scenarios in the Hengill area; different shapes, sizes, depths and emplacement times to try to reproduce the temperature field data observed in Hengill.

The field scale model is run with the commercial numerical simulation code TOUGH2 (Pruess, Oldenburg & Moridis, 2012) as implemented in forward and inverse mode within the iTOUGH2 code (Finsterle, 2007). The code describes the conservation of mass and energy and is a multiphase flow and transport simulation program for fractured and porous media (Pruess, Oldenburg & Moridis, 2012). Conduction and convection control the heat flow. Thermodynamic conditions are based on local equilibrium of all phases. Advection controls the mass flow and a multiphase version of Darcy's law is used to calculate advective mass fluxes in each phase (Pruess, Oldenburg & Moridis 2012). The field scale model uses the equation of state module EOS1 which describes water in liquid, vapor and two-phase state. The standard TOUGH2 module that the field scale model uses does not handle supercritical conditions and thus does not allow to explicitly represent the magmatic heat sources in the model structure, which is a limitation as supercritical resources could be expected to be located

at the top of such bodies (Scott et al., 2015). This limitation is solved by only simulating the upper part of the system within the field scale model and simulating the roots of the system by adjusting bottom boundary conditions that emulate injecting heat and hot fluid into the system at depth.

Unlike the Azores case, the Hengill reservoir model is laterally comprehensive. The permeability structure has been adapted such that it reproduces temperature distribution and production history. According to the model, the horizontal permeability values within the geothermal system range from $2 \cdot 10^{-18}$ – $4 \cdot 10^{-12}$ m² and the vertical permeability values range from $2 \cdot 10^{-18}$ – $1 \cdot 10^{-13}$ m². The model has manually been translated to a CAD model and meshed with ICEMCFD (ANSYS, 2010). An example of simulation results from a simplified representation of the Hengill area is shown in Figure 2..

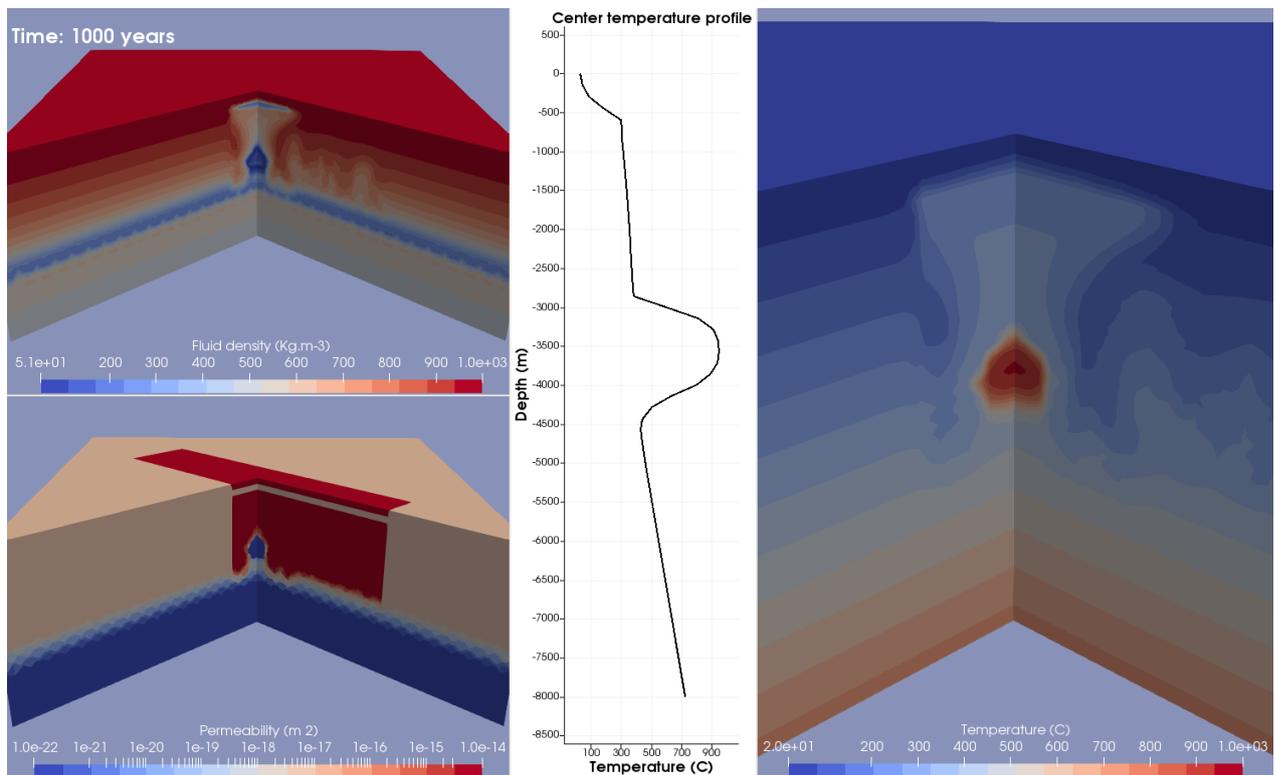


Figure 2.7 Reconnaissance simulation of a highly simplified representation of the Hengill area with a single magmatic intrusion at 3 km depth at the centre of the field, including a cap-rock.

As of the writing of this report, the 3D version of the CSMP++ CVFE implementation is being used to simulate how individual magmatic heat sources below the bottom of the current reservoir model would shape the thermal and flow patterns in the field.

In the later phases of the project we will explore how the CSMP++-based results from the process model simulations can aid in the location of the IDDP-3 well and then be implemented in the TOUGH2-based current model. A variety of approaches are possible that need to be evaluated as the first results are obtained. Currently the most plausible version is based on the idea that CSMP++ based-models naturally generate "correct" flow and thermal patterns that are mainly a function of the permeability distribution. If we succeed to obtain in this way a virtual geothermal field with the main characteristics of the Hengill system, then slices can be cut through that model that allow deriving boundary conditions that can later be used in the TOUGH2 model. Furthermore, the project seeks to model different production or injection scenarios for the deep well. These approaches are expected to be described in more detail in the Month 34 version of this report.

2.10IP and code availability

“CSMP++ can be licensed for academic, commercial and public sector (NGO) use. The license can be obtained from the CSMP Originator group. Currently (September 2019) it is represented by the Technology Transfer of the ETHZ, Switzerland ("Meyns Silke (F&W)" silke.meyns@sl.ethz.ch).” (CSMP++ consortium, 2015)

Nexus.sim is in its final developmental stages of its first release, and is aimed at being an open source project (contact: Julian Mindel, julian.mindel@erdw.ethz.ch).

2.11 References

ANSYS, 2010. *ANSYS ICEMCFD Brochure*. [Online]

Available at: <https://www.ansys.com/-/media/Ansys/corporate/resourcelibrary/brochure/ansys-icem-cfd-brochure.pdf> [Accessed 11 09 2019].

Bear, J., 1972. *Dynamics of Fluids in Porous Media*. New York: Dover Publications.

CSMP++ consortium, 2015. *CSMP++ v 1.0*. [Online]

Available at: <http://igp-sim.ethz.ch/index.php/csmp/2014-04-18-23-32-51> [Accessed 09 09 2019].

Driesner, T. (ed.), 2019. *HEATSTORE - Initial report on tools and workflows for simulating subsurface dynamics of different types of High Temperature Underground Thermal Energy Storage*, s.l.: GEOTHERMICA – ERA NET Cofund, 146 pp.

Finsterle, S. (2007). *iTOUGH2 User's Guide*. (LBNL-40040). California: Lawrence Berkeley National Laboratory.

Gunnarsson, G., Arnaldsson, A. and Oddsdóttir, A. L. (2011). Model Simulations of the Hengill Area, Southwestern Iceland. *Transport in Porous Media*, 90, pp. 3–22.

Lorensen, B., 2010. *VTKFileFormats*. [Online]

Available at: <https://lorensen.github.io/VTKExamples/site/VTKFileFormats/> [Accessed 09 09 2019].

Paluszny, A., Matthaei, S. K. & Hohmeyer, M., 2007. Hybrid finite element–finite volume discretization of complex geologic structures and a new simulation workflow demonstrated on fractured rocks. *Geofluids*, pp. 186–208.

Pruess, K., Oldenburg, C. and Moridis, G. (2012). *TOUGH2 User's guide, Version 2*. (LBNL-43134). California: Lawrence Berkeley National Laboratory.

Scott, S., Driesner, T. and Weis, P. (2015). Geologic controls on supercritical geothermal resources above magmatic intrusions. *Nature Communications*, 6(7837) DOI: 10.1038/ncomms8837

Weis P., Driesner T., Coumou D., and Geiger S. (2014). Hydrothermal, multiphase convection of H₂O-NaCl fluids from ambient to magmatic temperatures: a new numerical scheme and benchmarks for code comparison. *Geofluids*, 14, pp. 347-371.

3 MOOSE

3.1 Descriptive summary

The Multiphysics Object Oriented Simulation Environment (MOOSE) is a computational framework that allows users to develop fully-coupled, multiphysics, multiscale, parallel, finite element applications relatively easily (<https://www.mooseframework.org/>; Gaston et al., 2015). Due to its flexible coupling capacity, it has been used to simulate many physical phenomena from geomechanics to nuclear reactor physics to chemical transport to superconductivity. It utilizes the Portable, Extensible Toolkit for Scientific Computation (PETSc) for massive parallelization. MOOSE provides physics modules that are actively supported by the developer team, including *Porous Flow*, *Tensor Mechanics*, and *Heat Conduction*, which can be coupled together seamlessly for thermo-hydro-mechanical (THM) simulation capability. Each physics module has its own set of PDEs, material properties, boundary and initial conditions, and postprocessor quantities. MOOSE is capable of using structured and unstructured meshes, which can be produced within MOOSE or using external meshing software (e.g. Trelis, CUBIT, ICEMCFD). Material properties can be assigned to irregularly shaped regions within the domain to mimic geological units at HT-ATES sites. The Geothermal Energy and Geofluids (GEG) group at ETH has expertise in using MOOSE and a history of collaboration with the MOOSE developers at Idaho National Laboratory.

3.2 Assumptions and Governing equations

We consider the equations of thermo-poro-elasticity, which is a form of the coupled THM equations (Cheng, 2016). The thermo-poro-elastic equations are coupled through several terms including advective heat transfer, porosity, permeability, the fluid equation of state, and effective stress. For example, increases in pressure and temperature can cause expansion of the solid skeleton, which is associated with changes in porosity and permeability, which both appear in the hydrologic and thermal equations. The equations, assumptions, and constitutive relationships are discussed in more detail below. Equations (5), (8), and (9) represent the governing partial differential equations (PDEs) for hydrology, energy transport, and geomechanics, respectively. These equations are solved in terms of the primary variables of pressure, temperature, and the three components of displacement, respectively.

Hydrologic Equation:

The flow equation is derived from the continuity equation for fluid mass and the full Darcy's equation, which accounts for the relative velocity between the fluid and the solid.

$$\frac{\partial(\rho_f \phi)}{\partial t} + \rho_f \phi \cdot \mathbf{v}_s - \nabla \cdot \left(\frac{\rho_f k}{\mu} (\nabla P - \rho g) \right) - q = 0 \quad (5)$$

In the preceding equation, ρ_f is the fluid density, ϕ is the porosity, t is time, $\mathbf{v}_s = \partial \mathbf{u} / \partial t$ is the solid skeleton velocity where \mathbf{u} is the displacement vector of the solid skeleton, k is the permeability, μ is the fluid viscosity, P is the pore pressure, g is gravity, and q is the fluid source. The first term represents fluid accumulation, the second and third terms represent the flow of fluid into and out of the control volume with respect to the motion of the solid skeleton, and the fourth term represents sources and sinks. MOOSE contains many options for defining fluid properties, and for now we elect for a simple model wherein viscosity is constant and density is a function of pressure and temperature:

$$\rho_f = \rho_o \exp\left(\frac{P}{K_f} - \alpha_f T\right) \quad (6)$$

Where K_f is the fluid bulk modulus, T is the temperature, α_f is the thermal expansion coefficient of the fluid, and ρ_o is a reference density. Using the simple fluid properties facilitates specification of

thermo-hydro boundary conditions. For example, the enthalpy flow rate of a fluid injection at a known temperature can be calculated a priori and imposed as a boundary condition. However, viscosity is dependent on fluid temperature, and a more accurate option for fluid properties comes from the International Association for the Properties of Water and Steam (IAPWS) (Wagner et al., 2000). The porosity model is (Detournay and Cheng, 1993):

$$\frac{\partial \phi}{\partial t} = (\alpha_B - \phi) \frac{\partial}{\partial t} (\epsilon_{kk} - \alpha_T T + \frac{1 - \alpha_B}{K} P) \quad (7)$$

where α_B is the Biot coefficient, ϵ_{kk} is the volumetric expansion, “kk” indicates summation following index notation, α_T is the volumetric coefficient of thermal expansion of the solid skeleton, and K is the bulk modulus of the drained porous skeleton. This equation tells us that porosity increases with volumetric expansion of the solid skeleton and increased pore pressure. Likewise, porosity decreases with increases in temperature, if all else is held constant, because the solid grains expand. The porosity alters the permeability through the Kozeny-Carman relationship. The flow equations assume that flow is fully-saturated and single phase without any chemical reactions or transport.

Thermal Equations:

The heat equation is derived from the conservation of energy:

$$\frac{\partial K}{\partial t} + K \nabla \cdot \mathbf{v}_s - \nabla \cdot \left(\lambda \nabla T + h_f \rho_f \frac{k}{\mu} (\nabla p - \rho g) \right) - q_t = 0 \quad (8)$$

where $K = (1 - \phi) \rho_r c_r T + \phi \rho_f \epsilon_f$ is the heat energy density, ρ_r is rock density, c_r is the heat capacity of rock, T is the temperature, ϵ_f is the internal energy of the liquid, h_f is the specific enthalpy of the fluid, and q_t represents heat sources. The first term is the accumulation of thermal energy, the second term can be thought of as the advection of thermal energy via solid skeleton motion, the third term is thermal conduction, and the fourth term is heat advection via fluid motion. The internal energy and enthalpy for the exponential EOS are, respectively, $e = e_o + \int_{T_o}^T c_v(T) dT$ and $h_f = e + \gamma \frac{P}{\rho}$ where e_o is the reference enthalpy at a reference temperature T_o , $c_v(T)$ is the heat capacity of water at constant volume, and γ is the pore pressure coefficient, which is usually set to one. We assume that $e_o = 0$, $T_o = 0$ °C, and $c_v(T)$ is constant so that $e = c_v T$. This is a reasonable approximation for the relatively small temperature changes applicable to HT-ATES. The thermal equation assumes that radiative heat transfer and viscous dissipation are negligible and assumes local thermal equilibrium between the fluid and the rock matrix.

Mechanical Equations:

The equilibrium equations express Newton’s Second Law of motion:

$$-\rho_{mat} \frac{\partial}{\partial t} \left(\frac{\partial u_j}{\partial t} \right) + \frac{\partial \sigma_{ij}}{\partial x_i} + \rho_{mat} b_j = 0 \quad (9)$$

where the subscripts employ index notation so that Equation (9) represents three equations, corresponding to a force balance in each of the three dimensions. $\rho_{mat} = (1 - \phi) \rho_r + \rho \phi$ is the undrained density of the fluid/matrix mixture, and b_j is the body force, which corresponds to gravity. The first term is the acceleration term, which is sometimes neglected under the assumption of quasi-static equilibrium. In order to capture the thermo- and poro-elastic coupling, the definition of effective stress incorporates pore pressure and temperature:

$$\sigma'_{ij} = \sigma_{ij} + \alpha_B P \delta_{ij} + \alpha_T \delta_{ij} T \quad (10)$$

Where δ_{ij} is the Kronecker delta that equals 1 if $i = j$ and equals 0 if $i \neq j$. We neglect plastic strains because we assume that they would be unacceptably large and lead to wellbore damage and/or

damage to surface infrastructure. We also assume that the thermal production due to strains is negligible and that strains are small, reversible, and can be expressed as:

$$\epsilon_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (11)$$

3.3 Numerical discretization method

Spatial Discretization

MOOSE uses the Galerkin Finite Element method, typically with Lagrange shape functions that can be of arbitrary order. One of the advantages of using MOOSE is that the code allows users to provide Kernels that act as terms for the differential equation that is solved. The *Porous Flow* module kernels used to construct the heat transfer equation (Eq. (8)) are shown in Table 3.1. Kernels can be specified to construct the mechanical and hydrologic governing equations as well, and MOOSE will automatically create the weak form based on the governing PDEs, couple the equations correctly, and choose suitable basis functions.

Table 3.1 Kernels used to specify the heat transfer governing equation.

Kernel	Mathematical Term
PorousFlowEnergyTimeDerivative	$\partial K / \partial t$
PorousFlowHeatVolumetricExpansion	$K \nabla \cdot \mathbf{v}_s$
PorousFlowHeatConduction	$-\nabla \cdot (\lambda \nabla T)$
PorousFlowHeatAdvection	$-\nabla \cdot \left(h \frac{\rho k}{\mu} (\nabla P - \rho \mathbf{g}) \right)$

Temporal Discretization:

The time stepping is flexible within MOOSE. Users can specify adaptive or rigid time stepping with initial, and maximum time step sizes. More detail is provided in the “Basic Solution algorithm and logic” section.

Numerical stabilization and Mass Lumping:

Numerical diffusion and over/undershoots of the temperature solution can be a problem for advection and diffusion processes in some finite element formulations. To address these issues, the *Porous Flow* module offers mass lumping and numerical stabilization (full upwinding or Kuzmin-Turek stabilization). Full upwinding is relatively computationally inexpensive but introduces lots of numerical diffusion, while KT stabilization is computationally expensive while minimizing numerical diffusion. Since the equations of heat transfer are highly diffusive, the numerical diffusion from the full upwinding scheme may be an acceptable approximation, given the computational expense of KT stabilization.

3.3.1 Initial and Boundary conditions

It is very important to understand how boundary and initial conditions are applied in THM simulations, because they interact with each other and can cause non-intuitive or incorrect results if applied without careful consideration. For example, an elevated temperature BC induces thermal expansion

of fluid and the porous media, which has to be taken into account in the assignment of the hydrologic and mechanical boundary conditions.

Initial Conditions:

We follow an approach wherein *deviations* in primary variables, rather than their initial values are of interest. Under this approach, $P = 0$ corresponds to hydrostatic pressure and $\sigma = 0$ and $\epsilon = 0$ correspond to an initial condition in the pre-existing stress state. This approach is commonly-used in the poroelastic literature (Wang, 2000). We assume that $T = 0$ corresponds to an initial geothermal gradient. This assumption may result in errors in vertical heat conduction, but we think they will be minor relative to the amount of heat that is injected and extracted at the wells. Under this approach, the initial conditions are $P = 0$, $\sigma = 0$, $\epsilon = 0$, and $T = 0$.

Boundary Conditions:

Type I and Type II boundary conditions can be applied as *PresetBCs*, which specify the value or the gradient of a primary variable at a boundary. This is done for mechanical boundary conditions where displacements or stresses are specified as zero, either due to symmetry or due to being far away to avoid boundary effects. There are also a set of boundary conditions designed specifically for handling fluid flow and heat transfer within MOOSE's *Porous Flow* module (e.g. *PorousFlowSink* and its variations). These boundary conditions allow code users to specify physically consistent flow and temperature boundary conditions, which become relevant for the thermal-hydro coupling. For example, they can be used to ensure that the amount of enthalpy that is produced at an outflow boundary is based on the flow of liquid and the local temperature of the liquid.

3.3.2 Material discontinuities

Material parameters (e.g. Young's modulus, Poisson's ratio, permeability, porosity, thermal conductivity, etc.) can be specified to different regions of the model that reflect different geological layers. These regions can be specified by a MOOSE- or externally-generated mesh. We adopt an equivalent porous media approach to treat the hydrologic effect of fractures and faults on the flow field. This smooths out the hydraulic effect of fractures over a larger region instead of explicitly resolving fractures and faults as lower-dimensional elements. A similar equivalent porous media approach can be adopted to apply thermal and mechanical properties in fractured regions. For example, highly faulted or fractured regions could employ a more compliant value of Young's modulus. Fractures and faults are not included explicitly in the simulations, since resolving the poroelastic stresses and deformations within a fracture and matrix is a cutting-edge area of research and is generally limited to 2D modelling and/or a single fracture/fault (e.g. Jin and Zoback, 2017; Jha and Juanes, 2014), and there are many unanswered, important research questions about HT-ATES that can be explored with the equivalent porous media approach in three dimensions.

3.3.3 Wells

There are three general approaches to model wells within MOOSE, as shown in Figure 3.1. In order of increasing complexity, they are: (1) treat the wells as a point or line source for fluid and heat, (2) treat the wells as flat boundaries at the exterior of the mesh, and (3) treat the well as a rounded boundary at an interior or exterior boundary. Approach (1) and (2) assume that the well does not alter the local stress and strain. While this is incorrect in the immediate vicinity of the wellbore, it is justified because we simulate domains with spatial scales on the order of kilometres, while the local stress variations due to the well are much, much smaller, on the order of one wellbore radius (Kirsch, 1898). Approach (1) has the advantage that the well location can easily be moved within a mesh, without re-meshing, and no special considerations need to be considered for mechanics. As far as we are aware, approach (1) has not been proven to extract the correct amount of enthalpy at a production well, which is a potential drawback. Approach (2) can be used if there is symmetry within the conceptual model. The well(s) under this approach have to be at the exterior of the domain, and

they should use the same mechanical boundary condition as the rest of the domain boundary that they occupy, which is illustrated by the roller boundary condition in Figure 3.1(b). Approach (3) best reflects reality near the wellbore but has challenges with complexity and workflow. For example, the local mechanical stress boundary conditions need to be carefully considered inside and near the wellbore, and complicated meshing would be required to move the well location within the mesh.

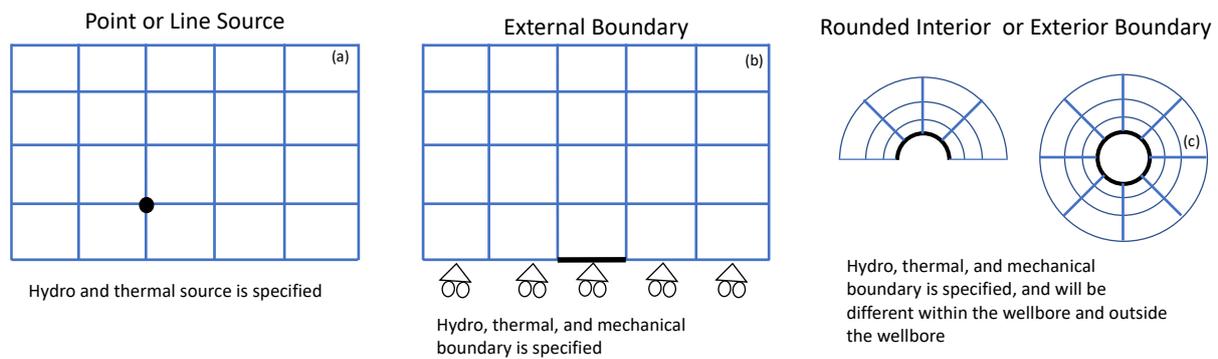


Figure 3.1 Wells can be treated as (a) a point or line source, (b) a boundary condition applied on a flat, exterior boundary, if there is symmetry, or (c) a boundary condition applied on a curved interior or exterior boundary. The blue lines represent the mesh, while the black point and lines represent the point source or boundaries where the well exists.

3.4 Basic Solution algorithm and logic

The THM governing equations (Equations (5), (8), and (9)) should all have a right-hand side value equal to zero if the numerical solution for the primary variables is perfectly correct. In reality, there will be some error in the numerical solution, and the right-hand side of each of these equations will have some non-zero residual value. The solution algorithm involves improving the numerical solution until each set of equations meet convergence criteria wherein the maximum residual at any point in the domain and the sum of all the residuals are below user-defined tolerances. The equations are solved monolithically (i.e. simultaneously and with a single solver). Since the tolerance for each set of equations (i.e. hydro, thermal, and mechanical) depends on the problem of interest, the residuals are typically weighted so that they have similar magnitudes. For example, if the acceptable residual of the thermal solution near steady-state is expected to be five orders of magnitude larger than the acceptable residual of the hydrologic solution, then the residual of the thermal equation is scaled by 10^{-5} so that the residuals are of similar magnitude, thereby helping the solver converge more effectively.

The *Porous Flow* module uses implicit time stepping and a Newton-Raphson iteration for THM simulations. The *Porous Flow* module has been optimized based on the physics of coupled THM problems. At the beginning of a new time step, the residual is calculated based on the current estimate of the primary variables (pressure, displacement, and temperature for THM simulations). Due to the potential for nonlinearities, there is usually an iteration to solve for the values of the primary variables within each time step. The iteration has the following steps:

- 1) The residual is calculated from the differential equations and the current estimate of the primary variables. If the convergence criteria are met, then the simulation advances to the next time step.
- 2) If the convergence criteria are not met, then the Jacobian is formed from the current estimate of the primary variables.

- 3) Then the Jacobian is inverted (which may involve a PETSc nonlinear iteration itself) and used to calculate a new estimate of the primary variables following the Newton-Raphson method. The process begins again at step one.

3.5 Pre-processing

Pre-processing involves incorporating information from the geological model and the energy systems scenario models into the THM model. The geological model and energy systems scenarios are both provided by UniGe. The geological information includes the depths and thicknesses of rock layers, the material properties of the rocks, and locations of major faults. The energy systems scenario models provide information about the desired injection and extraction flow rates, duration, and temperatures. With this information, a mesh can be generated and material properties can be assigned to reflect the geology of the Geneva and Bern sites, and well boundary conditions can be assigned to reflect the energy system scenarios. Because THM models are computationally expensive, there is some simplification required when assigning geological and system-integration data in the THM simulation scenarios. Nevertheless, the simulations will represent the Geneva and Bern sites, while also exploring uncertainty in the parameter space, with the hope that general conclusions can be drawn about HT-ATES. An example of the pre-processing workflow can be seen in Birdsell and Saar (2020).

3.6 Postprocessing

Stress, strain, pressure, surface uplift, temperature, well flow rate, and many other variables can be output and visualized in 3D or to text or csv files. This allows us to explore surface deformation as a function of geological parameters (e.g. aquifer permeability, Young's modulus) and operating decisions (e.g. well spacing, injection temperature). We can also examine where the thermal plume moves and how geological parameters and operational decisions alter deformation over the decades-long lifetime of a HT-ATES system. Predicted stresses within the domain can be shared with UniNe for their near-wellbore geomechanical study. Python, Paraview, MATLAB, and/or Excel will be used to analyse and plot the output data.

3.7 Swiss pilot site Geneva

A thorough site description of the Geneva pilot project was provided in D2.1 (Driesner, 2019). In early simulations of the site, we utilized the values presented in Table 3.2 and the conceptual model in **Figure 3.2** (Birdsell and Saar, 2020). That early work used hydro-mechanical coupling, and the thermal coupling will be added to future work. The input file is provided in the appendix for reference.

Table 3.2 Base Case Model Parameters from Birdsell and Saar (2020).

Parameter	Value	Unit
Injection Duration	216	d
Injection Rate	50	L/s
Fluid Bulk Modulus	2	GPa
Fluid Viscosity	0.001	Pa-s
Fluid Density at $P = 0$	1000	kg/s
Gravity	0	m ² /s
Aquifer Permeability ($k_{aquifer1}, k_{aquifer2}$)	10^{-15}	m ²
Aquifer Porosity ($\phi_{aquifer}$)	0.15	-
Aquitard Permeability ($k_{aquitard}$)	10^{-17}	m ²

Parameter	Value	Unit
Upper Aquitard Porosity ($\phi_{aquitard1}$)	0.02	-
Middle Aquitard Porosity ($\phi_{aquitard2}$)	0.02	-
Lower Aquitard Porosity ($\phi_{aquitard3}$)	0.01	-
Rock Young's Modulus (E)	50	GPa
Poisson's Ratio (ν)	0.3	-
Biot Coefficient (α)	1.0	-

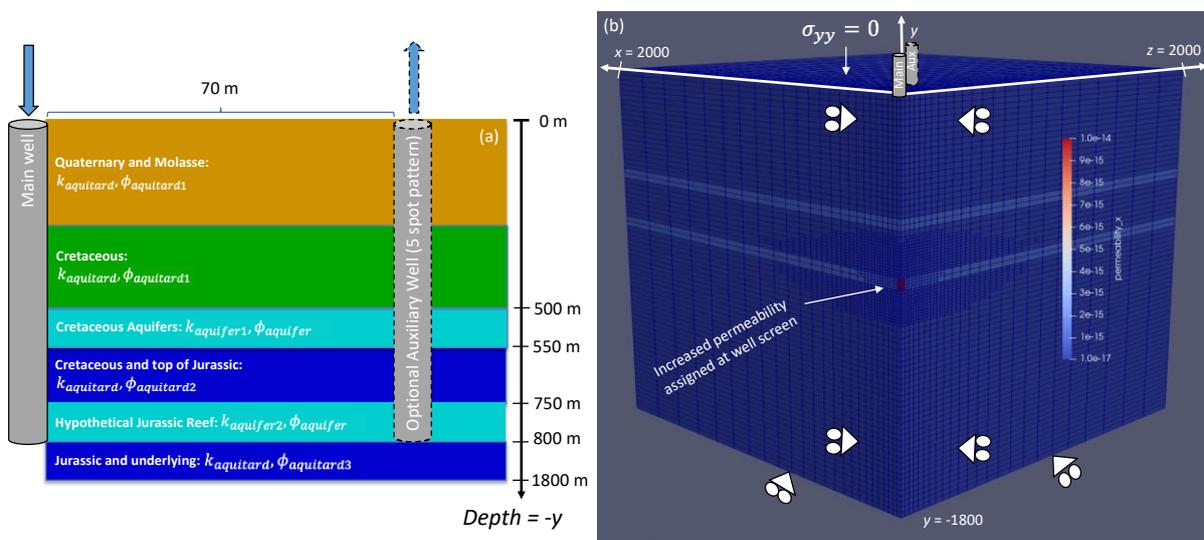


Figure 3.2 (a) Conceptual and (b) numerical model from Birdsell and Saar (2020). The input file for the base-case scenario is in the Appendix.

3.8 Swiss pilot site Bern Forsthaus

The Bern Forsthaus site was described in D2.1 (Driesner, 2019). The Geneva site will be prioritized by the GEG group. But Bern will also be investigated since comparison between the two sites could offer interesting insights.

3.9 IP and code availability

MOOSE is an open-source code developed primarily by Idaho National Laboratory. It is publicly available on GitHub with documentation at <https://www.mooseframework.org/>.

3.10 References

Birdsell, D. T., & Saar, M. O., 2020. *Modeling Ground Surface Deformation at the Swiss HEATSTORE Underground Thermal Energy Storage Sites*. World Geothermal Congress 2020. Reykjavik, Iceland: Submitted for publication.

Cheng, A. H., 2016. *Poroelasticity*. Oxford, Mississippi, USA: Springer.

E. Detournay & A. Cheng., 1993. "Fundamentals of poroelasticity," Chapter 5 in *Comprehensive Rock Engineering: Principles, Practice and Projects, Vol II, Analysis and Design Method*. Also available online. Pergamon Press, New York.

Driesner, T. (ed.), 2019. *HEATSTORE - Initial report on tools and workflows for simulating subsurface dynamics of different types of High Temperature Underground Thermal Energy Storage*, s.l.: GEOTHERMICA – ERA NET Cofund, 146 pp.

Gaston, D. R. et al., 2015. Physics-based multiscale coupling for full core nuclear reactor simulation. *Annals of Nuclear Energy*, 84, pp. 45-54.

Birendra, J. & Juanes, R., 2014. Coupled multiphase flow and poromechanics: A computational model of pore pressure effects on fault slip and earthquake triggering. *Water Resources Research*, 50(5), pp. 3776-3808.

Jin, L., & Zoback, M. D., 2017. Fully coupled nonlinear fluid flow and poroelasticity in arbitrarily fractured porous media: A hybrid-dimensional computational model. *Journal of Geophysical Research: Solid Earth*, 122(10), pp. 7626-7658.

Kirsch, G., 1898. Die Theorie der Elastizität und die Bedürfnisse der Festigkeitslehre. *Zeitschrift des Vereines deutscher Ingenieure*, 42, pp. 797–807.

Wagner, W., Cooper, J. R., Dittmann, A., Kijima, J., Kretschmar, H., Kruse, A., Mareš, R., Oguchi, K., Sato, H., Stöcker, I., Šifner, O., Takaishi, Y., Tanishita, I., Trübenbach, J., and Willkommen, T. (January 1, 2000). "The IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam ." ASME. *J. Eng. Gas Turbines Power*. January 2000; 122(1): 150–184.

Wang, H. F., 2000. *Theory of Linear Poroelasticity with Applications to Geomechanics and Hydrogeology*, Princeton Series in Geophysics. Princeton, NJ.

4 CODE_BRIGHT and RETRASO

4.1 Descriptive summary

Code_Bright is a Finite Element Method (FEM) program capable of performing coupled thermo-hydro-mechanical (THM) analysis in geological media. It has been developed at the Department of Civil and Environmental Engineering of the Polytechnic University of Catalonia (UPC), and works combined with the pre/post-processor GiD, developed by the International Center for Numerical Methods in Engineering (CIMNE). Details can be found in Olivella et al. (1996).

Code_Bright has been coupled to Retraso, that simulates transport processes (advection, dispersion and diffusion) and chemical reactions (acid-base reactions, redox, complexation, adsorption, cation exchange, precipitation and dissolution of minerals). Code_Bright calculates the flow and heat, which is used by Retraso to calculate reactive transport. Details can be found in Saaltink et al. (2004).

4.2 Assumptions and Governing equations

Code_Bright solves the balances of solid, water, air, energy and/or momentum (Table 4.1). Moreover, constitutive laws are used to write variables such as darcy flow (\mathbf{q}), mass fluxes (\mathbf{j}), phase densities (ρ), mass fractions (ω), and saturation (S) as a function of state variables such as deformation (\mathbf{u}), liquid and gas pressure (p_l and p_g) and/or temperature (T).

Table 4.1 Balance equations solved by Code_Bright. Superscripts a and w refer to air and water. Subscripts g , l , s and e refer to gas, liquid, solid and energy. e is internal energy, i is non advective flux, j is advective flux, p is pressure, S is saturation, u is deformation, ρ is density, σ is stress tensor, ϕ is porosity, and ω is mass fraction. More details can be found in Olivella et al. (1994).

Equation: effective stress	$\sigma' = \sigma - \max(p_g, p_l)$
	For compression positive
Equation: Mass balance of solid	$\frac{\partial(\rho_s(1-\phi))}{\partial t} + \nabla \cdot (j_s) = 0$
	Or
	$\frac{D_s \phi}{Dt} = \frac{(1-\phi)}{\rho_s} \frac{D_s \rho_s}{Dt} + (1-\phi) \nabla \cdot \frac{du}{dt}$
Equation: Mass balance of water	$\frac{\partial((\omega_g^w \rho_g S_g + \omega_l^w \rho_l S_l) \phi)}{\partial t} + \nabla \cdot (j_g^w + j_l^w) = 0$
	Or
	$\phi \frac{D_s(\omega_l^w \rho_l S_l + \omega_g^w \rho_g S_g)}{Dt} + (\omega_l^w \rho_l S_l + \omega_g^w \rho_g S_g) \frac{D_s \phi}{Dt} + (\omega_l^w \rho_l S_l + \omega_g^w \rho_g S_g) \phi \nabla \cdot \frac{du}{dt} + \nabla \cdot (j_l^w + j_g^w) = 0$
Equation: Mass balance of gas (dry air)	$\frac{\partial((\omega_g^a \rho_g S_g + \omega_l^a \rho_l S_l) \phi)}{\partial t} + \nabla \cdot (j_g^a + j_l^a) = 0$
	Or
	$\frac{D_s(\omega_l^a \rho_l S_l + \omega_g^a \rho_g S_g)}{Dt} + (\omega_l^a \rho_l S_l + \omega_g^a \rho_g S_g) \frac{D_s \phi}{Dt} + (\omega_l^a \rho_l S_l + \omega_g^a \rho_g S_g) \phi \nabla \cdot \frac{du}{dt} + \nabla \cdot (j_l^a + j_g^a) = 0$
Equation: Energy balance	$\frac{\partial(e_s \rho_s (1-\phi) + e_g \rho_g S_g \phi + e_l \rho_l S_l \phi)}{\partial t} - \frac{\phi S_g p_g}{\rho_g} \frac{\partial \rho_g}{\partial t} + \nabla \cdot (i_c + j_{es} + j_{eg} + j_{el}) = f^Q$
Equation: Momentum balance	$\nabla \cdot \sigma + \mathbf{b} = \mathbf{0}$

Given the flow of liquid and/or gas and temperature, Retraso solves the reactive transport equations (Table 4.2). Thermodynamic equilibrium constants and rate laws may depend on temperature.

Table 4.2 Balance equations solved by Retraso. Subscripts g , l , and s refer to gas, liquid and solid. c is a vector of concentrations, D is the diffusion/dispersion coefficient, q is Darcy flux, r_k is the vector of kinetic reaction rates, S_k is the stoichiometric matrix for kinetic reactions, U is the component matrix, θ is volume fraction, and ρ is density. $L()$ is a linear operator for transport.

Mass balance of
chemical components

$$\sum_{\alpha} \frac{\partial \theta_{\alpha} \rho_{\alpha} U c_{\alpha}}{\partial t} = \sum_{\alpha} L_{\alpha}(U c_{\alpha}) + U S_k^T r_k \quad \alpha = l, g, s$$

$$L_l() = -\nabla \cdot (\rho_l q_l()) + \nabla \cdot (\theta_l \rho_l D_l \nabla())$$

$$L_g() = -\nabla \cdot (\rho_g q_g()) + \nabla \cdot (\theta_g \rho_g D_g \nabla())$$

$$L_s() = 0$$

Reactions	
Aqueous complexation	Equilibrium mass action law and the extended Debye Hückel with B-dot for activity
Sorption	Adsorption isotherms: Linear, Freundlich, Langmuir Cation exchange Electro static models: Constant capacity, Double layer, Triple layer
Gas-liquid interaction	Equilibrium mass action law
Precipitation-dissolution	Equilibrium mass action law and kinetic approach

4.3 Numerical discretization method

Table 4.3 shows the numerical methods for discretization. The numerical discretization of Code_Bright and Retraso is the same. Details can be found in Olivella et al. (1996) and Saaltink et al. (2004).

Table 4.3 Discretisation characteristics of Code_Bright and Retraso.

Discretisation methods	Finite element method for space Finite differences in time
Dimensions	1D, 2D and 3D 1D and 2D elements in 3D meshes 1D elements in 2D meshes
Type of elements	1D: Linear segments 2D: Linear triangle (analytical), linear quadrilateral (4 GP + selective integration), quadratic triangle (3 GP), 3D: Linear tetrahedron (analytical), linear hexahedron (8 GP), quadratic tetrahedron (4 GP), triangular prism (6 GP)

4.3.1 Initial and Boundary conditions

Code_Bright and Retraso have standard Initial and boundary conditions and parameterization (Table 4.4).

Table 4.4 Initial and boundary conditions and parameterization in Code_Bright and Retraso.

Boundary conditions	
Equations: boundary conditions. T problem	<p>Prescribed temperature and mixed for heat flow rate:</p> $\mathbf{j}_e = j_e^0 + \gamma_e(T^0 - T) + e_i \mathbf{j}_i$ <p>where j_e, T and E_i are the heat flux, temperature and internal energy at the boundary and j_e^0, γ_e and T^0 are parameters</p>
Equations: boundary conditions. H problem	<p>Prescribed pressure and mixed for phase flow rate:</p> $\mathbf{j}_l = j_l^0 + \lambda_l(P_l^0 - P_l)$ <p>where j_l and P_l are the liquid mass flux and pressure at the boundary and j_l^0, λ_l and P_l^0 are parameters.</p>
Equations: boundary conditions. M problem	<p>Prescribed for displacement rate and mixed for force or stress:</p> $\boldsymbol{\sigma} = \boldsymbol{\sigma}_0 + \gamma_m(\mathbf{u}^0 - \mathbf{u})$ <p>where $\boldsymbol{\sigma}$ and \mathbf{u} are the stress tensor and displacement vector at the boundary and $\boldsymbol{\sigma}_0$, γ_m and \mathbf{u}_0 are parameters.</p>
Reactive transport	<p>Prescribed mass flux of a chemical component</p> <p>Prescribed concentrations</p> <p>Mixed flux-concentration condition</p> $j_i = \alpha(m_i^0 - m_i)$ <p>where j_i and m_i are the mass flux and concentration of chemical component i at the boundary and α and m_i^0 are parameters.</p>
Initial conditions	
Equations: initial conditions. T problem	Initial temperature and initial porosity
Equations: initial conditions. H problem	Initial pressures and initial porosity
Equations: initial conditions. M problem	Initial stress and initial porosity
Reactive transport	Initial concentrations
Parameterization	
Zonation of parameters	Parameters are given for each material. Materials are assigned to geometry volumes. The geometry volumes are discretised into finite elements.
Anisotropy	Parameters for anisotropic elasticity (Young modulus), thermal (thermal conductivity) or hydraulic (intrinsic permeability) are given at material level. Anisotropy angle is assigned on the geometry volumes.

4.3.2 Material discontinuities

Fractures for TH equations can be represented by 2D elements included in 3D meshes. In other words, surfaces can have properties for flow of fluids.

Coupling with the mechanical may imply changes in permeability. In this case, aperture can change with deformation, then permeability is calculated as a function of aperture. This is a kind of embedded discontinuity in which, the effect of the fractures is homogenised at the level of finite elements (Olivella and Alonso, 2008).

4.3.3 Wells

For both Code_Bright and Retraso wells can be represented with standard elements or lower dimensionality elements. Lower dimensionality elements can be used for 2D or 3D models.

4.4 Basic Solution algorithm and logic

Both Code_Bright and Retraso use the fully coupled Newton-Raphson method (Table 4.5). For reactive transport problems, at each time step first the thermohydraulic problem is solved by Code_Bright. Then, the fluxes and temperatures are used to solve the reactive transport by Retraso. Automatic time stepping using different alternative criteria can be applied.

Table 4.5 Numerical methods for Code_Bright and Retraso.

Solution methods (Code_Bright)	THM	Non-linear equations: Fully coupled, NR method Linear system: Direct method (LU decomposition + back substitution) or iterative (conjugate gradient squared method). Direct solver uses band storage while iterative solver uses compressed sparse row format
Solution methods (Retraso)	Reactive Transport	Non-linear equations: Fully coupled, NR method (also called Direct Substitution Approach) Linear system: Direct method (LU decomposition + backsubstitution) with band storage
Independent (state) variables for THM problems		Displacement, liquid pressure, gas pressure, temperature and solute concentration
Independent (state) variables for Reactive transport		Concentrations of primary species

4.5 Pre- and postprocessing

Codebright is integrated into GiD, a pre and post processor for finite elements (www.gidhome.com). No manipulation of files is necessary.

Also, a pre- and postprocessor exists for Retraso (VisualRetraso).

4.6 Swiss pilot site Bern Forsthaus

Both Code_Bright and Retraso as described above will be used to model the pilot site of Bern Forsthaus. A thorough description of the modelling approach was described in D2.1 (Driesner, 2019, chapter 2.4). Thermohydraulic (TH), thermohydrochemical (THC) models are planned.

4.7 IP and code availability

Both Code_Bright and Retraso are free. Code_Bright has a web page that includes updated versions of CODE_BRIGHT, user's guide, support documentation, examples and tutorials.

https://deca.upc.edu/en/projects/code_bright

RETRASO can be downloaded from:

<https://h2ogeo.upc.edu/en/investigation-hydrogeology/software/155-retraso-en>

Licenses for the pre- and postprocessor GiD can be purchased at CIMNE (www.gidhome.com).

4.8 References

Driesner, T. (ed.), 2019: *Initial report on tools and workflows for simulating subsurface dynamics of different types of High Temperature Underground Thermal Energy Storage*. GEOTHERMICA – ERA NET Cofund, 146 pp.

Olivella, S., Carrera, J., Gens, A. & Alonso, E.E., 1994. Nonisothermal multiphase flow of brine and gas through saline media. *Transp. Porous Media*, 15, pp. 271–293.

Olivella, S.; Gens, A.; Carrera, J.; Alonso, E.E. Numerical formulation for a simulator (CODE_BRIGHT) for the coupled analysis of saline media. *Eng. Comput.* 1996, 13, pp. 87–112.

Olivella, S.; Alonso, E.E. Gas flow through clay barriers. *Géotechnique* 2008, 58, pp. 157–176.

Saaltink, M.W., Batlle, F., Ayora, C., Carrera, J., Olivella, S. RETRASO, a code for modeling reactive transport in saturated and unsaturated porous media. *Geologica Acta*, 2004, 2(3), pp. 235-251.

5 SEAWAT simulator

5.1 Descriptive summary

SEAWATv4 is a groundwater model that couples MODFLOW (Harbaugh et al., 2000) and MT3DS (Zheng and Wang, 1999) and is designed to simulate groundwater flow and transport of heat and solutes. Previous versions of SEAWAT did not have the ability to simultaneously calculate the effect of temperature and solute concentrations on variable-density flow. In SEAWATv4 both temperature and solute concentration can linearly contribute to fluid density and according buoyancy flow. Next to this, the effect on fluid viscosity variation (mainly impacted by groundwater temperature) are taken into account (Langevin, 2009). SEAWATv4 is controlled by using the flopy model, which is operated in the python environment (Bakker et al., 2016).

5.2 Assumptions and Governing equations

The model is based on the finite difference simulator as used in the MODFLOW model (Harbaugh et al., 2000). MODFLOW solves the groundwater and transport equations with differential element numerical methods. The flow is calculated based on gradients in x,y and z direction, discharge at the sources or sinks and the storage in the aquifer. The Preconditioned Conjugate-Gradient package is used to solve the finite difference equations in each step of a MODFLOW stress period. Elaborate documentation of the PCG(2) package can be found in Hill (1990).

MT3DMS is designed to simulate solute transport. It is a modular three-dimensional multi-species transport model. MT3DMS simulates advection, dispersion, sorption and reaction of solute species in the groundwater (Langevin et al., 2008). The solute transport equation solved by MT3DMS expresses the rate of change of concentration. To account for sorption of species to the soil material, a species-specific retardation factor is used.

SEAWAT combines MODFLOW and MT3DMS to calculate the flow and heat transport at every time step. It is designed to simulate three-dimensional, variable density groundwater flow and multi-species transport (Langevin et al., 2008). It includes the effect of viscosity changes with temperature, difference in density.

Viscosity is calculated as (Langevin, 2009):

$$\mu(C,T) = 2.494 \cdot 10^{-5} \cdot (10^{\frac{248.37}{T+133.15}}) + 1.92 \cdot 10^{-6} (C - C_0) \quad (1)$$

The **density** equation of state, as is used in SEAWATv4 is (Thorne, Langevin & Sukop, 2006):

$$\rho(C,T) = \rho_f + \frac{\partial \rho}{\partial C} (C - C_0) + \frac{\partial \rho}{\partial T} (T - T_0) \quad (2)$$

Where ρ_f is the density of freshwater, $\frac{\partial \rho}{\partial C} (C - C_0)$ is the change in density due to concentration

change and $\frac{\partial \rho}{\partial T} (T - T_0)$ is the change in density due to a changing temperature. This means that

both solute concentration and temperature have a linear relation with density. For solute concentration this is limited violation of real fluid dynamics, however, the temperature-density relation is not linear at temperatures >25 °C (Figure 5.1).

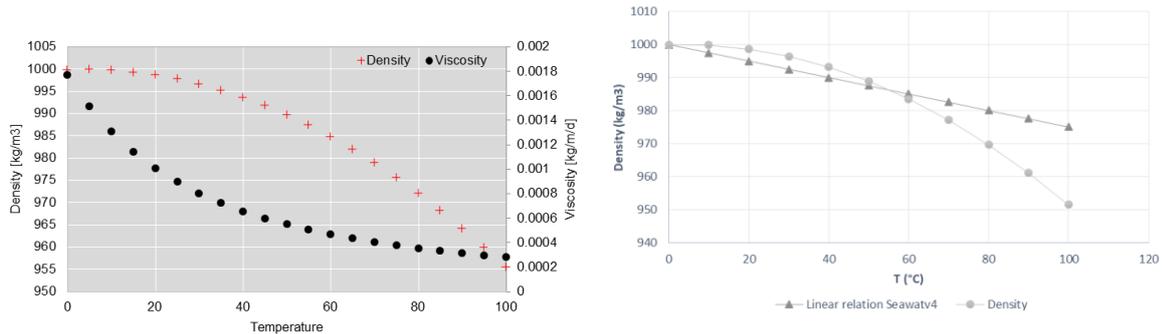


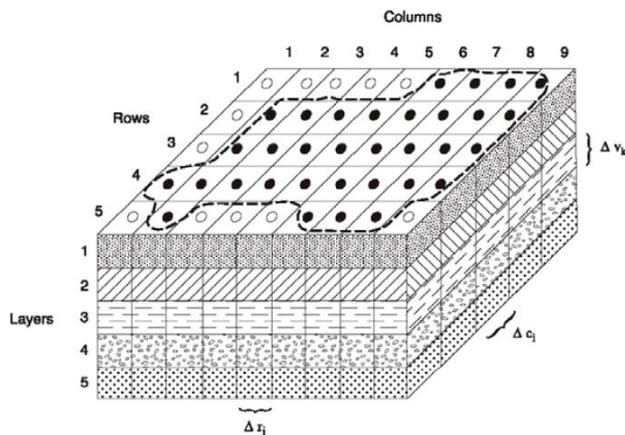
Figure 5.1 Left) Relation of density and viscosity with temperature. Right) Relation between temperature and density versus the linear approximation used in Seawat, shown here for denselp = - 0.25. Denselp stands for the slope of the linear density-temperature gradient (drho/dT).

Therefore, the density equation of state has been modified in the FORTRAN SEAWAT executable to calculate the **non-linear temperature-density** relation, similar to (van Lopik, Hartog & Zaadnoordijk, 2016). To do this we excluded the ability to include the effect of solute concentration, which results in:

$$\rho(T) = \rho_f - \frac{(T - 4)^2}{207} \quad (3)$$

We are working on a version where both solute concentration and the non-linear temperature relation can be included.

5.3 Numerical discretization method



EXPLANATION

- AQUIFER BOUNDARY
- ACTIVE CELL
- INACTIVE CELL
- Δr_j DIMENSION OF CELL ALONG THE ROW DIRECTION—
Subscript (j) indicates the number of the column
- Δc_i DIMENSION OF CELL ALONG THE COLUMN DIRECTION—
Subscript (i) indicates the number of the row
- Δv_k DIMENSION OF CELL ALONG THE VERTICAL DIRECTION—
Subscript (k) indicates the number of the layer

Figure 5.2 Example of a discretized model with aquifers and aquitards. Source: Harbaugh et al. (2000).

As can be seen in Figure 5.2, a MODFLOW grid consists of rows with width Δr_j , columns Δc_i and layers (vertical direction) Δv_k . The exchange of water from one cell to another (MODFLOW), and subsequently the transfer of heat/solutes (MT3DS) is calculated following the finite-difference equation that takes 6 adjacent cells into account (Figure 5.3). Looking at exchange from one cell to another (in this case from cell i,j,k and $i,j+1,k$) following Darcy's law the flow is (Harbaugh et al., 2000):

$$q_{i,j-1/2,k} = KR_{i,j-1/2,k} \Delta c_i \Delta v_k \frac{(h_{i,j-1,k} - h_{i,j,k})}{\Delta r_{j-1/2}} \quad (4)$$

Where:

- $h_{i,j,k}$ is the head at node i,j,k , and $h_{i,j-1,k}$ is the head at node $i,j-1,k$;
- $q_{i,j-1/2,k}$ is the volumetric flow rate through the face between cells i,j,k and $i,j-1,k$
- $KR_{i,j-1/2,k}$ is the hydraulic conductivity along the row between nodes i,j,k and $i,j-1,k$
- $\Delta c_i \Delta v_k$ is the area of the cell faces normal to the row direction; and
- $\Delta r_{j-1/2}$ is the distance between nodes i,j,k and $i,j-1,k$ (L)

By combing the grid dimensions ($\Delta c_i \Delta v_k$ & Δr_j) and the hydraulic conductivity the hydraulic conductance between both cells can be calculated:

$$CR_{i,j-1/2,k} = \frac{KR_{i,j-1/2,k} \Delta c_i \Delta v_k}{\Delta r_{j-1/2}} \quad (5)$$

By substituting this in equation 4, the flow can be calculated as:

$$q_{i,j-1/2,k} = CR_{i,j-1/2,k} (h_{i,j-1,k} - h_{i,j,k}) \quad (6)$$

By combining all flows from the adjacent cells a solution can be calculated, more detailed information and examples can be found in Harbaugh et al. (2000). In SEAWATv4, viscosity (equation 1) is also added to affect the conductance (CR), this is in more detail described in Thorne, Langevin & Sukop (2006).

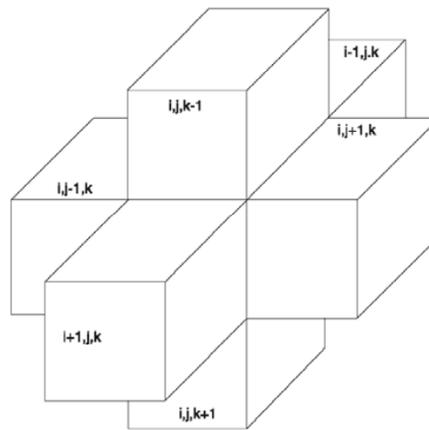


Figure 5.3 Indices for the six adjacent cells surrounding the specific cell (i,j,k) which is in the middle of the six visible cells. Source: Harbaugh et al. (2000).

The transfer of heat can subsequently be calculated by MT3DS (Langevin et al., 2008):

$$\left(1 + \frac{1-\theta}{\theta} \frac{\rho_s c_{Psolid}}{\rho c_{Pfluid}}\right) \frac{\partial(\theta T)}{\partial t} = \nabla \cdot \left[\theta \left(\frac{k_{Tbulk}}{\theta \rho c_{Pfluid}} + \alpha \frac{\mathbf{q}}{\theta} \right) \cdot \nabla T \right] - \nabla \cdot (\mathbf{q}T) - q'_s T_s \quad (7)$$

where

- ρ_s is the density of the solid (mass of the solid divided by the volume of the solid) [ML⁻³]
- c_{Psolid} is the specific heat capacity of the solid [L²T⁻²K⁻¹]
- c_{Pfluid} is the specific heat capacity of the fluid [L²T⁻²K⁻¹]
- k_{Tbulk} is the bulk thermal conductivity of the aquifer material [ML²T⁻²K⁻¹]
- T_s is the source temperature [°K].
- α the dispersivity tensor [L]

In SEAWATv4, the VDF process solves the variable-density groundwater flow equation as (Langevin et al., 2008):

$$\nabla \cdot \left[\rho \frac{\mu_0}{\mu} \mathbf{K}_0 \left(\nabla h_0 + \frac{\rho - \rho_0}{\rho_0} \nabla z \right) \right] = \rho S_{s,0} \frac{\partial h_0}{\partial t} + \theta \frac{\partial \rho}{\partial C} \frac{\partial C}{\partial t} - \rho_s q'_s \quad (8)$$

where

- ρ_0 is the fluid density [ML⁻³] at the reference concentration and reference temperature
- μ is dynamic viscosity [ML⁻¹T⁻¹]
- \mathbf{K}_0 is the hydraulic conductivity tensor of material saturated with the reference fluid [LT⁻¹]
- h_0 is the hydraulic head [L] measured in terms of the reference fluid of a specified concentration and temperature (as the reference fluid is commonly freshwater).
- $S_{s,0}$ is the specific storage [L⁻¹], defined as the volume of water released from storage per unit volume per unit decline of h_0
- t is time [T]
- θ is porosity [-]
- C is salt concentration [ML⁻³]
- q'_s is a source or sink [T⁻¹] of fluid with density ρ_s

Modelgrid

For efficient use of computational power most modelling work done in the HEATSTORE project an axisymmetric grid is used (Langevin, 2008). The grid applied within an axisymmetric SEAWAT grid is a vertical section of one row, with distance along the columns and depth along the layers (Figure 5.5). The wells screen is connected to different cells (layers) in the first column.

An HT-ATES doublet is represented by 3 rows, where the two outer rows, model the warm and cold well, while the middle row was set to inactive to prevent interaction between the outer rows. The horizontal discretization is approximately (1 to 5) m at the well face, the cell size grows towards the outer boundary of the model, in a predefined number of steps. The Courant number is the ratio transport distance during one time step over the cell size (Courant, Friedrichs & Lewy, 1967) and should be smaller than 1 for accurate calculation. The applied time step has a large influence of this

(e.g. day or month). In general the courant number should be smaller than 1 at several meters away from the well and onwards, where buoyancy flow, conduction, dispersion affect the simulation results.



Figure 5.4 grid building schedule for axisymmetric grid. Close to well same Δc_i are set (dmin), from a set distance (dmin_{bound}) the grid is logarithmically increasing towards a maximal value (dmax). Around_{all} defines the total size of all columns combined towards the outer boundary.

As can be seen in Figure 5.4, the grid used for the axisymmetric simulations done in this project is build based on some parameters. The vertical discretisation (dz) is equal for all layers in the grid (usually 0.5m). dmin is the smallest cellsize of the column width and is usually 1m. This continues until it reaches dmin_{bound} (usally 100-200m), from there the grid increases logarithmically from dmin to dmax (usually 100m). This continues until the minimal total column width is reached, restrained by Around_{all} (usually 2500).

To do axisymmetric modeling, some paramters have to be altered to represent the radial model field (Figure 5.5a). This means that while moving away from the well, some parameters increase with a factor 2π times the distance of the incremental cell to represent an entire circle, instead of only one cell (Louwyck et al., 2014). The parameters that are being corrected are:

- Permeability tensor (κ_{xyz})
- Porosity (θ)
- Specific Storage (SS)
- Bulk density (ρ_{bulk})

5.3.1 Initial and Boundary conditions

Boundary conditions are set to constant at the top, outer boundary and bottom of the axisymmetric modelling grid (e.g. temperature = 12 °C = ambient groundwater temperature (T_{amb}), pressure head = 0). The Left side of the domain, where the well is implemented, the cells are normal active cells and no boundary conditions is set here. Water, heat and solutes can therefore only flow from the well into the rest of the modelling domain (Figure 5.5b).

When ambient groundwater flow in the storage aquifer or other aquifers in the modelling domain is modelled, a 3D grid is used with a head difference (Δh) between one side of the modelling domain and the other. Ambient groundwater flow cannot be implemented when using an axisymmetric grid.

5.3.2 Material discontinuities

In the modelling approaches done with SEAWAT only unconsolidated materials are simulated, following the conditions in NL. When impermeable layers (or almost impermeable) layers are modelled, the permeability is set to a very low number (e.g. vertical permeability clay <0.001 m/day).

5.3.3 Wells

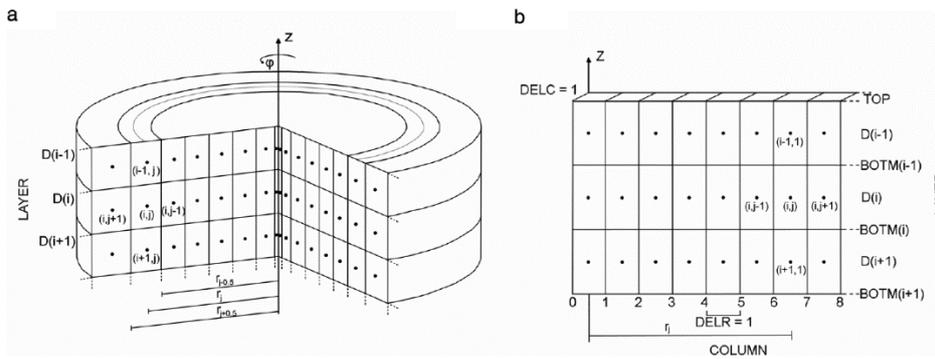


Figure 5.5 a) Representation of an axisymmetric, finite-difference grid. b) MODFLOW/SEAWAT grid structure, multiple rows can be used next to each other separated by an inactive row.

Wells are being implemented with the well package of MODFLOW, which is a specified flux boundary. In the axisymmetric grid, the well is located at the left boundary, or column 0 (Figure 5.5b). This package equally divides the total flux (injection or extraction, m³) proportional to the layers' hydraulic conductivity over the amount of cells (column 0) that are connecting to the well. In most cases the well is penetrating the entire depth of the aquifer and is therefore connected to all cells in column 0 in Figure 5.5b.

5.4 Basic Solution algorithm and logic

See 5.3.

5.5 Preprocessing

Define input data & make input file

Information about the specific geohydrology is studied to define proper layering of the system. From this also the geohydrological properties (K_h , K_v) are defined. The layering of the input file has to be similar to the Δz used in the model (e.g. 0.5m or 1m).

Next to this, the input properties of the modelled heat storage system have to be defined. An example input can be:

600,000 m³/year, T=80 °C.

Another way to simulate a heat storage system is to define daily values for the volume (in or out) and temperature, based on for instance real data. This should also be defined in this step (also see 5.8).

Set up model properties & grid in Flopy

Define all settings and input parameters (see Appendix) for the different parts of the model (MODFLOW, MT3DS and SEAWATv4). With this the grid is built (Figure 5.4), the SEAWAT model is initialized and the initial conditions are set.

Run model

When the model is ran, data is saved for each period (e.g. day) of parameters such as: injected volume, extracted volume, temperature of well(s), distribution of energy in the subsurface etc. Next to this the raw head and temperature data is saved as .UCN files as output. These files can afterwards be used to retrieve the final state of each period (head/temperature).

5.6 Postprocessing

Recovery efficiency

$$\eta(t_0 \rightarrow t) = \frac{E_{out}}{E_{in}} = \frac{\int_{t_0}^t (T_{out} - T_{amb}) \cdot Q_{out} \cdot c_w \cdot \rho_w dt}{\int_{t_0}^t (T_{in} - T_{amb}) \cdot Q_{in} \cdot c_w \cdot \rho_w dt} = \frac{\Delta \bar{T}_{out} \cdot V_{out} \cdot c_w \cdot \rho_w}{\Delta \bar{T}_{in} \cdot V_{in} \cdot c_w \cdot \rho_w} \quad (9)$$

The recovery efficiency (η) is calculated as the amount of extracted heat divided by the amount of injected heat. The amount of energy that is stored is relative (ΔT) to the ambient ground water temperature. T_{amb} is equal to the boundary/initial conditions of the model. Often, when the efficiency has to be calculated from real data (e.g. chapter 5.8), the T_{amb} is not exactly known. In this case the T_{amb} can be derived from the data or has to be measured in the field.

The heat capacity and the density of water are both of influence on the amount of energy that is stored per volume water, and both the heat capacity and the density of water change due to temperature variation. As can be seen in Figure 5.6, the influence of both c_w and ρ_w get more significant at higher temperatures. For the temperatures that we are interested in ($<100^\circ\text{C}$), c_w and ρ_w are taken into account, but don't have a significant influence.

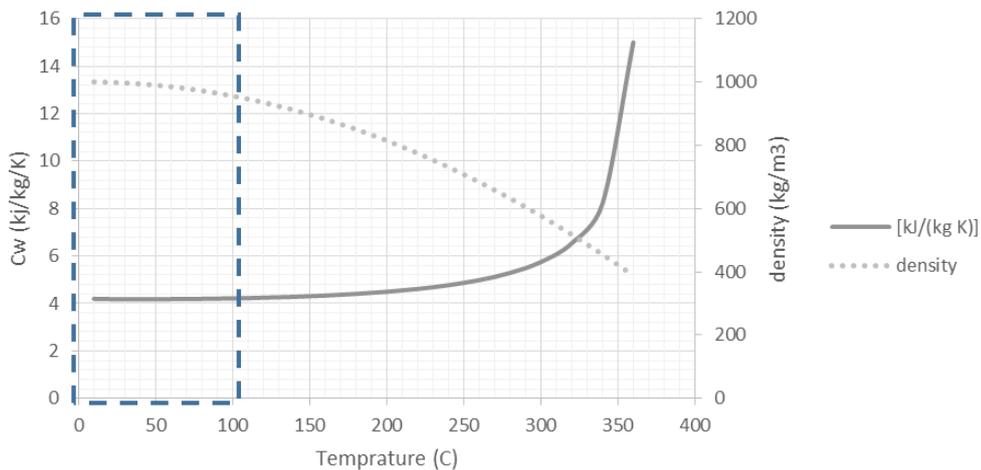


Figure 5.6 Isobaric heat capacity and density relation with increasing temperature. Below 100°C , changes are relatively small. Data from: https://www.engineeringtoolbox.com/specific-heat-capacity-water-d_660.html.

5.7 Dutch pilot site Koppert Cress

For the Koppert Cress site a 3D model was built. Layering was according to local characterization of the subsurface. With this model the history of the ATES system that is being used at Koppert Cress was being simulated to get insight in the current heating/cooling of the subsurface. The input data (daily volume/temperatures) are shown in Figure 5.7. The input data was calculated from hourly data (well temperatures, flow per well) from 2012 to 2019. By this we can compare the model results (well temperature) with the measured well temperature. When calculated efficiencies are lower than real efficiencies, this can indicate that model parameters are not set properly, or, that the grid density has to be increased.



Figure 5.7 Monitoring input data for the Koppert Cress model, during one timestep there is only or injection or extraction.

5.8 IP and code availability

USGS is owner of the SEAWAT models, but it is free to be used by anyone. Flopy is also freely available (<https://modflowpy.github.io/flopydoc/>). The flopy model used for this study is created by the TUDelft and KWR and is the intellectual property of these two organisations (contact: Martin Bloemendal, martin.bloemendal@kwrwater.nl).

5.9 References

Bakker, M., Post, V., Langevin, C.D., Hughes, J.D., White, J.T., Starn, J. J. & Fienen, M. N., 2016. Scripting MODFLOW Model Development Using Python and FloPy. *Ground Water*, 54(5), pp. 733-739

Courant, R., Friedrichs, K. & Lewy, H., 1967. On the partial difference equations of mathematical physics, *IBM Journal of Research and Development*, 11(2), pp. 215–234.

Harbaugh, A. W., Banta, E. R., Hill, M. C. & McDonald, M. G., 2000. *MODFLOW, THE U.S Geological survey modular ground-water model. User guide to Modularization concepts and the ground-water flow process*. U.S.G.S. and U. S. D. o. t. Interior. Reston, Virginia, U.S.G.S: 47.

Hill, M. C., 1990. *PRECONDITIONED CONJUGATE-GRADIENT 2 (PCG2), A COMPUTER PROGRAM FOR SOLVING GROUND-WATER FLOW EQUATIONS, second printing 2003*, U.S. Geological Survey, U.S. Department of the Interior: 31.

Langevin, C., 2009. *SEAWAT: A Computer Program for Simulation of Variable-Density Groundwater Flow and Multi-Species Solute and Heat Transport*. U.S Geological Survey, U.S. Department of the Interior

Langevin, C. D., 2008. Modeling Axisymmetric Flow and Transport. *Ground Water*, 46(4), pp. 579-590.

Langevin, C. D., Thorne, D. T., Dausman, A. M., Sukop, M. C. & Guo, W., 2008. *SEAWAT Version 4: A computer program for simulation of multi-Species Solute and heat transport*. Reston, Virginia, USGS.

Louwyck, A., Vandenbohede, A., Bakker, M. & Lebbe, L., 2014. MODFLOW procedure to simulate axisymmetric flow in radially heterogeneous and layered aquifer systems. *Hydrogeology Journal*, 22(5), pp. 1217-1226.

Thorne, D., Langevin, C. D. & Sukop, M. C., 2006. Addition of simultaneous heat and solute transport and variable fluid viscosity to SEAWAT. *Computers & Geosciences*, 32(10), pp.1758-1768.

van Lopik, J. H., Hartog, N. & Zaadnoordijk, W. J., 2016. The use of salinity contrast for density difference compensation to improve the thermal recovery efficiency in high-temperature aquifer thermal energy storage systems. *Hydrogeology Journal*, 24(5), pp. 1255-1271.

Zheng, C. & Wang, P., 1999. *MT3DMS v5.3. A modular three-dimensional multispecies transport model for simulation of advection, dispersion and chemical reactions of contaminants in groundwater systems. Documentation and User's Guide*, University of Alabama.

Appendix A: input files

CSMP++/Nexus

This simulator input files consists of 4 parts for any single simulation: a CSMP++ configuration file (*-configuration.txt), a CSMP++ regions file (*-regions.txt), a Nexus control file (*-control.txt), and a set of ICEMCFD mesher output files (*.asc + *.dat files). In all CSMP++ and Nexus files any line preceded by a “#” character is a comment line and does not have any effect on the simulator. In the descriptions below, these comment lines are often use as explanations of the significance of each option and how it should be used.

CSMP++ regions file

The regions file from CSMP++ essentially behaves as a filter for the groups of entities defined in the CAD/Mesh-model (e.g. from ICEMCFD). Very often, certain regions of elements can be left out from calculations by simply commenting out their name in this file.

```
geneva02
no properties
BOUNDARY_BACK
BOUNDARY_BOTTOM
BOUNDARY_FRONT
BOUNDARY_LEFT lateral
BOUNDARY_RIGHT lateral
BOUNDARY_TOP
V02_MOLASSE_1
V08_RES_3
V14_JURASSIC_2
GW_1 grwell vtu
#AUXW_1 grwell vtu
#FRACTURE vtu
```

ICEMCFD file example

These files are produced by the Ansys ICEMCFD meshing program. Only the contents of the .asc file is shown here (i.e. truncated, for brevity) for a particular geometry, since the .dat file is in binary format.

```
'./H2_R1_FLAT.asc' generated by ICEMCFD on 27-Jul-19
22 # Number of families
# Objectname      Elementtype      Material-ID      Number of elements
V08_RES_3         TETRA_4          0                22022
V02_MOLASSE_1    TETRA_4          0                2509
V14_JURASSIC_2   TETRA_4          0                2520
_11_BASE_CRET    TRI_3            0                443
BOUNDARY_TOP     TRI_3            0                145
BOUNDARY_BOTTOM TRI_3            0                207
_04_BASE_MOLASSE TRI_3            0                491
BOUNDARY_RIGHT   TRI_3            0                193
BOUNDARY_LEFT    TRI_3            0                186
BOUNDARY_FRONT   TRI_3            0                166
BOUNDARY_BACK    TRI_3            0                197
FRACTURE1        TRI_3            0                115
FRACTURE1        BAR_2            0                31
FRACTURE2        TRI_3            0                100
FRACTURE2        BAR_2            0                28
GEOM              BAR_2            0                178
GW_1              BAR_2            0                41
AUXW_1           BAR_2            0                50
AUXW_2           BAR_2            0                38
AUXW_3           BAR_2            0                43
AUXW_4           BAR_2            0                50
AUXW_5           BAR_2            0                40
# now the elements which make up each object are listed in sequence
V08_RES_3 TETRA_4 22022
0 1 3 4 5 6 7 8 9 10
11 12 13 16 18 19 20 21 23 24
25 28 29 30 31 33 35 37 38 39
40 43 44 45 47 48 49 50 51 52
```

CSMP++ configuration file

The options included in this file populate the model with material and flow properties, as well as boundary conditions.

```
#!/usr/bin/env python geneva02

# (1) global (default) material properties
e density rock 2400.0
e heat source 0
e mass source fluid 0
e permeability 1e-17
e porosity 0.2
e salinity slope 0
e spec heat capacity rock 860.2
e thermal conductivity fluid 0.6
e thermal conductivity rock 2.275
f mass fraction tracer 0
f salt mass fraction bulk 0
m gravity unit vector 0.0 0.0 -1.0
m nxyt 0
m iswit 5
n ground water pressure fluid 0.0
n heat source 0
n heat source initial 0
n integrated heat source 0
n mass fraction tracer 0.0
n mass fraction tracer array 0 0
n mass source fluid 0
n pressure fluid 10000000.0
n salt mass fraction bulk 0.0
n salt mass fraction bulk initial 0.0
n temperature 10.0
n viscosity fluid 0.001
r well damage factor 1.0
r well nearby permeability 1e-12
r well activity switch 0.0
r well bottom hole pressure 0.0
r well casing thermal conductivity 0.5
r well inner radius 0.10795
r well inner temperature 34.0
r well mass fraction tracer 0.0
r well outer radius 0.254
r well surface volume flow rate 0.0
r well volume flow rate 0.0
r well ATES threshold temperature 50.0
reference point 0.0 0.0 0.1
reference pressure fluid 101325.0
reference temperature celsius 10.0

# (2) Regional properties.
#-----
AQUITARD_TOP complete e density rock 2450.0
AQUITARD_TOP complete e permeability 1e-17
AQUITARD_TOP complete e porosity 0.01
AQUITARD_TOP complete e spec heat capacity rock 860.2
AQUITARD_TOP complete e thermal conductivity rock 2.275
#-----
AQUIFER complete e density rock 2670.0
AQUIFER complete e permeability 1e-13
AQUIFER complete e porosity 0.2
AQUIFER complete e spec heat capacity rock 832.91
AQUIFER complete e thermal conductivity rock 2.806
#-----
AQUITARD_BOTTOM complete e density rock 2680
AQUITARD_BOTTOM complete e permeability 1e-17
AQUITARD_BOTTOM complete e porosity 0.01
AQUITARD_BOTTOM complete e spec heat capacity rock 849.894
AQUITARD_BOTTOM complete e thermal conductivity rock 2.692
#-----
#FRACTURE complete e density rock 2670.0
```

```
#FRACTURE complete e permeability 1e-11
#FRACTURE complete e porosity 0.5
#FRACTURE complete e spec heat capacity rock 832.91
#FRACTURE complete e thermal conductivity rock 2.806
#FRACTURE complete thickness 0.1
#-----
GW_1 complete e permeability 1e-22
GW_1 complete e density rock 2670.0
GW_1 complete e porosity 1.0
GW_1 complete e thermal conductivity rock 0.0001
GW_1 complete r well volume flow rate 0.0
GW_1 complete r well mass fraction tracer 1.0
GW_1 complete r well inner temperature 90.0
#-----
#NO AUXILIARY WELLS
#-----

# (3) Essential conditions on regions.
# triggers imposition of either default or local regional properties in a region
#NONE

# (4) Boundary conditions. Apply flag and value to variable on boundary.
#-----
#this is for the gw pressure. (gets added to the existing hydrostatic conditions.
BOUNDARY_LEFT complete Dirichlet n ground water pressure fluid 0.0
BOUNDARY_RIGHT complete Dirichlet n ground water pressure fluid 0.0
#-----
#inflow and outflow of mass
BOUNDARY_LEFT complete INFLOW_OUTFLOW n density fluid 1.0
BOUNDARY_LEFT complete Dirichlet f density fluid 1.0
BOUNDARY_RIGHT complete INFLOW_OUTFLOW n density fluid 1.0
BOUNDARY_RIGHT complete Dirichlet f density fluid 1.0
#-----
#inflow and outflow of heat
BOUNDARY_LEFT complete INFLOW_OUTFLOW n spec enthalpy liquid 100.0
BOUNDARY_LEFT complete Dirichlet f spec enthalpy liquid 100.0
BOUNDARY_RIGHT complete INFLOW_OUTFLOW n spec enthalpy liquid 100.0
BOUNDARY_RIGHT complete Dirichlet f spec enthalpy liquid 100.0
#-----
#for calculation of diffusive heat flux
BOUNDARY_TOP complete Dirichlet n temperature 10.0
BOUNDARY_TOP complete Dirichlet n temperature initial 10.0
BOUNDARY_BOTTOM complete Flux f heat flux -0.064
#-----
#For calculation of transport of tracer
BOUNDARY_LEFT complete Dirichlet n mass fraction tracer 1.0
BOUNDARY_RIGHT complete INFLOW_OUTFLOW n mass fraction tracer 0.0
BOUNDARY_RIGHT complete Dirichlet f mass fraction tracer 0.0
```

Nexus control file

This file extends the functionality of the configuration file by providing essential simulator control features such as formulation, output, restart, monitoring, and algorithm control. Furthermore, a powerful boundary condition control feature expands the basic capabilities of CSMP++.

```
# geneva02 NEXUS control file.
# Do not worry about empty lines, or the vertical order in which parameters are read.

# Simulation End time information (seconds, hours, days, or years)
#end time 3 years
end time 15 years

# VTU frames output to disk in this simulation
# (WARNING: This number could limit your timestep if it is too large!)
#vtu frames 5

# Monitor frames output to disk in this simulation
# (WARNING: This number could limit your timestep if it is too large!)
#monitor frames 10000

# Uncomment the option 'monitor all timesteps' to override the monitoring frames and output
every timestep
# of this simulation. Similar effect is obtained for VTU files with 'output all timesteps'
# WARNING: if left unchecked this could lead to severe storage use!
monitor all timesteps
#output all timesteps
#output every n steps 10
#output simulation steps to screen
#output initialization steps to screen

#the option below should not alter the timestepping at all and simply output as soon as the gap
between outputs
#is at least this much
minimum time between output events 20 days

# The following option sets up the computing time interval between which
restart file output runtime interval 7200
#restart from another binary initial_output
#restart file out at end of sim initial_output

# Simulation Intervals (each interval can have a name or number assigned)
# interval name start time
interval name 1 0.0 days

# Specific output times (both monitor and vtu frames will be output for these times)
# output times unit(secs,days,hours,years) time1 time2 time3...
#output times days 0.9 1.5

# Specific monitor times (only monitor data will be output/appended for these times)
# monitor times unit(secs,days,hours,years) time1 time2 time3...
#monitor times days 0.9 1.5

# VTU output variables (separated by tabs). Uncomment to use.
# By default, only the full model is output.
# Depending on the <vtu> keyword in the regions file,
# these variables will also be output as separate vtu's for those regions
# You may change the order or reduce the list at will.
vtu output n density fluid n pressure fluid n temperature
vtu output e permeability e porosity
vtu output e velocity fluid n mass fraction tracer

#=====
# Monitored variables (separated by tabs). Uncomment to use.
# You may also change the order or reduce the list at will.
# You can request per-model-subdomain (or full model) monitoring via subdomain keywords below.

# VERY IMPORTANT: A region must exist (if created on the fly) before the monitor attempts to
calculate, or else monitor will crash
# In 3D, monitor dimension measures volume, while monitor perimeter measures surface area.
# In 2D, monitor dimension measures area, while monitor perimeter measures contour lengthsurface
area.
```

```
# ALSO IMPORTANT: If 'Model' is entered in the list of subdomains (for ranges, integrals,
single values, etc)
# then all regions or boundaries (depending on variable placement) will be monitored.
#monitor range e compressibility fluid e compressibility rock e density fluid e density rock
e enthalpy source e heat capacity total e heat source e mass source fluid
#monitor range e mass tracer source e permeability e porosity e pressure fluid e
pressure storativity e rock volume source e salinity slope

#monitor integral e compressibility fluid e compressibility rock e density fluid e density
rock e enthalpy source e heat capacity total e heat source e mass source fluid
#monitor integral e mass tracer source e permeability e porosity e pressure fluid e
pressure storativity e rock volume source e salinity slope

monitor value model time model time years model time days timestep cfl
monitor value r well volume flow rate r well injected mass flow rate r well inner
temperature r well activity switch
monitor value r well mass flow through r well volume flow through r well energy flow
through
monitor value r well cumulative mass production r well cumulative volume production
monitor value r well cumulative energy production r well cumulative exergy production r well
cumulative exergy threshold production
monitor value r well cumulative energy injection r well cumulative exergy injection r well
cumulative exergy threshold injection

#subdomain integrals AUX AUX1 AUX2 V01_QUATERNARY V02_MOLASSE_1 V03_RES_1
V04_MOLASSE_2 V05_CRETACEOUS_1
#subdomain dimension AUX AUX1 AUX2 V01_QUATERNARY V02_MOLASSE_1 V03_RES_1
V04_MOLASSE_2 V05_CRETACEOUS_1
#subdomain perimeter AUX AUX1 AUX2 V01_QUATERNARY V02_MOLASSE_1 V03_RES_1
V04_MOLASSE_2 V05_CRETACEOUS_1
subdomain values GW_1 AUXW_1 AUXW_2 AUXW_3 AUXW_4 AUXW_5
subdomain ranges GW_1 AUXW_1 AUXW_2 AUXW_3 AUXW_4 AUXW_5

MONITOR NODE PGW_1 0.0 0.0 -500.0 n pressure fluid n temperature
MONITOR NODE PAUXW_1 -141.4214 0.0 -500.0 n pressure fluid n temperature
MONITOR NODE PAUXW_2 -100.0 100.0 -500.0 n pressure fluid n temperature
MONITOR NODE PAUXW_3 100.0 100.0 -500.0 n pressure fluid n temperature
MONITOR NODE PAUXW_4 100.0 100.0 -500.0 n pressure fluid n temperature
MONITOR NODE PAUXW_5 -100.0 -100.0 -500.0 n pressure fluid n temperature

#CREATE ELEMENT REGION E1 novtu lde 0.0 0.0 -490.0
#subdomain values E1
#subdomain ranges E1
#monitor range e enthalpy source

# The following option allows you to calculate fluxes at boundaries for a particular DGEP
# This is typically independent of the type of equation (i.e. a diffusive flux, or advective
flux is possible)
# A BOUNDARY variable will be created with the suffix ' flux' added to the main solution
variable name
# MONITOR BOUNDARY FLUX sample equation name

# The option below allows the update of pore volumes (particularly for the NCFVT-based classes).
#UPDATE PORE VOLUMES

#-----
#time dependent conditions could be at the boundary, region, or point (closest nodal location)
# values needed are 4. a starting value, and end value, a starting time, and an end time.
# values (of the variable) can be formula strings (e.g a number "4.0" or "4.0*x+y*t")
# where x,y,z, and t are possible variables.
# example below
#initial condition boundary BOUNDARY_TOP n temperature 10+0.005175983*(x+466.5)
DIRICH

#GW_1 WELL CONDITIONS
#charging
```

```

time dependent condition      region GW_1  r well volume flow rate 0.0 days 1.0 days
0.06*(6*(t)^5-15*(t)^4+10*(t)^3) DIRICH cycle 15 365.25 days
time dependent condition      region GW_1  r well activity switch 0.0 days 1.0 days 1.0
DIRICH cycle 15 365.25 days
time dependent condition      region GW_1  r well volume flow rate 1.0 days 119.0 days
0.06 DIRICH cycle 15 365.25 days
time dependent condition      region GW_1  r well activity switch 1.0 days 119.0 days
1.0 DIRICH cycle 15 365.25 days
time dependent condition      region GW_1  r well volume flow rate 119.0 days 120.0
days -0.06*(6*(t-119)^5-15*(t-119)^4+10*(t-119)^3)+0.06 DIRICH cycle 15 365.25 days
time dependent condition      region GW_1  r well activity switch 119.0 days 120.0
days 1.0 DIRICH cycle 15 365.25 days
#storage
time dependent condition      region GW_1  r well volume flow rate 120.0 days 180.0
days 0.0 DIRICH cycle 15 365.25 days
time dependent condition      region GW_1  r well activity switch 120.0 days 180.0
days 0.0 DIRICH cycle 15 365.25 days
#discharging
time dependent condition      region GW_1  r well volume flow rate 180.0 days 181.0
days -0.06*(6*(t-180)^5-15*(t-180)^4+10*(t-180)^3) DIRICH cycle 15 365.25 days
time dependent condition      region GW_1  r well activity switch 180.0 days 181.0
days -1.0 DIRICH cycle 15 365.25 days
time dependent condition      region GW_1  r well volume flow rate 181.0 days 299.0
days -0.06 DIRICH cycle 15 365.25 days
time dependent condition      region GW_1  r well activity switch 181.0 days 299.0
days -1.0 DIRICH cycle 15 365.25 days
time dependent condition      region GW_1  r well volume flow rate 299.0 days 300.0
days 0.06*(6*(t-299)^5-15*(t-299)^4+10*(t-299)^3)-0.06 DIRICH cycle 15 365.25 days
time dependent condition      region GW_1  r well activity switch 299.0 days 300.0
days -1.0 DIRICH cycle 15 365.25 days
#storage
time dependent condition      region GW_1  r well volume flow rate 300.0 days 365.25
days 0.0 DIRICH cycle 15 365.25 days
time dependent condition      region GW_1  r well activity switch 300.0 days 365.25
days 0.0 DIRICH cycle 15 365.25 days

#AUXILIARY WELL CONDITIONS
#NONE FOR THIS SCENARIO

#-----
# setup of loop exit criteria (i.e. for non linear loops)
# loop name(from code!) criteria type max iterations abs. tolerance variable1 to use in
criteria variable2 to use in criteria unique name identifier for criteria visitor
#example below
#set group exit criteria initial pressure loop L2 100 1.e-9 n pressure fluid n
pressure fluid NL_k init_p
set group exit criteria initial pressure loop L2 10 1.e-9 n pressure fluid n pressure
fluid NL_k init_p
set group exit criteria determine timestep loop INF 1 0.01 timestep future timestep sim
set group exit criteria main non linear loop L2 1 1.e-8 n pressure fluid n pressure
fluid NL_k sim_p

#=====
# THC Simulator control section
#-----
mass based cfl equations advection density fluid
with maximum timestep 3600.0
input timestep safety factor 0.6
with increasing initial safety factor 0.3
timestep change factor 0.05

#=====
#options for the pressure equation
#-----
#with cvfe pressure diffusion
#with cvfe implicit pressure

```

```
with new hydrostatic lateral boundaries

#####
#options for the temperature equation
#-----
#isothermal
with implicit temperature diffusion
with initial temperature solution
#with cvfe implicit temperature
#with cvfe temperature diffusion
#with kinetic energy
#with energy correction
#with joule thomson
#with initial groundwater energy

#####
#options for calculating ATEs thermal efficiency
thermal efficiency ates cycle length 365.25 days

#####
#options for the salinity equation
#-----
#with salinity
#with initial salinity solution

#####
#options for tracer advection
#-----
with tracer advection
#with tracer array advection
```

Moose

A summarized input file is shown below for the base-case scenario in Birdsell and Saar (2020). Comments that add readability follow hash marks, and ellipses show portions of code that have been removed for brevity.

```
#Example of specification of rock regions
```

```
[MeshModifiers]
```

```
[./reef_aquifer]
```

```
type = SubdomainBoundingBox
```

```
block_id = 2
```

```
bottom_left = '20 -200 20'
```

```
top_right = '2000 -150 2000'
```

```
[./]
```

```
...
```

```
[]
```

```
## Boundary conditions
```

```
[BCs]
```

```
[./confinex]
```

```
type = PresetBC
```

```
variable = disp_x
```

```
value = 0
```

```
boundary = 'left'
```

```
[./]
```

```
[./confinez]
```

```
type = PresetBC
```

```
variable = disp_z
```

```
value = 0
```

```
boundary = 'back'
```

```
[./]
```

```
[./right]
```

```
type = DirichletBC
```

```
variable = porepressure
```

```
value = 0.0e6
```

```
boundary = right
```

```
[./]
```

```
[./front]
```

```
type = DirichletBC
```

```
variable = porepressure
```

```
value = 0.0e6
```

```
boundary = front
```

```
[./]
```

```
[./topload]
```

```
type = NeumannBC
```

```
variable = disp_y
value = -0.
boundary = top
[./]
[./bottom_confine]
type = PresetBC
variable = disp_y
value = 0
boundary = 'bottom'
[./]
[]

## This is the injection well
[DiracKernels]
[./pls]
type = PorousFlowPolyLineSink
fluid_phase = 0
point_file = loweraquifer.bh
SumQuantityUO = pls_total_outflow_mass
variable = porepressure
p_or_t_vals = '-1E15 1E15'
fluxes = '-0.25 -0.25' #Flux (kg/m/s), which is weighted by total line length. this is like PiecewiseLinearSink. If set to same value, you get same flux.
[./]
[]

## The Equation of State
[Modules]
[./FluidProperties]
[./the_simple_fluid]
type = SimpleFluidProperties
thermal_expansion = 0.0
bulk_modulus = 2.e9
viscosity = 0.001
density0 = 1000.
porepressure_coefficient = 0.0
[./]
[./]
[]

## Material properties
[Materials]
[./elasticity_tensor]
type = ComputeIsotropicElasticityTensor
```

```
youngs_modulus = 50.e9 # this specifies the youngs modulus
poissons_ratio = 0.3 # this specifies the poissons ratio
[./]
[./strain]
type = ComputeSmallStrain
[./]
[./stress]
type = ComputeLinearElasticStress
[./]
...
[./porosity_reef_aquifer]
type = PorousFlowPorosity
mechanical = true
ensure_positive = true
block = reef_aquifer
porosity_zero = 0.15
biot_coefficient = 1.0
[./]
[./biot_modulus]
type = PorousFlowConstantBiotModulus
biot_coefficient = 1.0
fluid_bulk_modulus = 2.e9
solid_bulk_compliance = 0.25
[./]
...
[./permeability_reef_aquifer]
type = PorousFlowPermeabilityConst
block = reef_aquifer
permeability = '1E-15 0 0 0 1E-15 0 0 0 1E-15'
[./]
...
[]

## Solver options
[Preconditioning]
[./preferred_but_might_not_be_installed]
type = SMP
full = true
petsc_options_iname = '-pc_type -pc_factor_mat_solver_package'
petsc_options_value = 'lu mumps'
[./]
[]
[Executioner]
type = Transient
```

```
solve_type = Newton
dt = 0.5e6
end_time = 1.866e7 # seconds to 216 days
l_max_its = 30
l_tol = 1e-10
# controls for nonlinear iterations
nl_max_its = 20
nl_rel_tol = 1e-8
nl_abs_tol = 1e-12
[]

## Output options
[Outputs]
perf_graph = true
exodus = true
[./csv]
type = CSV
[./]
[]

[VectorPostprocessors]
[./pressure]
type = SideValueSampler
variable = porepressure
sort_by = x
execute_on = timestep_end
boundary = back
[./]
[]
```

CODE_BRIGHT and RETRASO

Code_Bright needs two input files (which can be prepared by the GID pre and postprocessor, Figure A.1):

root_gen.dat: It contains general information, options, parameter values for materials and boundary conditions.

root_gri.dat: It contains the finite element grid.

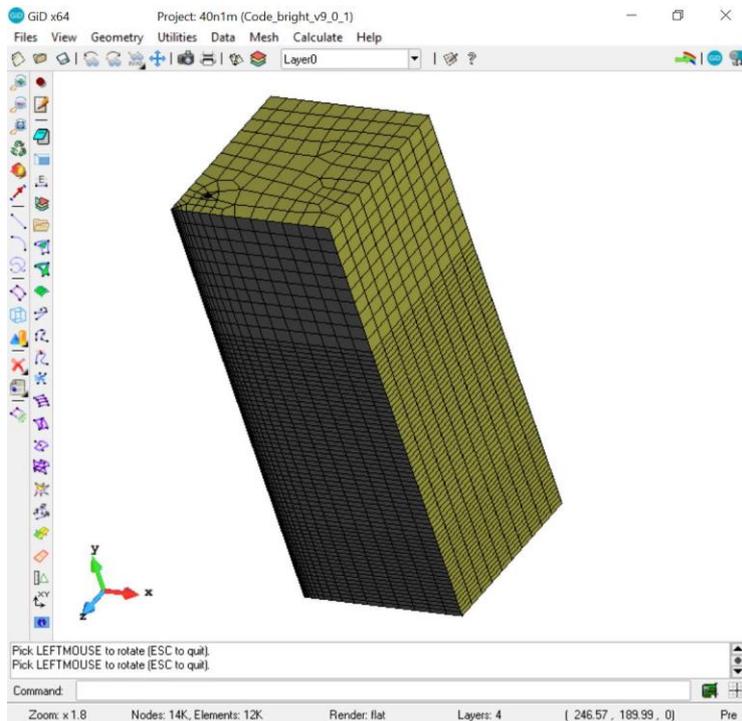


Figure A.1 Snapshot of the GID pre-processor.

The same files are required by Retraso. In addition, it requires the following files:

root_che.inp: It contains the chemical characterization of the reactive transport problem. Figure A.2 gives an example.

root_trn.inp: It contains the transport properties of the reactive transport problem.

mastertemp.dat: It is a thermodynamic database

kinetics.dat: It is a database of kinetic rate laws.

```
1 'TITLE OF THE PROBLEM : Component problem 1'
2 '-----'
3 'DEFINITION OF THE GEOCHEMICAL SYSTEM'
4 15. !initial temp. of the system
5 'PRIMARY AQUEOUS SPECIES'
6 'SiO2 (aq) '
7 'H+'
8 'Na+'
9 'Cl-'
10 'Ca++'
11 'Mg++'
12 'K+'
13 'Al (OH) 2+'
14 'HCO3-'
15 'SO4--'
16 '*'
17 'AQUEOUS COMPLEXES'
18 'OH-'
19 'CO2 (aq) -45bar'
20 'CO3---45bar'
21 'H3SiO4-'
```

Figure A.2 Snapshot of the asci file `root_che.inp`, necessary for Retraso.

SEAWAT

Example input files are shown here below.

SEAWATv4 input files and corresponding package

Parameter	Value	Package
Solid heat capacity*	710 kJ/kg °C	RCT
Water reference density	1,000 kg/m ³	RCT
Solid density*	2,640 kg/m ³	RCT
Water thermal conductivity	0.58 W/m/°C	RCT
Solid thermal conductivity	2.55 W/m/°C	RCT
Thermal distribution coefficient [#]	1.7 · 10 ⁻⁴ m ³ /kg	RCT
Thermal retardation ⁺	2.21	RCT
Porosity	0.3	BTN
Specific storage aquifer	6 · 10 ⁻⁴ /m	LPF
Longitudinal dispersion	0.5 m	DSP
Transversal dispersion	0.05 m	DSP
Vertical dispersion	0.005 m	DSP
Effective molecular diffusion heat [#]	0.15 m ² /day	DSP
Effective molecular diffusion salt	8.64 · 10 ⁻⁶ m ² /day	DSP

Layering input file

Layer	zTop	zBot	Kh	Kv	por	ss	s1	s2	type
1	0	-40	0.05	0.01	0.3	1.00E-05	12	12	0 aquitard
2	-40	-90	35	7	0.3	1.00E-05	12	12	0 aquifer
3	-90	-130	0.05	0.01	0.3	1.00E-05	12	12	0 aquitard

Well(s) input file

xcor	ycor	parent	SystemP air	YrStart	IsCold	T_inj	QdMax	Qy	IsMono
0.0	0.0	0	0	2005	0	90	4,545	500,000	1

Grid information

- dmin = 1 #smallest cell size at well [m] (1)
- dmin_bound = 100 #total boundary length with dmin cells.
- dmax = 100 #largest cell size at boundary [m]
- dz = 0.5 # gridlayer thickness [m] !! synchronize with layer tops in xls file
- aroundAll = 2500 #size of grid around well.
- nstep = 40 #minimum number of steps
- grid_extents #[-300,300,-300,300]