

# **IDEA** League

MASTER OF SCIENCE IN APPLIED GEOPHYSICS

RESEARCH THESIS

---

## **Optimization of heat extraction within sedimentary reservoirs for CO<sub>2</sub> Plume Geothermal (CPG) electricity generation.**

**Marat Ravilov**

---

August 9, 2019



# **Optimization of heat extraction within sedimentary reservoirs for CO<sub>2</sub> Plume Geothermal (CPG) electricity generation.**

MASTER OF SCIENCE THESIS

for the degree of Master of Science in Applied Geophysics

by

Marat Ravilov

August 9, 2019



IDEA LEAGUE  
JOINT MASTER'S IN APPLIED GEOPHYSICS

Delft University of Technology, The Netherlands  
ETH Zürich, Switzerland  
RWTH Aachen, Germany

Dated: *August 9, 2019*

Supervisor(s):

---

Benjamin Adams

---

Martin Saar

Committee Members:

---

Benjamin Adams

---

Martin Saar

---

Florian Wellmann



---

# Abstract

The primary goal of the present work is evaluation and comparison of vertical and horizontal well placements and their impact on the power output of a CPG (CO<sub>2</sub> Plume Geothermal) system. Performances of vertical and horizontal wells are evaluated for a repeated five-spot pattern. Six numerical models were developed in MOOSE (Multiphysics Object Oriented Simulation Environment), tested and compared against each other and a benchmark study. Simulations show that the reservoir will respond differently to different well configurations, with buoyancy playing a major role in its response. The study concludes with recommendations on ways to improve the accuracy of present models and directions for future research on optimal well placement for CPG systems.





---

# Acknowledgements

I am very grateful for the opportunity to work on this interesting and challenging topic in a friendly environment. I would like to thank all Geofluids group members at ETH and in particular Anozie and Phil whose contributions were invaluable. My deepest gratitude to Ben, my supervisor, for his guidance, involvement and willingness to support me in my work.

ETH Zurich  
August 9, 2019

Marat Ravilov



---

# Table of Contents

<b>Abstract</b>	<b>v</b>
<b>Acknowledgements</b>	<b>vii</b>
<b>Nomenclature</b>	<b>xv</b>
<b>Acronyms</b>	<b>xvii</b>
<b>1 Introduction</b>	<b>1</b>
1-1 CPG and Reservoir Simulation . . . . .	1
1-1-1 Five-spot well pattern . . . . .	2
1-2 Reservoir Impedance . . . . .	2
<b>2 Theory and numerical implementation</b>	<b>5</b>
2-1 Fluid flow . . . . .	5
2-1-1 Darcy's Law . . . . .	6
2-2 Heat flux . . . . .	6
2-3 Overview of the FEM and MOOSE . . . . .	6
2-4 Borehole Implementation . . . . .	7
<b>3 Method</b>	<b>9</b>
3-1 List of models for the present study . . . . .	9
3-2 Input parameters . . . . .	9
3-3 Input meshes . . . . .	11

<b>4</b>	<b>Model validation</b>	<b>13</b>
4-1	Comparing with the benchmark Randolph and Saar model . . . . .	13
4-1-1	Benchmark model . . . . .	13
4-1-2	Base case model (Model A) . . . . .	14
4-1-3	Comparison . . . . .	14
4-2	Analytical solution . . . . .	17
4-3	Mesh sensitivity . . . . .	19
4-4	Implementing boreholes . . . . .	19
4-5	Summary . . . . .	20
<b>5</b>	<b>Single-phase models</b>	<b>23</b>
5-1	Results . . . . .	23
5-2	Discussion . . . . .	24
<b>6</b>	<b>Multi-phase models</b>	<b>29</b>
6-1	Results . . . . .	29
6-2	Discussion . . . . .	31
<b>7</b>	<b>Summary</b>	<b>33</b>
7-1	Areas for future research on optimization of heat extraction for CPG . . . . .	33
7-2	Ways to improve numerical models for present study . . . . .	34
	<b>Bibliography</b>	<b>37</b>
<b>A</b>	<b>Numerical code for the multiphase model</b>	<b>39</b>
A-1	Input file . . . . .	39
<b>B</b>	<b>Numerical code for the single-phase models</b>	<b>45</b>
B-1	Input file . . . . .	45
	<b>Bibliography</b>	<b>49</b>

---

# List of Figures

1-1	5-spot well configuration . . . . .	2
1-2	$P_e$ output and mass flowrate vs reservoir impedance. . . . .	3
1-3	Schematic representation of a CPG system. . . . .	4
3-1	Mesh 1: refined around the vertical boreholes. . . . .	12
3-2	Mesh 2: refined around vertical injector and horizontal producer. . . . .	12
4-1	Mesh comparison: Model A (left) vs Benchmark (right). Injector in red, producer in blue. Refinement around the boreholes for the Model A not shown . . . . .	14
4-2	Comparison with the Benchmark: pressure-temperature plots. . . . .	15
4-3	Comparison with the Benchmark: $P_{th}$ and RI . . . . .	16
4-4	Pressure-temperature plots for matched $E_{th}$ . . . . .	16
4-5	Model A: streamlines and porepressure contours. Z-plane slice. Streamlines in white, pressure contours in gray.After 1 month of simulation . . . . .	17
4-6	Model A: streamlines and porepressure contours. Z-plane slice. Streamlines in white, pressure contours in gray.After 25years of simulation . . . . .	17
4-7	Problem domain for 1D analytical five-spot. . . . .	18
4-8	Mesh sensitivity: linear plot. . . . .	20
4-9	Mesh sensitivity: semi logarithmic plot. . . . .	20
4-10	Peaceman correction factor . . . . .	21
5-1	Models A and B: reservoir impedance vs time. . . . .	24
5-2	Models C and D: reservoir impedance vs time, including initial transient stage. . . . .	25
5-3	Models C and D: reservoir impedance vs time, showing quasi-steady state portion of the curve. . . . .	26
5-4	Fluid flow patterns for Model C after 25 years of simulation . . . . .	26
5-5	Fluid flow patterns for Model D after 25 years of simulation . . . . .	27

6-1	Brine saturated reservoir at gravitational equilibrium. Used as brine porepressure IC for the models E and F. . . . .	30
6-2	Models E and F: reservoir impedance vs time. . . . .	30
6-3	Temperature distribution in Model E after 25 years of simulation. . . . .	31
7-1	All models: reservoir impedance vs time. . . . .	34

---

# List of Tables

3-1	List of models. . . . .	9
3-2	Reservoir parameters. . . . .	10
3-3	Boundary conditions . . . . .	10
3-4	Injection and production parameters. . . . .	10
3-5	BH injection and production pressures for models with buoyancy. . . . .	11
3-6	Reservoir fluid parameters. . . . .	11
4-1	Performance of a 5-spot system: Benchmark vs Model A. All values given after 25 years of simulation. . . . .	15
4-2	1D analytical solution for $m$ and RI . . . . .	19
5-1	Summary of results: single-phase models. . . . .	24
6-1	Summary of results: multi-phase models. . . . .	29
7-1	Summary of results: all models. . . . .	33





Unless otherwise specified for a particular case, the following notations were used throughout the text:

$p$  : pressure

$T$ : temperature

$P_{th}$ : heat extraction rate (thermal power output)

$P_e$ : electrical power output

$E_{th}$ : thermal energy

$\dot{m}$  : mass flow rate

$\mu$  : (dynamic) viscosity

$k$ : permeability

$k_r$  : relative permeability

$\rho$ : mass density

$\lambda$  thermal conductivity

$A$ : cross-sectional area

$h$ : height or thickness



---

# Acronyms

**BC** Boundary Condition

**BH** Bottom hole

**CPG** CO<sub>2</sub> Plume Geothermal

**IC** Initial Condition

**FEM** Finite Element Method

**MOOSE** Multiphysics Object Oriented Simulation Environment

**RI** Reservoir Impedance

**Pa** Pascals (pressure units)

**K** Kelvin (temperature units)

**kPa** Pa×10<sup>3</sup>

**MPa** Pa×10<sup>6</sup>



---

# Chapter 1

---

## Introduction

Geothermal power plants use the temperature difference between the Earth's hot subsurface rock and the cooler surface to generate electricity. Such systems transport thermal energy from underground to the surface using a working fluid, and in a power plant at the surface, a portion of the fluid's thermal energy is converted into electricity. The cooled working fluid is then typically reinjected into a subsurface reservoir [B. M. Adams, 2014a]. CO<sub>2</sub> has been proposed as the subsurface working fluid for geothermal energy extraction as it has advantages over brine: high mobility, low solubility of amorphous silica, and higher density sensitivity to temperature [B. M. Adams, 2014b]. Geothermal systems that use CO<sub>2</sub> as a working fluid in naturally permeable, porous reservoirs are termed CPG (CO<sub>2</sub> Plume Geothermal) systems. Naturally permeable reservoirs do not need artificial permeability enhancement to maintain flowrates sufficient for power plant operation. This makes possible an economically efficient extraction of heat energy by CPG. Previous studies and numerical models had suggested that geologic reservoirs with CO<sub>2</sub> as the heat mining fluid would be viable geothermal energy sources for power production for decades, potentially even in regions with relatively low geothermal temperatures and heat flow rates. While the focus of the previous studies was on principal validation of CPG technology, the goal of the present study is to examine the effect of different well placements for CPG electricity generation using thoroughly validated numerical model of the subsurface portion of CPG system.

### 1-1 CPG and Reservoir Simulation

Figure 1-3 shows a schematic representation of a CPG system. It can be subdivided into separate subdomains: reservoir and surface plant coupled through boreholes. Reservoir plays a key role in CPG plant performance, setting a limit to its power output. Simulations of CPG power output significantly vary depending on reservoir parameters and injection-production conditions [Randolph and Saar, 2011b].

### 1-1-1 Five-spot well pattern

Each 5-spot system configuration involves four major components: a reservoir, an injection well, four production wells, and a surface power plant [Randolph and Saar, 2011a]. Figure 1-1 shows the five-spot configuration, reproduced from [Randolph and Saar, 2010]. The symmetry of the five-spot computational grid reduces modeling requirements to 1/8th of the system domain. Models in present study were run on 1/4th of the domain, due to better representation of the wellbore vicinity (more adjacent nodes connecting to the borehole nodes).

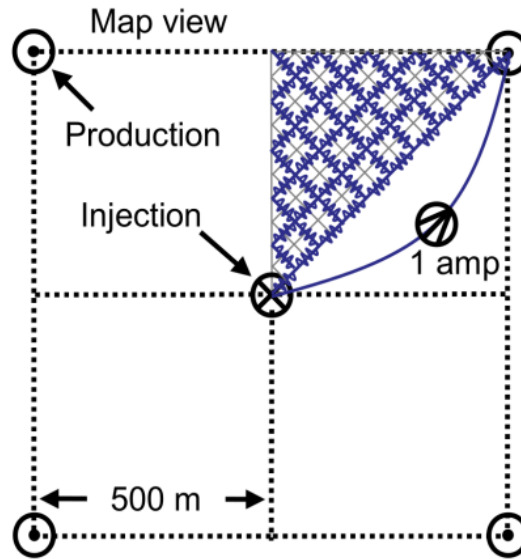


Figure 1-1: 5-spot well configuration

## 1-2 Reservoir Impedance

The state of the fluid produced from the reservoir depends on several factors, including reservoir depth, temperature, permeability, fluid mass flowrate, injection pressure, and injection temperature [B. M. Adams, 2014a]. These factors are determined by the system as a whole. For a given reservoir at a fixed temperature and depth, the pressure change between injection and production wells tends to vary primarily with mass flowrates. These two parameters in turn will determine the operation of the surface plant. To describe the system behaviour as a whole, a reservoir impedance parameter (RI) is introduced. It is defined as:

$$RI = \frac{\Delta P}{\dot{m}} \quad (1-1)$$

where  $\Delta P$  (measured in kPa) is the pressure loss occurring between the injection and production well and  $\dot{m}$  is mass flowrate (measured in  $\frac{kg}{s}$ ). RI represents drainage efficiency of a reservoir, this in turn controls the power output of a CPG system. For given fluid properties, an optimal flow rate at the surface power plant can be established that maximizes its power output. Assuming that increase in  $\Delta P$  leads to a proportional increase in  $\dot{m}$  (i.e. ignoring

dependence of fluid properties on pressure), we see that achieving the optimal mass flow rates at the surface is a trivial matter of setting required  $\Delta P$  and for any given value of RI we can find a corresponding value of  $P_e$  of a surface power plant operating at optimal conditions Figure 1-2. Thus we do not need to parameterize mass flowrate, but instead operate at mass flowrates which provide maximum power generation [B. M. Adams, 2014a]. The maximized  $P_e$  found in this way is given for all simulated models in the corresponding chapters.

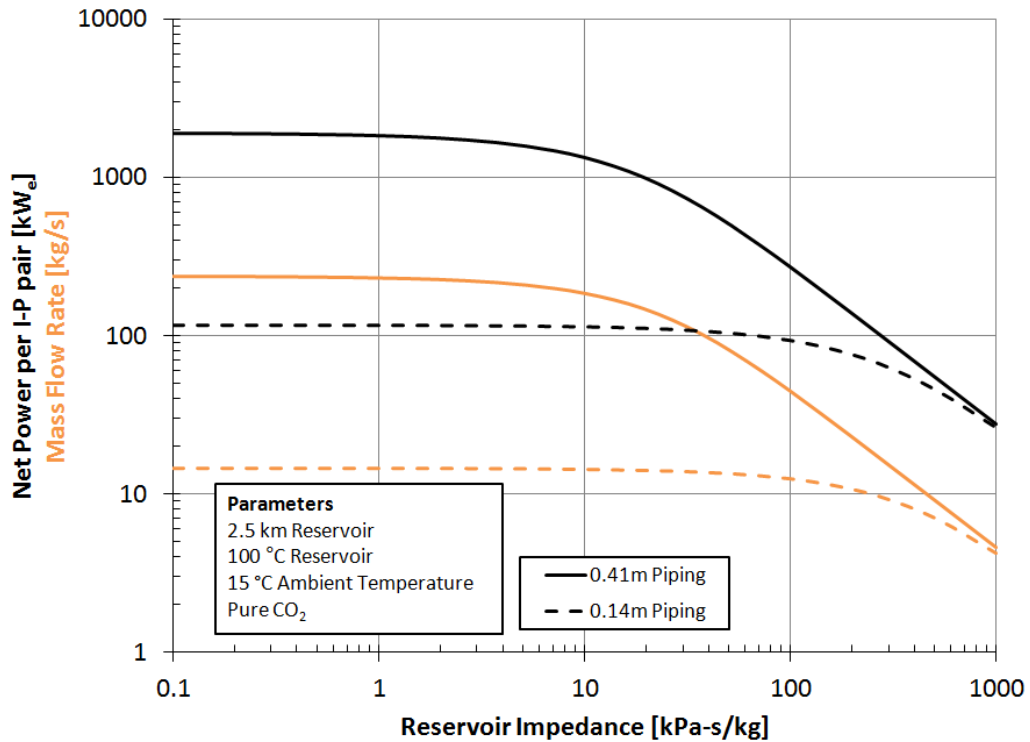
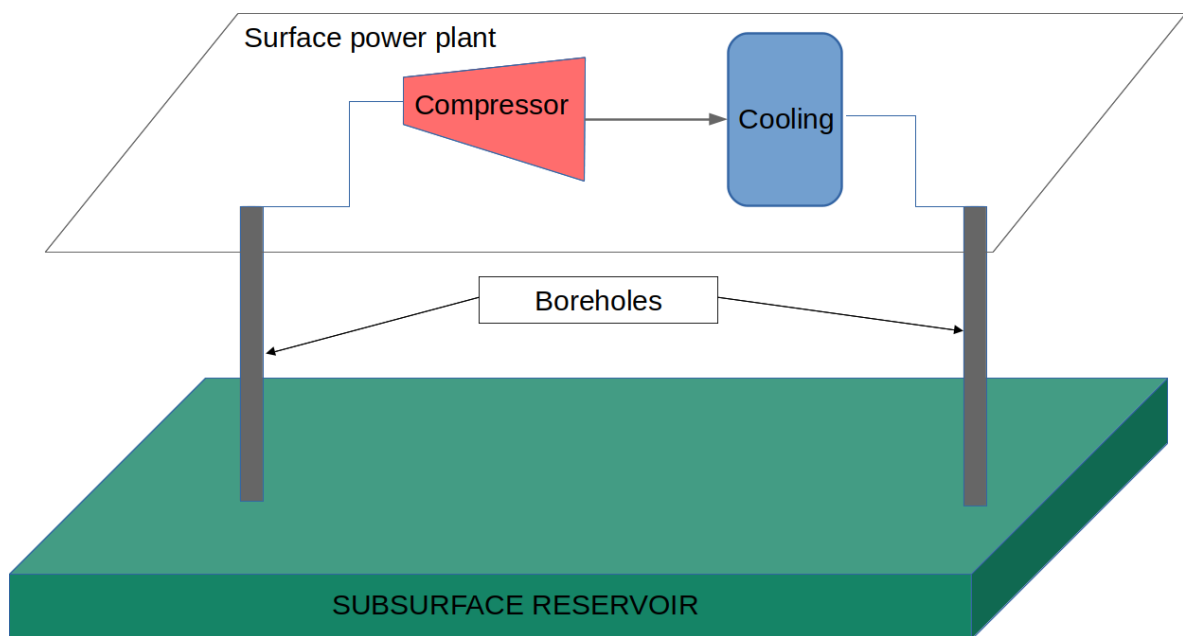


Figure 1-2:  $P_e$  output and mass flowrate vs reservoir impedance.



**Figure 1-3:** Schematic representation of a CPG system.



# Theory and numerical implementation

The equations for heat flow, fluid flow, solid mechanics, and chemical reactions are defined here. The final section provides an overview of the simulator used for this study. In present work the fluid flow was modelled for the following cases:

- 1) Fully saturated single-phase flow.
- 2) Two phase, two components flow. The liquid phase is filled with brine only, while the gas phase is filled with CO<sub>2</sub> only. The fluids considered to be immiscible. There is capillarity between the brine and CO<sub>2</sub>.

In both cases heat is advected with the fluids as well as conducted through the fluid-rock system. The density  $\rho_\beta$  was computed as a function of pressure and temperature.

## 2-1 Fluid flow

Mass conservation for fluid species  $\kappa$  is described by the continuity equation Eq. (2-1) (radioactive decay and chemical precipitation / dissolution are omitted):

$$\frac{\partial M^\kappa}{\partial t} + M^\kappa \nabla \cdot \mathbf{v}_s + \nabla \cdot \mathbf{F}^\kappa + q^\kappa = 0 \quad (2-1)$$

An equivalent formulation for the energy conservation is obtained by replacing  $M^{kappa}$  in the equation above by the heat energy density .

Here  $M$  is the mass of fluid per bulk volume (measured in  $\text{kg} \cdot \text{m}^{-3}$ ),  $\mathbf{v}_s$  is the velocity of the porous solid skeleton (measured in  $\text{m} \cdot \text{s}^{-1}$ , ignored in present work),  $\mathbf{F}$  is the flux vector (a vector, measured  $\text{kg} \cdot \text{s}^{-1} \cdot \text{m}^{-2}$  and  $q$  is a source measured in  $\text{kg} \cdot \text{m}^{-3} \cdot \text{s}^{-1}$ ).

The coupling to the solid mechanics is via the  $M^\kappa \nabla \cdot \mathbf{v}_s$  term, as well as via changes in porosity and permeability. Coupling to heat flow and chemical reactions is via the equations of state used within the terms of Eq. (2-1) as well as the source term  $q^\kappa$ .

The species are parameterised by  $\kappa = 1, \dots$ , where each specie cannot be decomposed into

other species, but can change phase. The mass of species  $\kappa$  per volume of rock is written as a sum over all phases present in the system (absorption term is omitted) Eq. (2-2):

$$M^\kappa = \phi \sum_{\beta} S_{\beta} \rho_{\beta} \chi_{\beta} \quad (2-2)$$

The solid's porosity is  $\phi$ ,  $S_{\beta}$  is the saturation of phase  $\beta$ ,  $\rho_{\beta}$  is the density of phase  $\beta$ .

### 2-1-1 Darcy's Law

The flux is a sum of advective flux and diffusive-and-dispersive flux:

$$\mathbf{F}^{\kappa} = \sum_{\beta} \chi_{\beta}^{\kappa} \mathbf{F}_{\beta}^{advective} + \mathbf{F}^{\kappa}_{diffusion+dispersion} \quad (2-3)$$

Advective flux is governed by Darcy's law. Each phase is assumed to obey Darcy's law. Each phase has its own density,  $\rho_{\beta}$ , relative permeability  $k_{r,\beta}$ , viscosity  $\mu_{\beta}$  and pressure  $P_{\beta}$ . These may all be nonlinear functions of the independent variables. With them, we can form the advective Darcy flux:

$$\mathbf{F}_{\beta}^{advective} = \rho_{\beta} * \mathbf{v}_{\beta} = -\rho_{\beta} \frac{k k_{r,\beta}}{\mu_{\beta}} (\nabla P - \rho \mathbf{g}) \quad (2-4)$$

Diffusive and dispersive fluxes in the equation above are ignored in this study.

## 2-2 Heat flux

The heat flux is a sum of heat conduction and convection with the fluid ((2-5)):

$$\mathbf{F}_{\beta}^T = -\lambda \nabla T + \sum_{\beta} h_{\beta} \mathbf{F}_{\beta} \quad (2-5)$$

Where  $\lambda$  is the thermal conductivity,  $h_{\beta}$  is the specific enthalpy of phase  $\beta$ , is the advective Darcy flux [INL, 2018].

## 2-3 Overview of the FEM and MOOSE

Reliable reservoir performance predictions of geothermal systems require accurate modeling for the coupled thermal-hydrological processes. MOOSE (Multiphysics Object Oriented Simulation Environment) is an object-oriented C++ finite element framework for the development of tightly coupled multiphysics solvers. The simulator makes use of a number of freely distributed physics libraries, including for the solution of transport problems in porous media. The PorousFlow module is a library of physics for fluid and heat flow in porous media. It is formulated in a general manner, and is capable of solving problems with an arbitrary number of phases and fluid components. MOOSE is very well documented and further details on its capabilities and implementation details can be looked up from its online documentation [INL, 2018].

**Finite Element Method** In the FEM, the domain is discretized into a mesh of finite elements of any shape, such as hexahedral elements used in present work. The PDEs describing physical processes are not solved directly but replaced by a variational formulation called a weak form. At each node of an element, the values of the pressure, temperature and saturation are treated as the unknowns (nodal values). Details on the FEM solution and discretization process with a focus on the groundwater flow problems can be consulted in [Kukreti and Rajapaksa, 1989], [Diersch, 2018], [Bear, 2010].

## 2-4 Borehole Implementation

Wellbores are implemented using the method first described by Peaceman [Peaceman., 1983]. They are represented in the simulator as a collection of nodal point sources arranged along a line representing a borehole. The flow rate is defined in terms of the pressure at a point at the wall of the wellbore, as in the equation below:

$$f(p_i, x_i) = WC \frac{k_r \rho}{\mu} (p_i - p_{wellbore}) \quad (2-6)$$

Where  $C = 1$  for producer and  $C = -1$  for injector well.  $W$  is called the well constant. Peaceman showed that the computed block pressure is associated with the steady-state pressure for the actual well at an equivalent radius  $r_e$ . It enters the well constant term, which is defined as as:

$$W = \frac{2\pi \sqrt{k_{xx} * k_{yy}} L_z}{\ln(r_e/r_{bh})} \quad (2-7)$$

The value of  $r_e$  will depend on mesh geometry [Chen and Zhang., 2009]. For models with buoyancy, pressure at the bottom was fixed and the corresponding pressure at the top calculated by simulator using average weight of the fluid column.



---

# Chapter 3

---

## Method

Simulations are conducted to study the performance of a five-spot system operating using either vertical or horizontal producers. Each case was simulated for a) vertical injector and producer and b) vertical injector and horizontal producer pair, operating at constant bottom-hole pressure. Neumann BC for pressure, saturation and temperature variables were imposed on every side of the simulation domain.

### 3-1 List of models for the present study

In total six model problems were simulated that either differ in terms of their well configuration (horizontal or vertical producer) or in terms of modelled physics (buoyancy forces, single and multiphase flow). The table below a complete list with description. Elsewhere in this report I refer to the models by their abbreviations (model A, B, etc.).

Abbreviation	Model Description
Model A	Single-phase, vertical producer, no buoyancy
Model B	Single-phase, horiz producer, no buoyancy
Model C	Single-phase, vertical producer, with buoyancy
Model D	Single-phase, horiz producer, with buoyancy
Model E	Multi-phase, vertical producer, with buoyancy
Model F	Multi-phase, horiz producer, with buoyancy

**Table 3-1:** List of models.

### 3-2 Input parameters

For all models the reservoir is located at 2.5 km depth and is represented by a porous rock initially fully saturated by CO<sub>2</sub>. Temperature was calculated for a constant geothermal

gradient of 35 C/km the initial pressure is equal to hydrostatic for that depth. Models A, B, C and D use uniform IC for porepressure. Models E and F use IC for porepressure at gravitational equilibrium. Input parameters for all models are given below.

Parameter	Unit	Value
Depth to the top of the reservoir	m	2500
Initial porepressure (at 2500 m)	MPa	25
Initial temperature	K	373
Permeability	$m^2$	$5 \times 10^{-14}$
Thermal conductivity	$W/m/K$	2.1
Porosity	(decimals)	0.2
Rock specific heat	$J/kg/K$	1000
Lateral dimensions	m	500 x 500
Thickness	m	305

**Table 3-2:** Reservoir parameters.

Boundary	Condition
Sides	No fluid or heat flow
Top and bottom	No fluid flow, heat conduction

**Table 3-3:** Boundary conditions

To ease a comparison between models with horizontal and vertical producers the BHP was set so that for CO<sub>2</sub> filled boreholes corresponding pressures at 2500 m were equal to the values given in the table 3-4.

Multiphase models were run with the fluid parameters listed in table 3-6. Multiphase flow was modelled as unsaturated 2-phase 2-component CO<sub>2</sub>-brine flow, with the following additional assumptions:

The liquid phase is filled with brine only, while the gas phase is filled only with CO<sub>2</sub>. Fluids considered to be immiscible. Fluid diffusion and dispersion is not active because it is assumed that brine only exists in the liquid phase and CO<sub>2</sub> in the gas phase. To facilitate the convergence, the initial CO<sub>2</sub> saturation at the injection nodes is set to 0.5 (and zero elsewhere in the domain).

Parameter	Unit	Value
Injection temperature	m	2500
Injection / production rate	kg/s	variable
Injection BHP at 2500m	MPa	26
Production BHP at 2500m	MPa	24
Injection / production duration	years	25

**Table 3-4:** Injection and production parameters.

Borehole fluid Borehole type	CO2		Brine	Unit
	Inj.	Prod.	Prod.	
BHP at 2805 m	28.9	25.7	25.7	MPa
BHP at 2500 m	26.0	24.0	22.6	MPa

**Table 3-5:** BH injection and production pressures for models with buoyancy.

Parameter	Unit	Value
Residual brine saturation	(decimals)	0.3
van Genuchten m	-	0.457
Brine NaCl saturation	(ppm)	100,000
Residual CO2 saturation	(decimals)	0.05
van Genuchten a	-	$5.1 \times 10^{-5}$
Initial CO2 saturation	0	(decimals)
Initial brine saturation	1.0	(decimals)

**Table 3-6:** Reservoir fluid parameters.

### 3-3 Input meshes

Mesh shown in figure Figure 3-1 was used in models with vertical producers, and the mesh on Figure 3-2 in models with horizontal producers. Both meshes were refined around the boreholes to improve the accuracy and convergence rates of the models. Input files in the appendix can be consulted for meshing parameters.

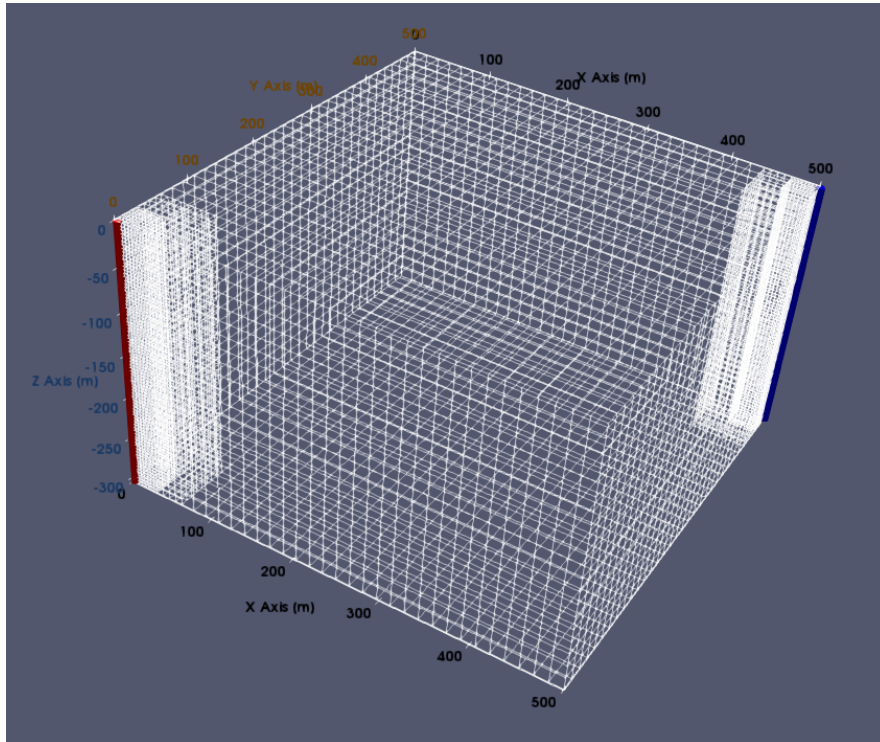


Figure 3-1: Mesh 1: refined around the vertical boreholes.

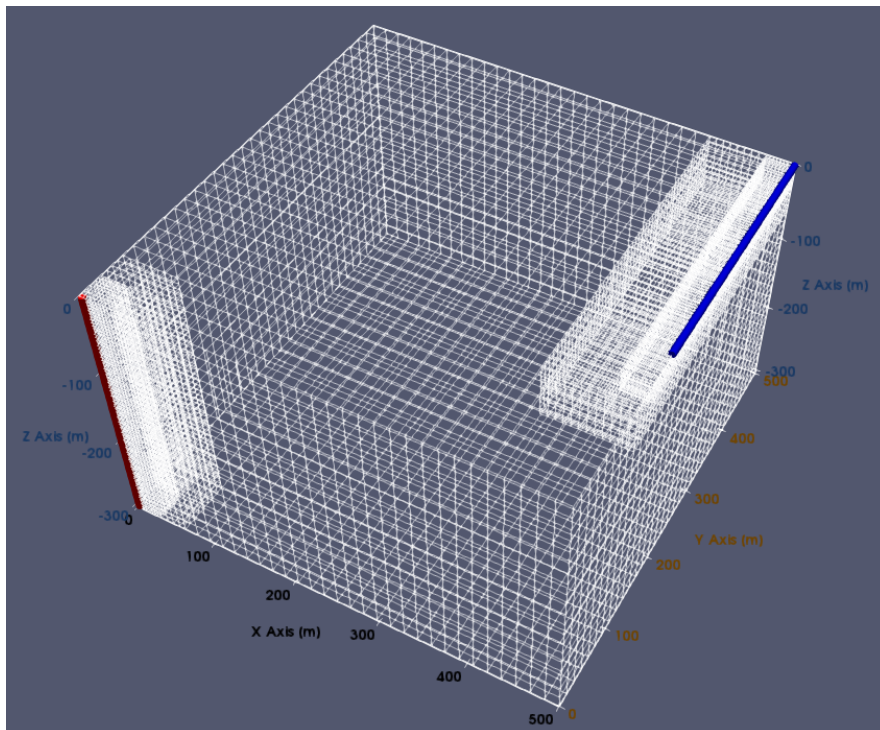


Figure 3-2: Mesh 2: refined around vertical injector and horizontal producer.



---

## Chapter 4

---

# Model validation

When a new numerical model a proper validation and verification procedure is required to ensure that the numerical code adequately simulates physical situation with given parameters. Solutions obtained using the code written for the present study were compared with a simplified analytical 1D solution for the five-spot and with a benchmark numerical solution obtained in TOUGH2 simulator by Randolph and Saar [Randolph and Saar, 2010].

### 4-1 Comparing with the benchmark Randolph and Saar model

#### 4-1-1 Benchmark model

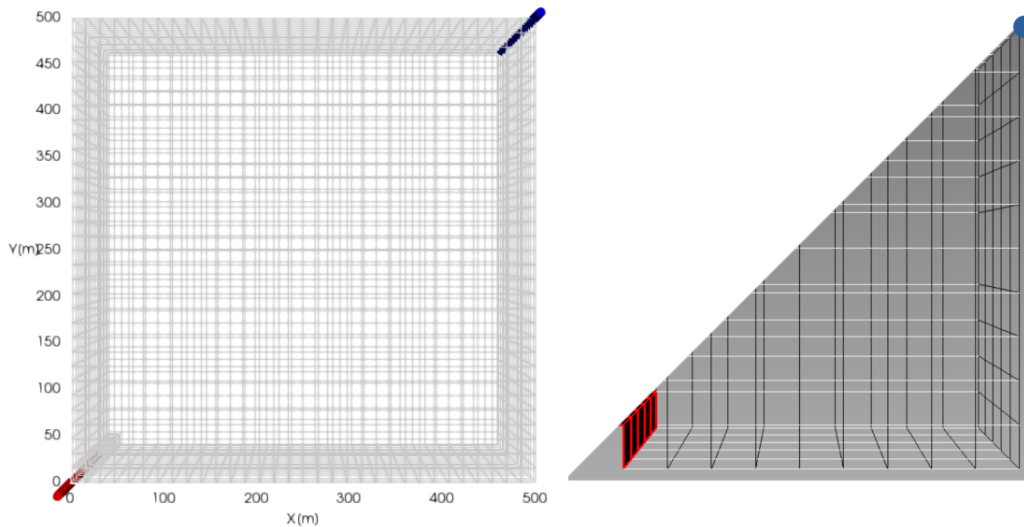
The benchmark model was developed in TOUGH2 integrated finite difference method (IFDM) simulator, which is widely used for solving problems of groundwater flow in porous media. It represents a coupled thermal-hydro model with single phase fully-saturated CO<sub>2</sub> flow in a porous reservoir. The simulation was carried out on a two-dimensional grid representing 1/8th of the 5-spot system domain (Figure 1-1), for a vertical injector-producer pair. The reservoir parameters for the model are identical to those given in the table 3-2. The two-dimensional domain was discretized into 37 grid blocks, each with 70.71m side length. Fluid injection and production parameters are given in the table 3-4, similarly, the flow rates were determined by specifying a 2 MPa pressure difference between wells. Injection and production wells were implemented in a different manner, with injection pressure fixed for an entire grid cell, production well was modeled as a point sink using Peaceman formulation. The effect of this differences on simulation results will be discussed below. Heat extraction rate ( $P_{th}$ ) and fluid flow rate ( $\dot{m}_{prod}$ ) were monitored at a production well, with  $P_{th}$  computed as:  $P_{th} = \dot{m}_{prod} (h_{prod} - h_{inj})$ , where  $h_{prod}$  and  $h_{inj}$  are respectively the enthalpy of the produced fluid enthalpy of the fluid at injection conditions ( $T = 293$  K). The numerical code was verified by comparing the modeling results with the existing CO<sub>2</sub>-based based EGS studies.

### 4-1-2 Base case model (Model A)

The model setup reproduces that used by Randolph and Saar except for the extent of the simulation domain: Randolph and Saar used 1/8th of the 5-spot system, whereas in the present study 1/4 of the fivespot was simulated (fig. The simulation settings as in the tables 3-2 and 3-4, buoyancy is ignored. Model A was run on a 3D FEM mesh refined around the boreholes, the mesh is shown on Figure 3-1 Both injection and production wells were modelled as nodal source / sink functions modified as per Peaceman formulation. Heat extraction rate was computed in the following way:  $P_{th} = \dot{m}_{prod} * h_{prod} - \dot{m}_{inj} * h_{inj}$ , where  $\dot{m}_{inj}$  is injection rate.  $P_{th}$  estimated in this manner is more accurate than the method used in the benchmark study, as it accounts for density differences at the injection and production conditions.

### 4-1-3 Comparison

Figure 4-1 shows meshes for both models, mesh for the benchmark model was reproduced from description. Differences in the size of the domains can easily be accounted for. The figure also shows a conceptual difference in implementation of the injection boreholes: Model A uses Peaceman formulation, with the borehole at a nodal position, oriented along the Z-axis. In the Benchmark model the pressure is set at the entire side of the first grid cell, i.e. injection occurring along 70.7 m boundary (the diagonal length of 50x50 m square grid cell).



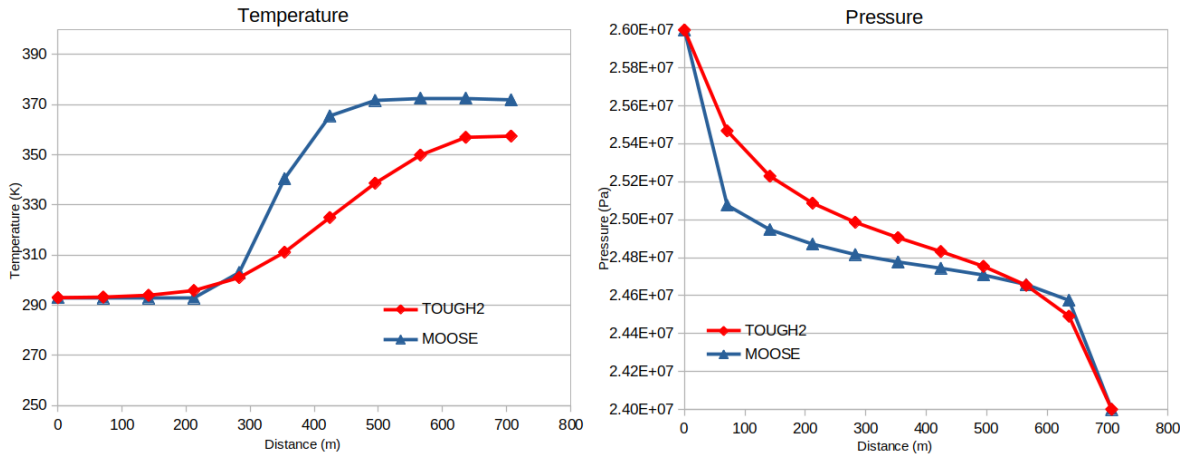
**Figure 4-1:** Mesh comparison: Model A (left) vs Benchmark (right). Injector in red, producer in blue. Refinement around the boreholes for the Model A not shown

Results of both simulations are summarized in the table 4-1 below: Both models show similar RI values, but are considerably different with regard to the heat extraction. Figure 4-2 compares distribution of temperature and porepressure for both models after 25 years of simulation. Values were linearly interpolated between the sampled nodal points (Model A) and the grid cell centers (Benchmark). Pressure gradients are drastically different near the injector well, but become similar for both models near the production boreholes. Observed differences can be attributed to different borehole implementation. Benchmark implementation of the injector is equivalent using a borehole with a radius of 64 m (calculating  $r_w$  from

Model	RI [kPa*s/kg]	$P_{th}$ [MW]	$E_{th}$ extracted [J $\times 10^{16}$ ]
Model A	60.3	25.4	2.04
Benchmark	59.5	37.0	3.57

**Table 4-1:** Performance of a 5-spot system: Benchmark vs Model A. All values given after 25 years of simulation.

(4-3) for  $x_0 = 50m$ ) or can alternatively be thought as a shortening of the simulation domain by 1 grid cell. This results in a smaller pressure drop between the injector and the adjacent grid cells see (Figure 4-2). Temperature profiles too show very different characters. While the Model A shows a cooling front with sharp boundaries that has advanced to approximately the middle of the 5-spot domain, the temperature plot of the benchmark has significantly more diffusive character. Sharper cooling fronts develop in systems where advection plays greater role in transporting heat than thermal conduction, which is also expected in the present case. Diffusive temperature profile most likely results from a coarser spatial discretization in the benchmark model. As was already mentioned in the Chapter 3, coarse spatial discretization requires additional numerical stabilization, which can take form of added numerical diffusion. Hence diffusive temperature profile is likely a numerical artifact.



**Figure 4-2:** Comparison with the Benchmark: pressure-temperature plots.

Similar conclusions can be drawn from observing temporal changes in thermal power output and reservoir impedance Figure 4-3. For Model A  $P_{th}$  stays almost constant corresponding to heat removal at a constant rate by the advancing front, whereas the benchmark shows higher rates in the beginning that gradually decline as the temperature distribution equalizes over the entire domain.

The benchmark model results can be reproduced by modifying reservoir permeability in the injector vicinity (to reduce pressure drop in the adjacent nodes) and running simulation on a coarser mesh with upwinding stabilization. Matching  $E_{th}$  extracted value required using a uniform mesh with  $50 \times 50 \times 30.5$  meter elements and setting permeability in the first element to be equal to  $1.5 \times 10^{-13}$ . Pressure-temperature profiles for the modified model are shown in fig:pt

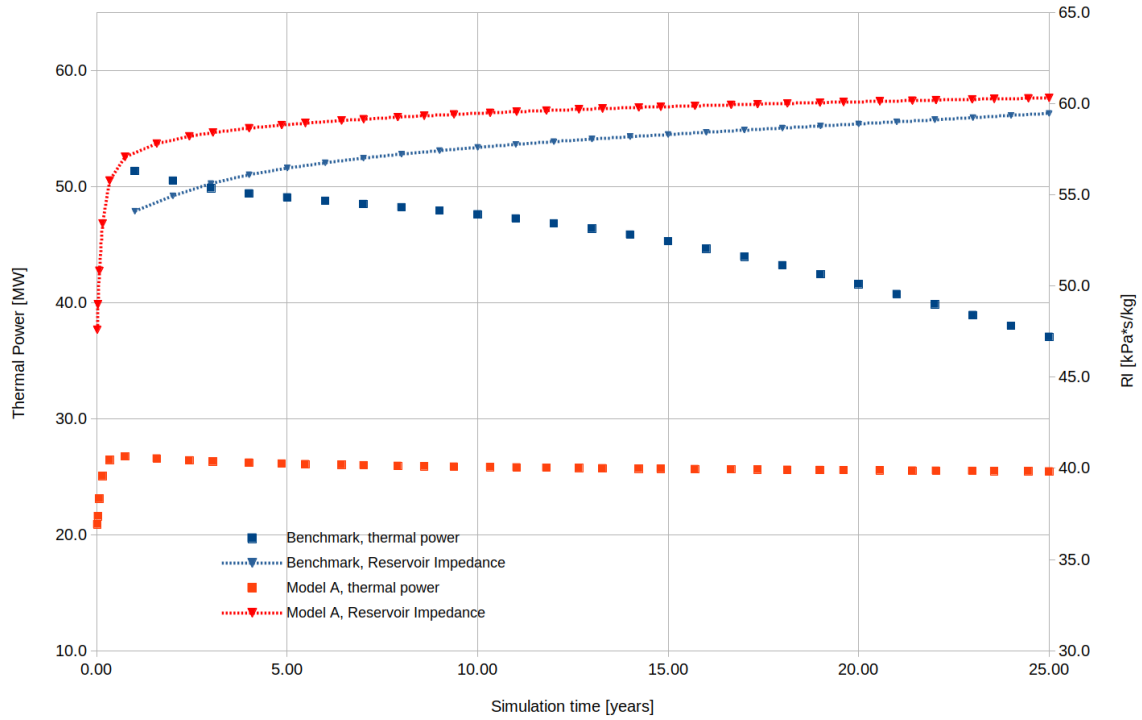


Figure 4-3: Comparison with the Benchmark:  $P_{th}$  and RI

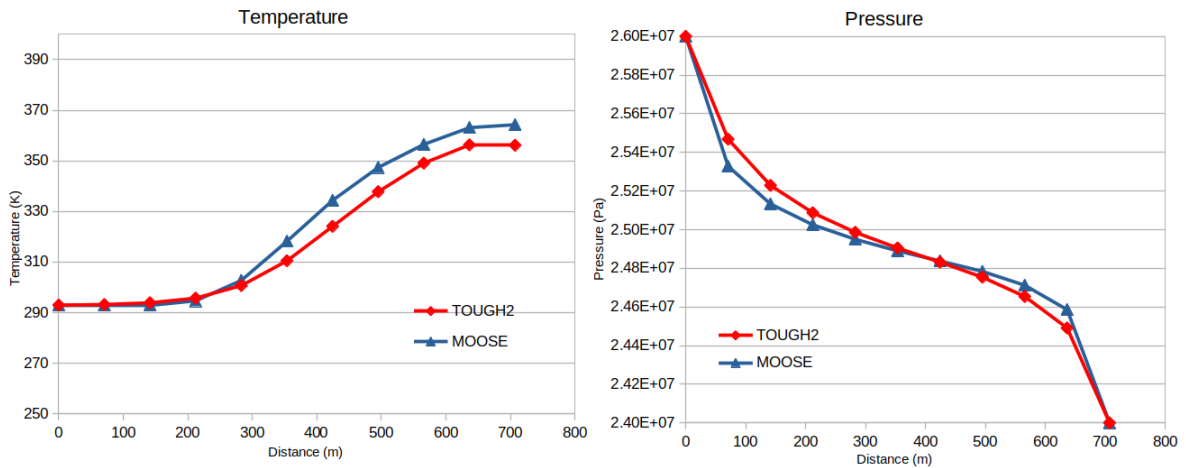
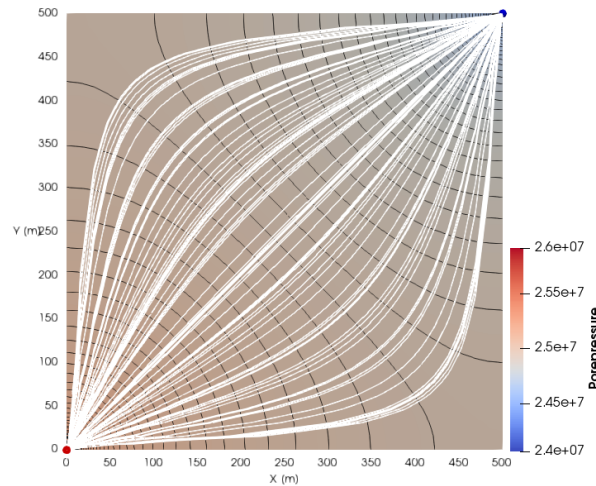


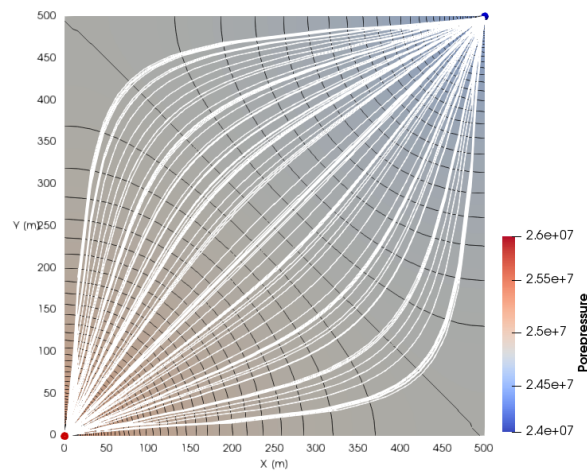
Figure 4-4: Pressure-temperature plots for matched  $E_{th}$ .

Summarizing the discussion above we see that the discrepancies in results stem from the numerical artifacts and less accurate conceptualizations in the benchmark model. By reproducing the benchmark results with modified simulation parameters we show that under similar assumptions (low pressure drop across the first element, added numerical diffusion for the advected variable) the results of both models match. Therefore we conclude that the numerical code for the present model is verified.

## 4-2 Analytical solution



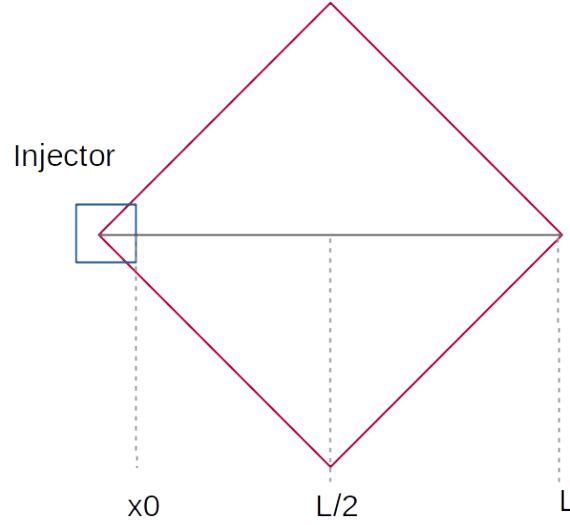
**Figure 4-5:** Model A: streamlines and porepressure contours. Z-plane slice. Streamlines in white, pressure contours in gray. After 1 month of simulation



**Figure 4-6:** Model A: streamlines and porepressure contours. Z-plane slice. Streamlines in white, pressure contours in gray. After 25 years of simulation

Figures 4-5 and 4-6 show the computed pore pressure contours and streamlines after 1 month and 25 years of simulation respectively. It is clear for each case that the four boundaries of the reservoir form a single streamline, while the pressure contours are all perpendicular to the boundaries, indicating that the Neumann boundary conditions (no fluid flow) are satisfied. Figures also indicate that the near well regions are characterized by a higher velocity and pressure gradient. The pressure contours are essentially concentric circles near both the injection and production wells. For this situation a simple 1D analytical Darcy solution for the 5-spot can be derived by making use of symmetry of the problem.

Figure Figure 4-7 shows the simplified problem domain, with homogeneous no-flow boundary conditions for pressure. Here the injection well is represented by a square with a side of  $x_0$ .



**Figure 4-7:** Problem domain for 1D analytical five-spot.

The flow can be computed from Darcy's equation by integrating over the distance (4-1):

$$\int_{x_0}^{L/2} dP dx = \frac{\mu}{\rho * k * 2h} \frac{\partial M}{\partial t} \int_{x_0}^{L/2} \frac{1}{x} dx \quad (4-1a)$$

$$\forall x \in [x_0, L/2] \quad (4-1b)$$

In the governing equation,  $A$  is cross-sectional area, and  $h$  is height or thickness, other symbols are included in the nomenclature. From (4-1) we get the following expression for pressure (4-2):

$$\Delta P = \frac{\mu}{\rho * k * 2h} \frac{\partial M}{\partial t} \ln\left(\frac{L}{2x_0}\right) \quad (4-2)$$

Requiring that the cross-sectional area of the square representing a well to be equal to that of a circle (i.e. base of a cylindrical well) with a radius  $r_w$  we get (4-3):

$$x_0 = \frac{r_w \pi}{4} \quad (4-3a)$$

$$\Delta P = \frac{\mu}{\rho * k * h} \frac{\partial M}{\partial t} \ln\left(\frac{2L}{\pi r_w}\right) \quad (4-3b)$$

We solve this equation for our model problem: 500m x 500m domain (i.e.  $L = 707.14$  m),  $r_w = 0.2$  and  $\Delta P = 2$  MPa (see table 3-2) using average values for density and viscosity corresponding to the pressure and temperature distribution established in the reservoir after 25 years of operation. Resulting mass flow rate  $\dot{m}$  and RI together with the averaged inputs are listed in the table ??:

Numerical results correspond to the analytical 1D solution computed for the injector vicinity, while the solution using averaged parameters for the reservoir. This may in part be attributed to inaccuracies introduced by averaging fluid properties. It may also indicate that the injection flowrate has greater influence on overall system response.

Parameter	Unit	Average for the reservoir	Injector vicinity	Producer vicinity
p	MPa	24.78	26.00	24.00
T	K	340	293	372
$\mu$	Pa*s	$6.50^* \times 10^{-5}$	$11.2^* \times 10^{-5}$	$4.48^* \times 10^{-5}$
$\rho$	kg/m <sup>3</sup>	757	971	572
$\dot{m}$	kg/s	46.05	34.33	50.45
RI	kPa*s/kg	43.43	58.25	39.65

Table 4-2: 1D analytical solution for  $\dot{m}$  and RI

### 4-3 Mesh sensitivity

Comparison with the benchmark highlights the role of spatial discretization on the accuracy of the numerical model of a five-spot. Mesh sensitivity analysis was carried out for hexahedral meshes with varying element sizes and horizontal-to-vertical dimension ratios. The simulation was run for one month due to high computational costs. As shown in Figure 4-5 Figure 4-6 pressure and streamline contours for 1 month and 25 years are distributed in a similar way, thus the results obtained for 1 month of simulation can be applied to characterize the system for longer simulation times. In Figure 4-8 the average reservoir impedance for one month of simulation is plotted vs total number of nodes in the model mesh. Blue dot indicates the position of a locally refined mesh (the mesh in Figure 3-1). We see that the RI values approach an asymptote as the number of nodes increase, i.e. as the spatial discretization gets finer. The plot indicates that for the given problem, non isotropic elements with different side lengths in vertical and horizontal directions can also be used. Plotting only isotropic meshes (which generally assumed to give more accurate results) on a semi logarithmic plot (Figure 4-9) we see that the RI value at the limit (mesh with an infinite number of nodes) equals 58.57 kPa\*s/kg. This value represents the solution for a continuous domain that does not have any spatial discretization errors. Percent error to the value at the limit is indicated by point labels. Hence, vertically refined mesh underestimates RI of the model by 16.85% Here it is assumed that non-uniform, locally refined meshes should have the same value in the limit as uniform ones.

### 4-4 Implementing boreholes

Accurate well models are of critical importance for reservoir simulations. They act as a source for reservoir flow and the accuracy with which the flow from /into the wells is represented directly impacts the heat energy extraction rates and the RI of the five-spot system. Implementing boreholes in a numerical model poses significant challenges due to steep, non-linear pressure gradients in their vicinity. The pressure at a gridblock that contains a well is different from the average pressure in that block and different from the flowing bottom hole pressure for the well. Previous studies by Peaceman, Van Poolen and other authors had suggested that the flow in a borehole vicinity is essentially radial. It has been shown that the Peaceman's correction factor,  $r_e$  depends on the discretization method. Deriving exact solution for a hexahedral FEM mesh was beyond the scope of this work. Instead different  $r_e$  values were compared with pressure gradients of the analytical radial solution, for which  $r_e = 0.56$  used

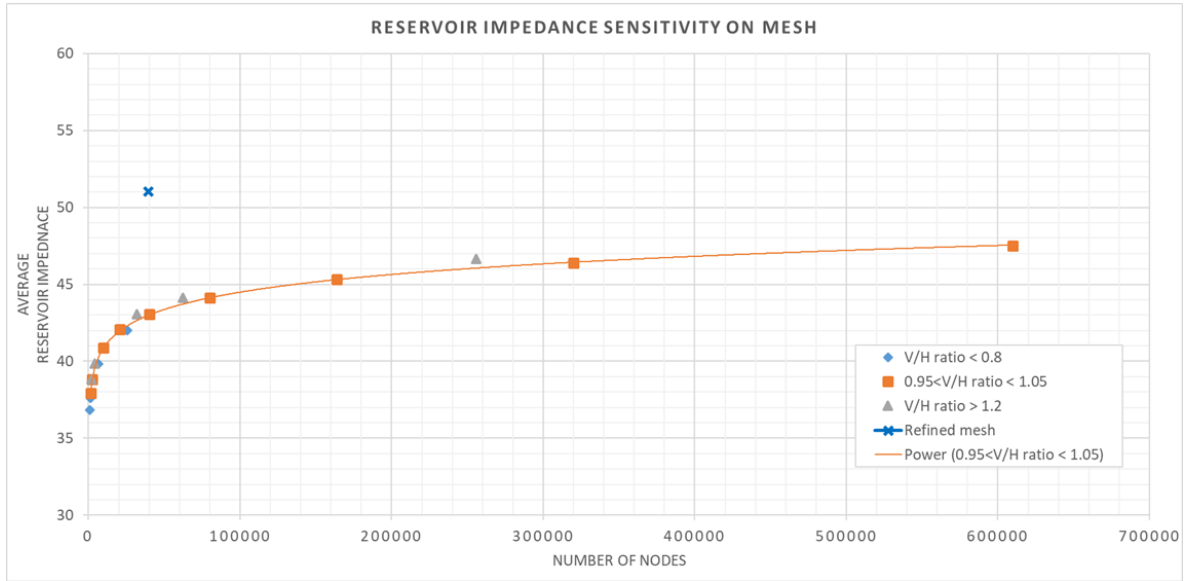


Figure 4-8: Mesh sensitivity: linear plot.

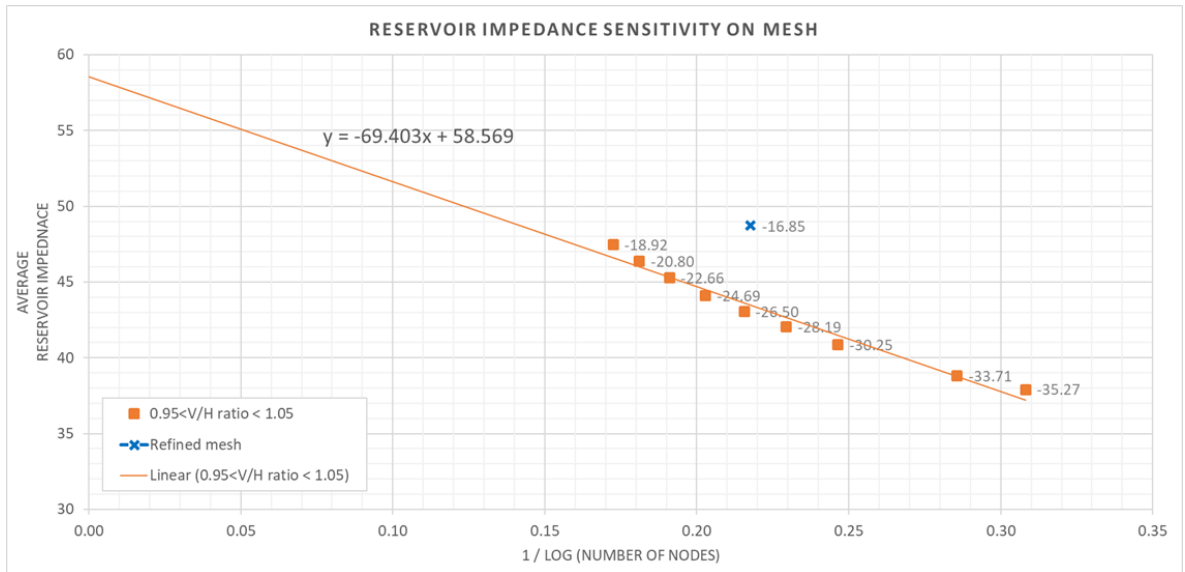


Figure 4-9: Mesh sensitivity: semi logarithmic plot.

with the locally refined mesh produced the best match, Figure 4-10.

### 4-5 Summary

- 1) Numerical code for the Model A was validated against the benchmark. Observed discrepancies were shown to proceed from less accurate conceptualizations in the benchmark model and from numerical artifacts.
- 2) Mesh sensitivity analysis shows that numerical models tend to underestimate reservoir



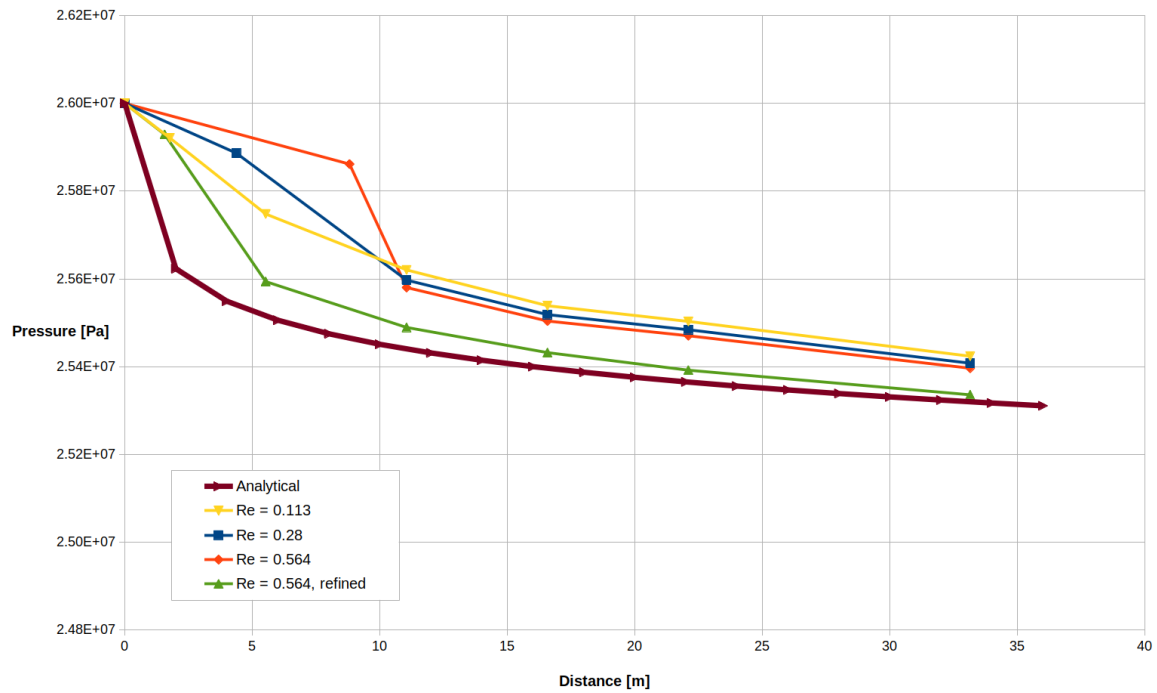


Figure 4-10: Peaceman correction factor

impedance. Resulting RI for the models should be increased by 16.85%.

3) The  $r_e$  parameter in borehole functions was modified to more closely match theoretical pressure gradients around the boreholes. This translates to a higher accuracy of computed flowrate and better convergence of the model.

4) Time constraints did not allow to carry out mesh sensitivity analysis for the models with horizontal boreholes, corrections factors are assumed to be the same as for the verticals.



# Single-phase models

For models discussed below the reservoir is assumed to contain CO<sub>2</sub> only. While the displacement of native formation brine by CO<sub>2</sub> is important and should not be neglected, a single phase model can be a good approximation for a situation where native fluid was displaced before the start of CPG operation. Single-phase fully saturated flow was simulated for two cases: with and without buoyancy forces acting inside the reservoir. Input parameters for the models are given in tables 3-2, 3-3, 3-4. For models with buoyancy the injection-production conditions are given in the table 3-5.

## 5-1 Results

Results for all four models after 25 years of injection-production are given in the table 5-1 below.

**Models A and B: no buoyancy** These two models represent a simplified case in which buoyancy forces inside the reservoir are ignored. Figure 5-1 shows reservoir impedance changes with time.

**Model C and D: with buoyancy** CO<sub>2</sub> density strongly depends on thermobaric conditions and buoyancy plays a major role in reservoir simulations involving CO<sub>2</sub> under changing thermobaric conditions. Figures 5-4 and 5-5 show flow patterns established in the reservoir after 25 years for corresponding models. Figures 5-2 and 5-3 show reservoir impedance change with time. Initial high RI in Figure 5-2 is caused by setting the initial reservoir porepressure to a constant value, without establishing gravitational equilibrium before starting injection-production simulation. A discussion on possibilities for improvement of models developed for this study is included in the summary chapter. Figure 5-3 shows the actual reservoir response after the transient stage caused by gravitational disequilibrium has passed.

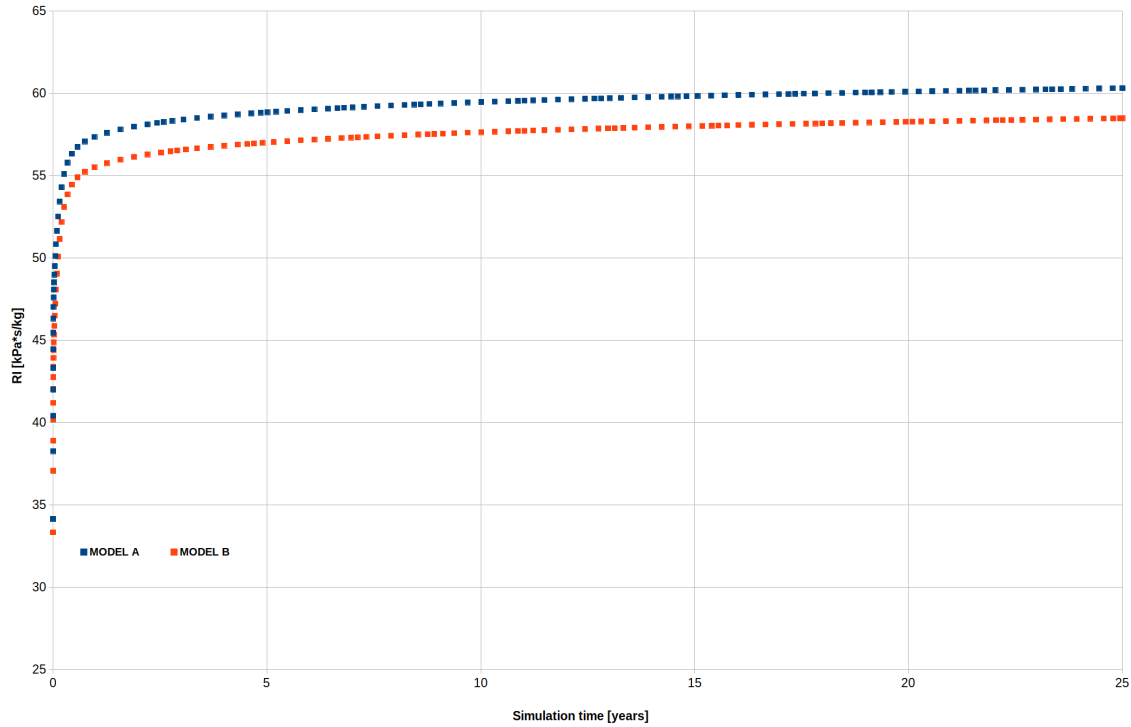


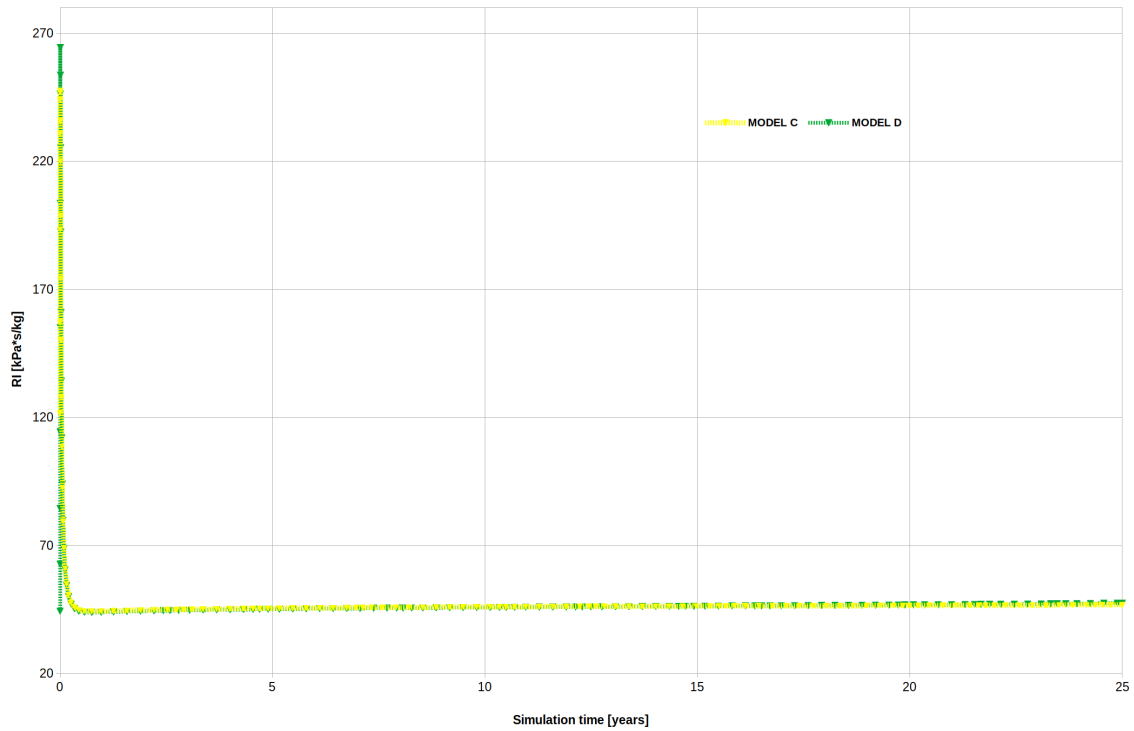
Figure 5-1: Models A and B: reservoir impedance vs time.

Model	RI [kPa*s/kg]	$P_e$ [kW <sub>e</sub> ]
Model A	60.3	463.3
Model B	58.5	477.7
Model C	46.9	570.9
Model D	47.5	566.0

Table 5-1: Summary of results: single-phase models.

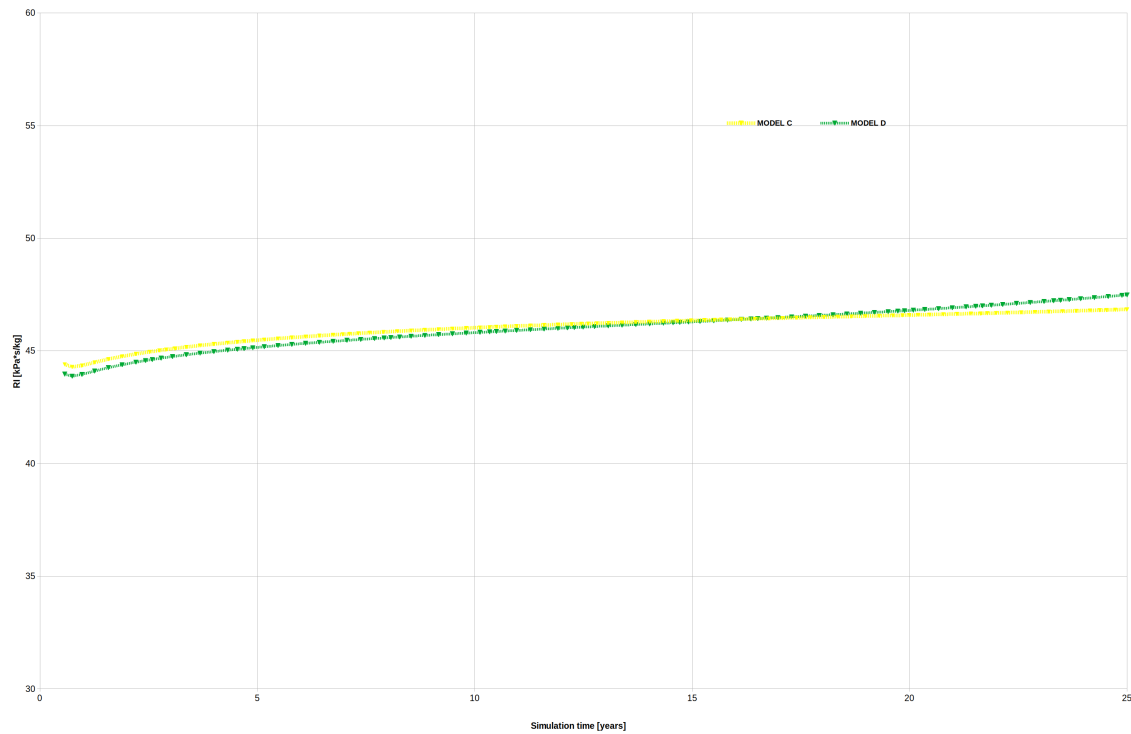
## 5-2 Discussion

Results shown in table 5-1 illustrate a significance of the buoyancy for CPG systems. Both models accounting for the buoyant forces show improved reservoir performance. It is interesting to note that while for the models where the buoyancy is ignored, a horizontal producer marginally outperforms a vertical one, this situation is reversed for the models accounting for it. Streamlines plotted in figures Figure 5-4 and Figure 5-5 show convective mixing occurring in the vicinity of the injector borehole close to the top of the reservoir. In the upper part (approx. upper 50 m ) velocity vectors are pointing back towards the injector. This effect is more pronounced for the model with a horizontal well. CO<sub>2</sub> density at the injection conditions is 970 kg/m<sup>3</sup> vs 590 kg/m<sup>3</sup> at the initial reservoir conditions. This density contrast results in an initially downward directed flow in vicinity of the injector well. Heavy injected CO<sub>2</sub> sinks in the lighter reservoir fluid, heats up on its way down and then travels back up displacing cold CO<sub>2</sub> at the top. This phenomena should elevate the RI for the system, as some portion

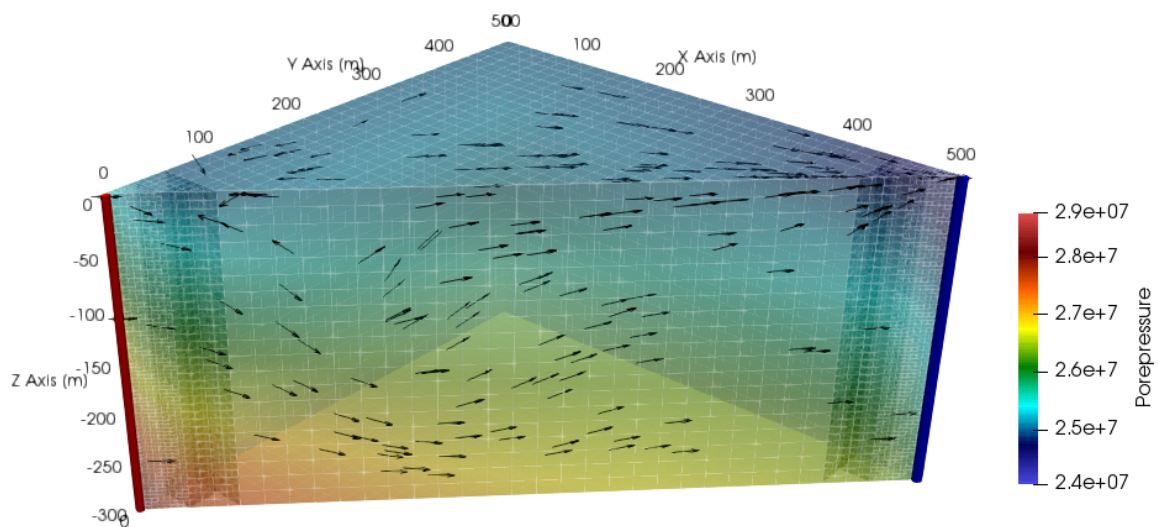


**Figure 5-2:** Models C and D: reservoir impedance vs time, including initial transient stage.

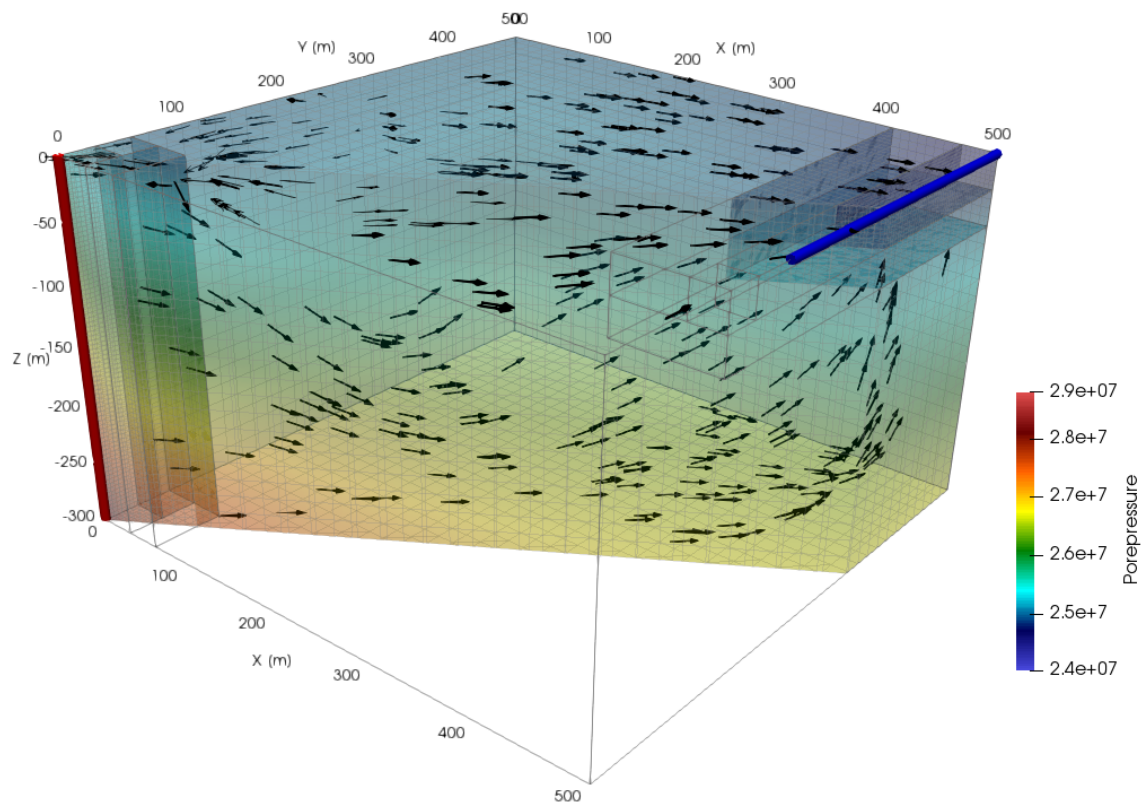
of the pressure drop is being used to circulate the fluid at the top without contributing to the flow towards the production well.



**Figure 5-3:** Models C and D: reservoir impedance vs time, showing quasi-steady state portion of the curve.



**Figure 5-4:** Fluid flow patterns for Model C after 25 years of simulation



**Figure 5-5:** Fluid flow patterns for Model D after 25 years of simulation





---

## Chapter 6

---

# Multi-phase models

The same two well configurations are tested for an initially brine-saturated reservoir. A separate model was run to establish the hydrostatic pressure gradient in a reservoir that is due to gravity. This is the equilibrium state of the reservoir, which was used as the initial condition (see Figure 6-1) for porepressure in both models (Models E and F).

The RI for both models was calculated as the sum of the CO<sub>2</sub> and brine mass flowrates at the production well. At 25 years Models E and F have 95% and 96% fraction of CO<sub>2</sub> entering the production well. Results table shows the time needed to achieve 94% in the produced fluid, as this ratio was typically quoted as a minimum requirement for operation of turbines [N. Garapati, 2015].

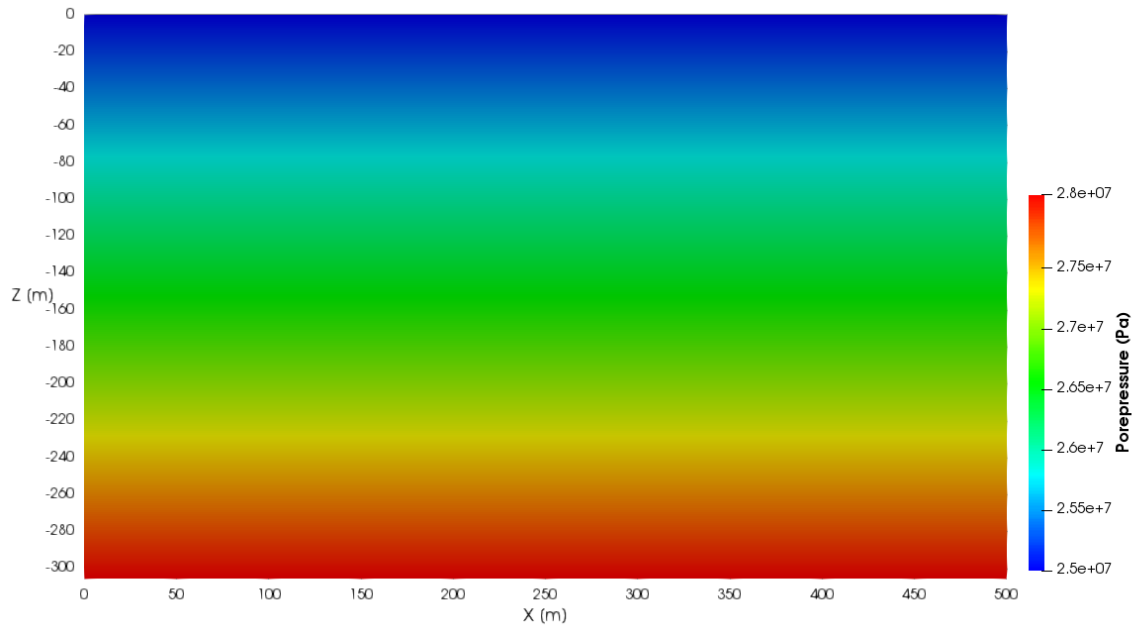
The reservoir rock properties are identical to the single-phase models (see table 3-2). Reservoir fluid parameters are listed in the table 3-6, and the injection-production conditions are given in the table 3-5. A short note in section 3-3 lists some additional assumptions and simplifications applied to the Models E and F.

### 6-1 Results

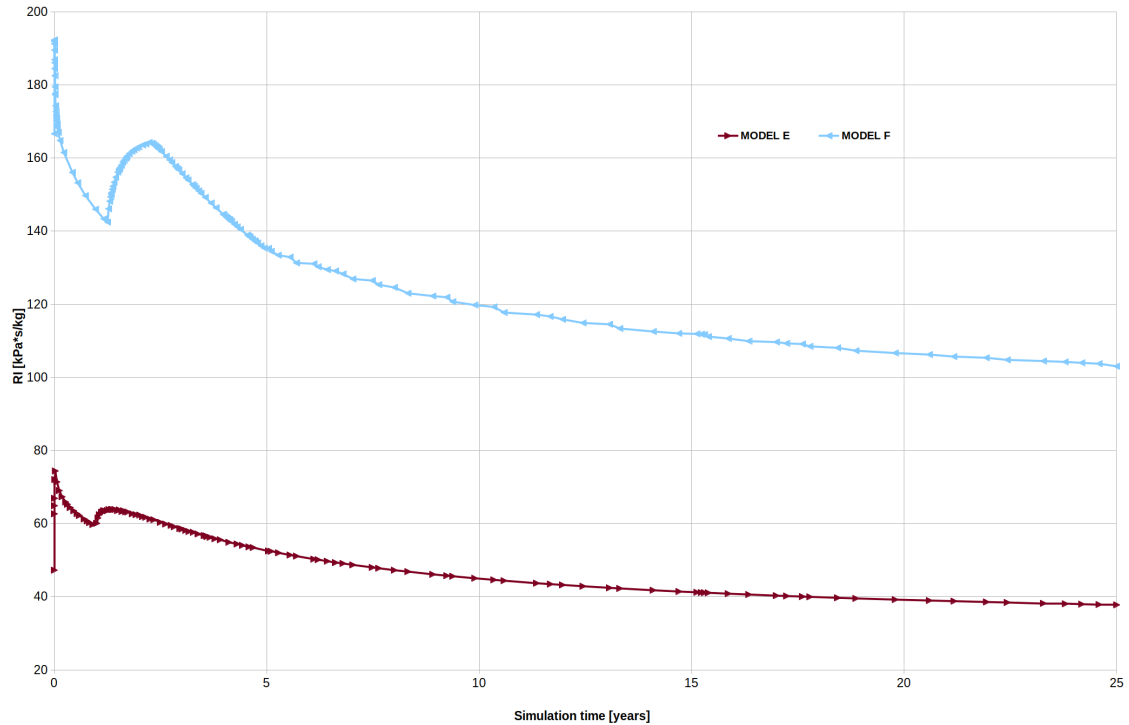
Results obtained for the multiphase models after 25 years of injection-production simulation are given in the table 6-1 below. 94% CO<sub>2</sub> in the last column of the tables is the time needed to reach 94% CO<sub>2</sub> fraction in produced fluid. Reservoir impedance change with time shown in Figure 6-2. To support discussion of the results, a temperature distribution plot for the Model E is shown in Figure 6-3.

Model	RI [kPa*s/kg]	$P_e$ [kW <sub>e</sub> ]	94% CO <sub>2</sub> years
Model E	37.8	643.9	17.7
Model F	103	289.9	19.8

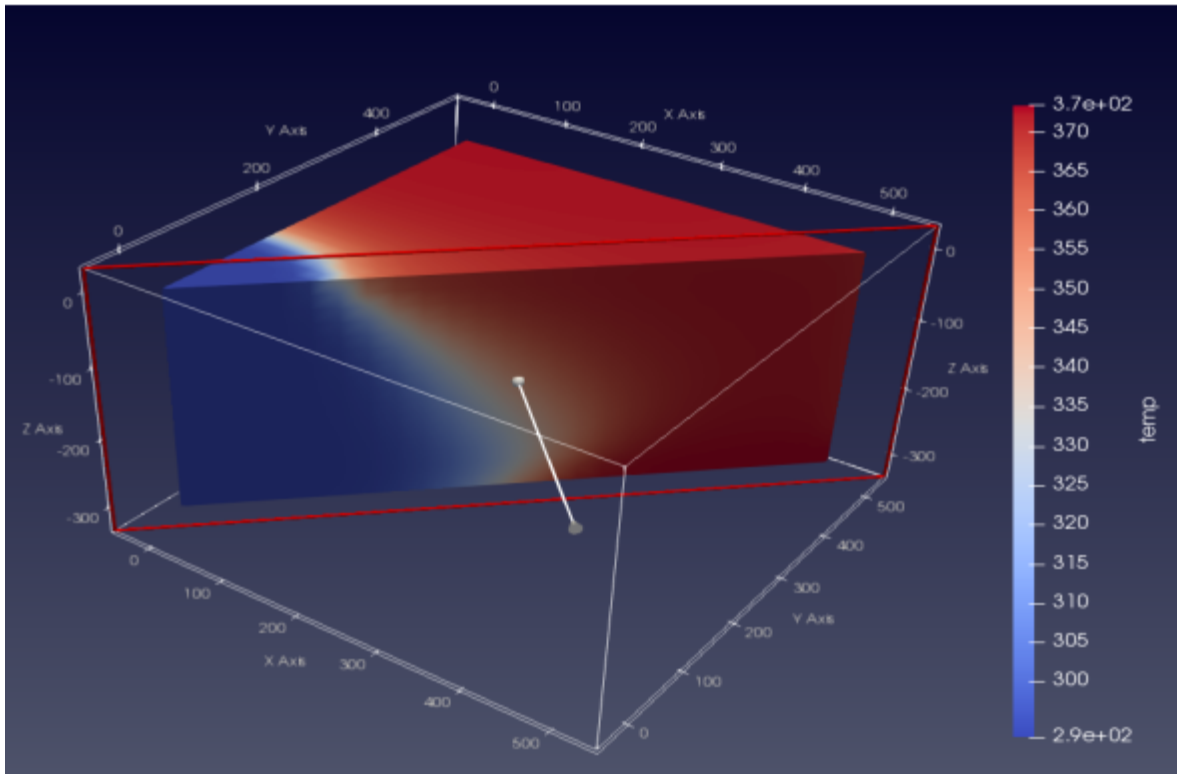
**Table 6-1:** Summary of results: multi-phase models.



**Figure 6-1:** Brine saturated reservoir at gravitational equilibrium. Used as brine porepressure IC for the models E and F.



**Figure 6-2:** Models E and F: reservoir impedance vs time.



**Figure 6-3:** Temperature distribution in Model E after 25 years of simulation.

## 6-2 Discussion

Difference in density between the fluid causes upward directed flow of CO<sub>2</sub>. This has implications for heat extraction, Figure 6-3 shows that the heat is largely extracted from the bottom of the reservoir while the heat extraction from the upper part is lagging. Such temperature distribution is due to buoyancy of CO<sub>2</sub> in brine: the lower part of the reservoir always receives colder fluid than the upper part. The latter receives hotter CO<sub>2</sub> flux coming from below and maintains an elevated temperature. This effect may be responsible for the observed better performance of the model E when compared with the single-phase models. Due to high computational cost of the multiphase models, the vector velocity fields were not explicitly evaluated in the simulator and therefore flow patterns for these models are not discussed here. It is however reasonable to expect that convective mixing also takes places in the multiphase models.

Model E sees further improvement in reservoir performance compared to the single-phase models. Model F on the other hand has the worst reservoir performance among all 6 models. This will need further examination in future studies, presently, by analogy with the single-phase cases, we may point out the convective mixing as one of the possible causes. The initial elevated RI in Figure 6-2 present in both models reflects initially slow displacement of the native formation brine near the injection well due to initial low mobility ( $\frac{k_r \rho}{\mu}$ ) of displacing CO<sub>2</sub>. The nature of the second spike can also be related to mobility, it occurs shortly after CO<sub>2</sub> breakthrough to the producer (at 1 and 1.4 year marks for the Models E and F) when a system reaches a state where  $k_r$  for both phases is low and hence a combined flow is

diminished.

---

# Chapter 7

---

## Summary

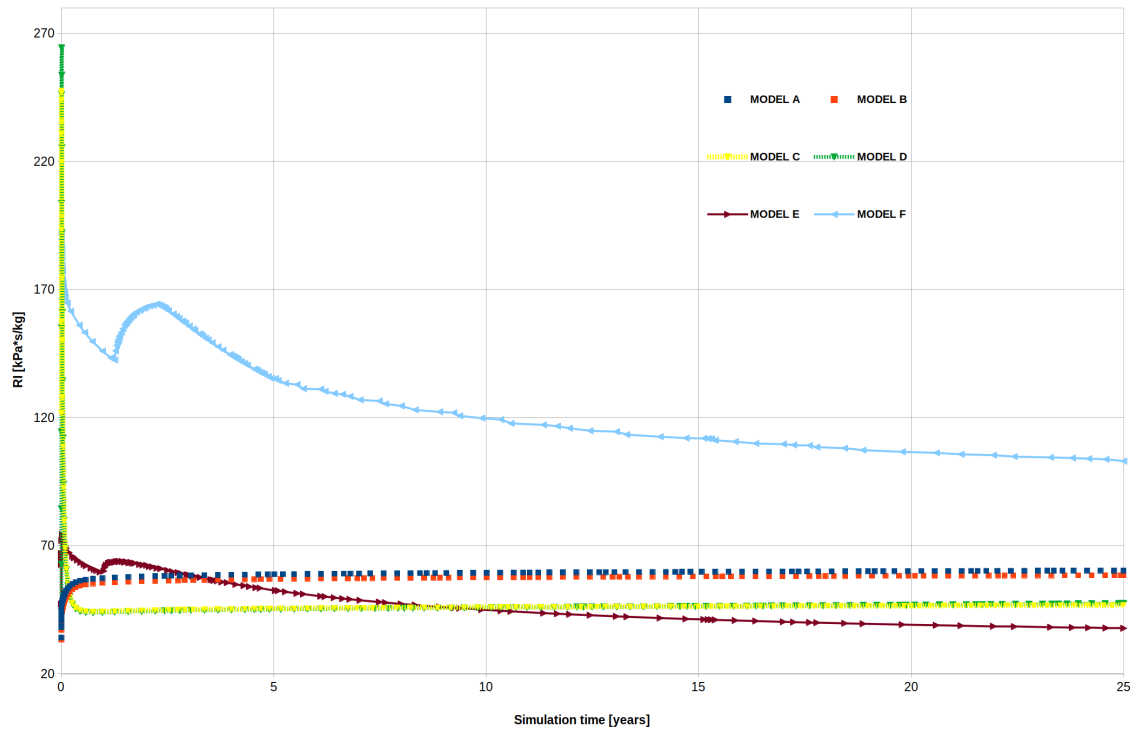
For all single-phase models in the present study, the reservoir performance is controlled by changes in fluid viscosity and density. The ratio of these parameters enters the Darcy's equation and a change in it controls changes in the fluid flow. For multi-phase models change in flowrates is determined by the mobility of CO<sub>2</sub> which increases over time as more of the native formation brine is displaced. Buoyancy plays an important role in the joint response of the reservoir-well system and tends to decrease the efficiency of the horizontal producers. Comparison between the single and multiphase models shows that the native formation fluid properties should be taken into account when choosing well configuration for a CPG system. While further study on production-injection parameters and well placement is needed, we can presently conclude that CPG system will not see any improvement from horizontal producers placed near the top of the reservoir. Summary of results for all models in this study is shown in table 7-1 and a combined RI vs time plot is shown in Figure 7-1.

Model	RI [kPa*s/kg]	$P_e$ [ $kW_e$ ]	94% CO <sub>2</sub> years
Model A	60.3	463.3	na
Model B	58.5	477.7	na
Model C	46.9	570.9	na
Model D	47.5	566.0	na
Model E	37.8	643.9	17.7
Model F	103	289.9	19.8

**Table 7-1:** Summary of results: all models.

### 7-1 Areas for future research on optimization of heat extraction for CPG

1) Modification of the injection conditions. Injection conditions may be optimized by modifying well geometry and injection fluid properties (i.e by injecting at different temperature



**Figure 7-1:** All models: reservoir impedance vs time.

and/or pressure) to minimize the convective mixing described in the previous chapters.

2) Well placement for horizontals. Placing horizontals at different depths or orientation may change the reservoir impedance. Care should be taken to ensure that such placements satisfy the five-spot symmetry.

3) Test different well placements for reservoirs with anisotropic permeability and reservoirs with different temperatures. The latter will largely control CO<sub>2</sub> mobility and the efficiency of brine displacement.

## 7-2 Ways to improve numerical models for present study

- 1) For models C and D: establishing gravitational equilibrium in the reservoir prior to injection.
- 2) Model miscible fluid flow in multiphase simulations.
- 3) Derive Peaceman correction factor for 3D FEM or use alternative formulation that allows for more accurate interpolation of radial flow between the nodes.
- 4) Mesh convergence study for models with buoyancy, multiphase flow and horizontal wells.

Bibliography





---

# Bibliography

- [B. M. Adams, 2014a] B. M. Adams, e. a. (2014a). On the importance of the thermosiphon effect in cpge (co<sub>2</sub> plume geothermal) power systems.
- [B. M. Adams, 2014b] B. M. Adams, Thomas H. Kuehn, J. M. B. J. B. R. M. O. S. (2014b). A comparison of electric power output of co<sub>2</sub> plume geothermal (cpge) and brine geothermal systems for varying reservoir conditions.
- [Bear, 2010] Bear, J. (2010). *Modeling Groundwater Flow and Contaminant Transport*. Springer.
- [Chen and Zhang., 2009] Chen, Z. and Zhang., Y. (2009). Well flow models for various numerical methods.
- [Diersch, 2018] Diersch, H. (2018). *Finite Element Modeling of Flow, Mass and Heat Transport in Porous and Fractured Media*. Springer.
- [INL, 2018] INL (2018). Moose online documentation.
- [Kukreti and Rajapaksa, 1989] Kukreti, A. R. and Rajapaksa, Y. (1989). A numerical model for simulating two-phase flow through porous media.
- [N. Garapati, 2015] N. Garapati, J. B. Randolph, M. O. S. (2015). Brine displacement by co<sub>2</sub>, energy extraction rates, and lifespan of a co<sub>2</sub>-limited co<sub>2</sub>-plume geothermal (cpge) system with a horizontal production well.
- [Peaceman., 1983] Peaceman., D. W. (1983). Interpretation of well-block pressures in numerical reservoir simulation with nonsquare grid blocks and anisotropic permeability.
- [Randolph and Saar, 2010] Randolph, J. B. and Saar, M. O. (2010). Coupling geothermal energy capture with carbon dioxide sequestration in naturally permeable, porous geologic formations: A comparison with enhanced geothermal systems.
- [Randolph and Saar, 2011a] Randolph, J. B. and Saar, M. O. (2011a). Coupling carbon dioxide sequestration with geothermal energy capture in naturally permeable, porous geologic formations: Implications for co<sub>2</sub> sequestration.

[Randolph and Saar, 2011b] Randolph, J. B. and Saar, M. O. (2011b). Impact of reservoir permeability on the choice of subsurface geothermal heat exchange fluid: Co<sub>2</sub> versus water and native brine.

---

# Appendix A

---

## Numerical code for the multiphase model

Unsaturated 2-phase 2-component CO<sub>2</sub>-brine flow with 1 injection and 1 production horizontal boreholes (1/4 of a 5-spot system). Simulation of cold CO<sub>2</sub> injection into an initially brine-filled reservoir. There are two phases, and two components. The liquid phase is filled with brine only, while the gas phase is filled with CO<sub>2</sub> only. There is capillarity between the brine and CO<sub>2</sub>. Heat is advected with the fluids as well as conducted through the fluid-rock system. Fluids are injected and extracted using Peaceman Boreholes (nodal point sources). Fluids considered to be immiscible. Fluid diffusion and dispersion is not active because brine only exists in the liquid phase and CO<sub>2</sub> in the gas phase.

### A-1 Input file

[Mesh] Input mesh for simulation with horizontals has been additionally refined @ top of the reservoir type = FileMesh file = mesh<sub>horiz</sub>32x32x20<sub>uniref</sub>.xdr []

```
[MeshModifiers] [./injectionnodeset]type = BoundingBoxNodeSetnewboundary = 'ColdInjection'bottomleft = ' 00 - 305'topright = ' 000'location = INSIDE[.][./injectionsubdom]type = SubdomainBoundingBoxblockid = 1blockname = injectionsubdomlocation = INSIDEbottomleft = ' 00 - 305'topright = ' 31.2531.250'[.][./injectionsubdom]type = SubdomainBoundingBoxblockid = 3blockname = injectionvIClocation = INSIDEbottomleft = ' 00 - 305'topright = ' 15.62515.6250'[.][./horizYproducersubdom]type = SubdomainBoundingBoxblockid = 4blockname = producersubdomlocation = INSIDEbottomleft = ' 468.75156.25 - 30.5'topright = ' 5005000'[.][ ]
```

[GlobalParams] PorousFlowDictator = 'dictator' gravity = '0 0 -9.8' gravity = '0 0 0' []

```
[UserObjects] [./dictator] type = PorousFlowDictator porousflowvars = 'ppbrinesatgastemp'numberfluidphasess = 2numberfluidccomponents =
```

```

2[./][./producedmassco2]type = PorousFlowSumQuantity[./][./producedmassbrine]type =
PorousFlowSumQuantity[./][./injectedmassco2]type = PorousFlowSumQuantity[./][./heatoutco2]type =
PorousFlowSumQuantity[./][./heatoutbrine]type = PorousFlowSumQuantity[./][./pc]type =
PorousFlowCapillaryPressureVGalpha = 5.1e - 5m = 0.457m =
0.55satlr = 0.3pcmax = 1e7[./][./soln]type = SolutionUserObjectmesh =
graveq32x32x20mesh.xdasystemvvariables = ppbrineexecuteon = 'initial'es =
graveq32x32x20sol.xda[./][.]

```

```

[Modules] [./FluidProperties] [./trueco2]type = CO2FluidPropertieshttps :
//mooseframework.org/source/userobjects/CO2FluidProperties.htm[./][./truewater]type =
Water97FluidProperties[./][./tabulatedco2]Thisisusedtospeedupcomputation.Fluidpropertiesarecompute
TabulatedFluidPropertiesfp = trueco2pressuremin = 1pressuremax =
40E6nump = 60temperaturemin = 283temperaturemax = 393numT =
60fluidpropertyfile = co2tabulated.csvsavefile = false[./][./tabulatedwater]type =
TabulatedFluidPropertiesfp = truewaterpressuremin = 1pressuremax = 40E6nump =
60temperaturemin = 283temperaturemax = 393numT = 60fluidpropertyfile =
water97tabulated.csvsavefile = false[./][./brine]type = BrineFluidPropertieswaterfp =
tabulatedwater[./][./][.]

```

```

[Variables] [./ppbrine]scaling = 1e7[./][./satgas]initialcondition = 0.05scaling =
1E8[./][./temp]scaling = 1e2[./][.]

```

```

[ICs] Input mesh for simulation with horizontals has 9 blocks [./satgaslC]type =
ConstantICboundary = 'ColdInjection'variable = satgasvalue =
0.5[./][./ppbrineic]type = FunctionICvariable = ppbrinefunction =
ppbrineic[./][./rampedtemp]type = FunctionICfunction = tempivariable = tempblock = '
3'[./][./consttemp]type = ConstantICvariable = tempvalue = 373block = '012'[./][.]

```

```

[BCs] In 1D, left = 0, right = 1 In 2D, bottom = 0, right = 1, top = 2, left = 3
In 3D, back = 0, bottom = 1, right = 2, top = 3, left = 4, front = 5 Xminface :
leftXmaxface : rightYminface : bottomYmaxface : topZminface : backZmaxface :
front[./constbrineflux]type = NeumannBCboundary = '012345'variable =
ppbrinevalue = 0[./][./constgasflux]type = NeumannBCboundary = '012345'variable =
satgasvalue = 0[./][./zeroheatflux]type = NeumannBCboundary = '1234'variable =
tempvalue = 0[./][./topheatflux]type = NeumannBCboundary = 'front'variable =
tempvalue = -0.034[./][./bottomheatflux]type = NeumannBCboundary = '
back'variable = tempvalue = 0.034[./][./coldinjection]type = PresetBCboundary = '
ColdInjection'variable = tempvalue = 293[./][./gasinjection]type =
PorousFlowSinkboundary = '20'variable = satgasfluxfunction = -1E - 4fluxfunction =
injectionratefluidphase = 1usemobility = falsesavein = fluxPFS[./][.]

```

```

[Kernels] [./massbrinedot]type = PorousFlowMassTimeDerivativevariable =
ppbrinefluidcomponent = 0[./][./brineflux]type = PorousFlowAdvectionFluxvariable =
ppbrinefluidcomponent = 0[./][./massco2dot]type = PorousFlowMassTimeDerivativevariable =
satgasfluidcomponent = 1[./][./co2flux]type = PorousFlowAdvectionFluxvariable =
satgasfluidcomponent = 1[./][./energydot]type = PorousFlowEnergyTimeDerivativevariable =
temp[./][./heatflux]type = PorousFlowHeatAdvectionvariable =
temp[./][./heatconduction]type = PorousFlowHeatConductionvariable = temp[./][.]

```

```

[AuxVariables] The mass fractions of each component are un-
changing [./massfracph0sp0]initialcondition = 1allbrineinphase =

```

```

0[./][./massfrac_ph1_sp0]initial_condition = Onobrineinphase =
1[./][./pp_gas]family = MONOMIALorder = CONSTANT[./][./sat_brine]family =
MONOMIALorder = CONSTANT[./][./rho_o2]order = CONSTANTfamily =
MONOMIAL[./][./rho_brine]order = CONSTANTfamily =
MONOMIAL[./][./xnacl]initial_condition = 0.1[./][./darcy_x]order =
CONSTANTfamily = MONOMIAL[./][./darcy_y]order = CONSTANTfamily =
MONOMIAL[./][./darcy_z]order = CONSTANTfamily = MONOMIAL[./][.]

```

```

[AuxKernels] [./sat_brine]type = PorousFlowPropertyAuxvariable = sat_brineproperty =
saturationexecute_on = initialtimestep_end'phase = 0[./][./pp_gas]type =
PorousFlowPropertyAuxvariable = pp_gasproperty = pressurephase = 1execute_on =
initialtimestep_end'[./][./rho_o2]type = PorousFlowPropertyAuxvariable =
rho_o2property = densityphase = 1execute_on = initialtimestep_end'[./][./rho_brine]type =
PorousFlowPropertyAuxvariable = rho_brineproperty =
densityphase = 0execute_on = initialtimestep_end'[./][./darcy_xc_o2]type =
PorousFlowDarcyVelocityComponentvariable = darcy_xcomponent = xfluid_phase =
1[./][./darcy_yc_o2]type = PorousFlowDarcyVelocityComponentvariable =
darcy_ycomponent = yfluid_phase = 1[./][./darcy_zc_o2]type =
PorousFlowDarcyVelocityComponentvariable = darcy_zcomponent = zfluid_phase =
1[./][.]

```

```

[Materials] [./temperature] type = PorousFlowTemperature temperature = 'temp'
[./] [./saturation_calculator_qp]type = PorousFlow2PhasePSphase0_porepressure =
pp_brinephase1_saturation = sat_gasacapillary_pressure = pc[./][./brine]type =
PorousFlowBrinecompute_enthalpy = truecompute_internal_energy =
truecompute_density_and_viscosity = truexnacl = xnaclphase =
0[./][./co2_qp]type = PorousFlowSingleComponentFluidat_nodes =
falserequiredbyPorousFlowFullySaturatedHeatAdvectionkernelfp =
tabulated_co2phase = 1[./][./co2_nodal]type = PorousFlowSingleComponentFluidat_nodes =
truerequiredbyPorousFlowEnergyTimeDerivativekernelfp = tabulated_co2phase =
1[./][./rock_heat]type = PorousFlowMatrixInternalEnergySpecificHeatCapacity =
1000[J/kg/K]density = 2650[kg/m3][./][./rock_thermal_conductivity]type =
PorousFlowThermalConductivityIdealDryThermalConductivity =
2.10002.10002.1'assumedtobeindependentofsaturation[./][./massfrac]type =
PorousFlowMassFractionmass_fraction_vars = massfrac_ph0_sp0massfrac_ph1_sp0'[./][./permeability]type =
PorousFlowPermeabilityConstpermeability = 5E - 140005E - 140005E -
14'[./][./relpermVG_gas]type = PorousFlowRelativePermeabilityVGphase =
1m = 0.457m = 0.55s_res = 0.05wetting = falsesum_s_res =
0.35zero_derivative = trueeff_turnover = 0.9[./][./relpermVG_liq]type =
PorousFlowRelativePermeabilityVGphase = 0m = 0.457m = 0.55s_res = 0.3wetting =
truesum_s_res = 0.35zero_derivative = trueeff_turnover = 0.9[./][./porosity]type =
PorousFlowPorosityConstporosity = 0.2'[./][.]

```

```

[DiracKernels] [./co2_injection]type = PorousFlowPeacemanBoreholevariable =
sat_gasSumQuantityUO = injected_mass_co2point_file = injection_80z_bhfunction_of =
pressurefluid_phase = 1mass_fraction_component = 1bottom_pore =
28.9E6re_constant = 0.564forverticals,fromlocalmassconservationre_constant =
0.7forrectangularFE,seederivationinCheng,Zhan" Wellflowmodels.."unit_weight =
00 - 9.55E + 03'grav.accel * densityofCO2(974kg/m3)use_mobility = truecharacter =

```

```

-1[.]/[./co2production]type = PorousFlowPeacemanBoreholevariable =
satgasSumQuantityUO = producedmassco2pointfile = horizprodtop128y.bhfunctionof =
pressurefluidpphase = 1massfractioncomponent = 1bottomport =
24E6reconstant = 0.474forhorizontals,fromlocalmassconservationreconstant =
0.564forverticals,fromlocalmassconservationunitweight = 00 - 5.62E +
03'grav.accel * avgdensityofCO2(temprange : 368 - 373K, 23to24MPA)unitweight =
000'forhorizontalsusemobility = truecharacter = 1[.]/[./brineproduction]type =
PorousFlowPeacemanBoreholevariable = ppbrineSumQuantityUO =
producedmassbrinepointfile = horizprodtop128y.bhfunctionof =
pressurefluidpphase = 0massfractioncomponent = 0bottomport =
24E6reconstant = 0.474forhorizontals,fromlocalmassconservationreconstant =
0.564forverticals,fromlocalmassconservationunitweight = 00 - 1.02E + 04'brinedensity =
1030kg/m3 * (-9.8)unitweight = 000'forhorizontalsusemobility = truecharacter =
1[.]/[./removeheatatproductionco2]type = PorousFlowPeacemanBoreholevariable =
tempSumQuantityUO = heatoutco2pointfile = horizprodtop128y.bhfunctionof =
pressurefluidpphase = 1massfractioncomponent = 1bottomport =
24E6reconstant = 0.474forhorizontals,fromlocalmassconservationreconstant =
0.564forverticals,fromlocalmassconservationunitweight = 00 - 5.62E +
03'grav.accel * avgdensityofCO2(temprange : 368 - 373K, 23to24MPA)unitweight =
000'nogravityusemobility = trueuseenthalpy = truecharacter =
1[.]/[./removeheatatpproductionbrine]type = PorousFlowPeacemanBoreholevariable =
tempSumQuantityUO = heatoutbrinepointfile = horizprodtop128y.bhfunctionof =
pressurefluidpphase = 0massfractioncomponent = 0bottomport =
24E6reconstant = 0.474forhorizontals,fromlocalmassconservationreconstant =
0.564forverticals,fromlocalmassconservationwellconstant = 4 * pi * unitweight =
00 - 1.02E + 04'unitweight = 000'forhorizontalsusemobility = trueuseenthalpy =
truecharacter = 1[.]/[]

```

```

[Preconditioning] [./preferred] type = SMP full = true petscoptionsiname = -pctype -
pcfactormatsolverpackage'petscoptionsvalue = lumumps'[.]/[]

```

```

[Postprocessors] [./massprodco2]type = PorousFlowPlotQuantityuo =
producedmassco2executeon = timestepend'[.]/[./massprodbrine]type =
PorousFlowPlotQuantityuo = producedmassbrineexecuteon =
timestepend'[.]/[./massinjco2]type = PorousFlowPlotQuantityuo =
injectedmassco2executeon = timestepend'[.]/[./heatextractedco2]type =
PorousFlowPlotQuantityuo = heatoutco2[.]/[./heatextractedbrine]type =
PorousFlowPlotQuantityuo = heatoutbrine[.]/[./fluidmass0]type =
PorousFlowFluidMassexecuteon = timestepbegin'[.]/[./fluidmass1]type =
PorousFlowFluidMassexecuteon = timestepend'[.]/[./zmasserrorrel]type =
FunctionValuePostprocessor function = massbalrelnewexecuteon =
timestepend'[.]/[./zmasserrorabs]type = FunctionValuePostprocessor function =
massbalabseexecuteon = timestepend'[.]/[./heatproductionco2]type =
FunctionValuePostprocessor function = heatproducedco2executeon =
timestepend'[.]/[./heatproductionbrine]type = FunctionValuePostprocessor function =
heatproducedbrineexecuteon = timestepend'[.]/[./timestep]type =
TimestepSizeexecuteon = timestepend'[.]/[./flowrateCO2prodBh]type =
FunctionValuePostprocessor function = flowrateprodco2[.]/[./flowratebrineprodBh]type =

```

```
FunctionValuePostprocessor function = flowrateprodbrine[.]/[./heatrateco2prodBh]type =
FunctionValuePostprocessor function = heatrateprodco2[.]/[./heatratebrineprodBh]type =
FunctionValuePostprocessor function = heatrateprodbrine[.]/[]
```

```
[VectorPostprocessors] [./DarcyVelx]type = ElementValueSamplervariable = '
darcy'xsortby = idexecuteon = ' timestepend'[.]/[./DarcyVely]type =
ElementValueSamplervariable = ' darcy'ysortby = idexecuteon = '
timestepend'[.]/[./DarcyVelz]type = ElementValueSamplervariable = ' darcy'zsortby =
idexecuteon = ' timestepend'[.]/[]
```

```
[Functions] [./tempic]type = ParsedFunctionvalue = ' 293 + a *
sqrt(x2 + y2)'vars = ' a'vals = ' 3.62'[.]/[./ppbrineic]type =
SolutionFunctionsolution = solnfromovariable = ppbrine[.]/[./heatproducedco2]type =
ParsedFunctionvalue = ' (c + a * b)'vars = ' abc'vals = ' -2.7407E +
05massinjco2heatextractedco2'FlowrateattheinjectionwellisusedformonitoringVariable'a'istheavgCO2em
26MPA, 293K) * *massinjisanegativenumber, therefore" + "heatextracted(var'c') =
mdot * hattheproductionbh * *a = -274592@26MPA, 293K[.]/[./heatproducedbrine]type =
ParsedFunctionvalue = ' (c + a * b)'vars = ' abc'vals = ' -2.7407E +
05massinjco2heatextractedbrine'FlowrateattheinjectionwellisusedformonitoringVariable'a'istheavgCO2
26MPA, 293K) * *heatextracted(var'c') = mdot * hattheproductionbh * *a =
-274592@26MPA, 293K[.]/[./flowrateprodco2]type = ParsedFunctionvalue = '
a/b'vars = ' ab'vals = ' massprodco2timestep'[.]/[./flowrateprodbrine]type =
ParsedFunctionvalue = ' a/b'vars = ' ab'vals = ' massprodbrinetimestep'[.]/[./heatrateprodco2]type =
ParsedFunctionvalue = ' a/b'vars = ' ab'vals = ' heatproductionco2timestep'[.]/[./heatrateprodbrine]type =
ParsedFunctionvalue = ' a/b'vars = ' ab'vals = ' heatproductionbrinetimestep'[.]/[./massbalrelnew]type =
ParsedFunctionvalue = ' abs((a - c + d + f + g)/(2 * (a - c)))'vars = ' acdfg'vals = '
fluidmass1fluidmass0massinjco2massprodco2massprodbrine'[.]/[./massbalabs]type =
ParsedFunctionvalue = ' abs((a - c + d + f + g))'vars = ' acdfg'vals = '
fluidmass1fluidmass0massinjco2massprodco2massprodbrine'[.]/[]
```

```
[Executioner] type = Transient scheme = implicit-euler solvettype = Newtonnlreltol =
1E - 10nlabstol = 1e - 10endtime = 7.8894E8dtmax = 1.5e7automaticscaling =
true[./TimeStepper]type = SolutionTimeAdaptiveDTdt = 20E3initialdirection =
1percentchange = 0.2[.]/[]
```

```
[Outputs] executeon = ' timestepend'printilinearresiduals =
trueperfgraph = truexda = false[./exodus1]type = Exodusinterval =
1filebase = MultigreqHorre05632x32x20UNIREFmob[.]/[./csv1]type =
CSVinterval = 1executevectorpostprocessorson = NONEfilebase =
MultigreqHorre05632x32x20UNIREFmobfilebase = CO2Brinegraveq45x45x20re011hzprodiop[.]/
```

```
[]
```

```
[Debug] showvarresidualnorms = true[]
```





---

## Appendix B

---

# Numerical code for the single-phase models

### B-1 Input file

Fully saturated single-phase CO2 case with 1 injection and 1 production vertical boreholes.  
[Mesh] type = FileMesh file = mesh\_ert32x32x20.xdr[]

```
[MeshModifiers] [./injection_nodeset]type = BoundingBoxNodeSetnew_boundary = 'ColdInjection'bottom_left = ' 00 - 305'top_right = ' 000'location = INSIDE[.]/[./injector_subdom]type = SubdomainBoundingBoxblock_id = 1block_name = injector_subdomlocation = INSIDEbottom_left = ' 00 - 305'top_right = ' 15.62515.6250'[.]/[]
```

```
[GlobalParams] PorousFlowDictator = 'dictator' Needed to use PorousFlow functions gravity = '0 0 -9.8' []
```

```
[UserObjects] [./dictator] Needed to use PorousFlow functions type = PorousFlowDictator porous_flow_vars = ' ptemp'number_fluid_phases = 1number_fluid_components = 1[.]/[produced_mass]type = PorousFlowSumQuantityUsedbyDiracKernels(seebelow)[injected_mass]type = PorousFlowSumQuantity[heat_out]type = PorousFlowSumQuantity[]
```

[Modules] FluidProperties module uses pressure, temperature as primary variables to compute specified fluid properties. list of calculated properties

```
https://mooseframework.org/modules/fluid_properties/index.html[./FluidProperties][./true_co2]type = CO2FluidPropertieshttps://mooseframework.org/source/userobjects/CO2FluidProperties.html[.]/[./densityviscosity]type = TabulatedFluidPropertiesfp = true_co2pressure_min = 22E6pressure_max = 30E6num_p = 80temperature_min = 273temperature_max = 383num_T = 110fluid_property_file = co2_tabulated.csvsave_file = false[.]/[.]/[]
```

```
[Variables] [./pp] initial_condition = 25E6scaling = 1E7[.]/[./temp]initial_condition = 373scaling = 1[.]/[]
```

[ICs] `./rampedtemp` type = *FunctionICfunction* = *temp<sub>i</sub>variable* = *tempblock* = '1' `./consttemp` type = *ConstantICvariable* = *tempvalue* = 373 *block* = '023456' `./` []

[BCs] In 1D, left = 0, right = 1 In 2D, bottom = 0, right = 1, top = 2, left = 3 In 3D, back = 0, bottom = 1, right = 2, top = 3, left = 4, front = 5 *X<sub>m</sub>inface* : *leftX<sub>m</sub>axface* : *rightY<sub>m</sub>inface* : *bottomY<sub>m</sub>axface* : *topZ<sub>m</sub>inface* : *backZ<sub>m</sub>axface* : *front* `./constmassflux` type = *NeumannBCboundary* = '012345' *variable* = *ppvalue* = 0 `./` `./zeroheatflux` type = *NeumannBCboundary* = '1234' *variable* = *tempvalue* = 0 `./` `./topheatflux` type = *NeumannBCboundary* = 'front' *variable* = *tempvalue* = -0.034 `./` `./bottomheatflux` type = *NeumannBCboundary* = 'back' *variable* = *tempvalue* = 0.034 `./` `./coldinjection` type = *PresetBCboundary* = 'ColdInjection' *variable* = *tempvalue* = 293 `./` []

[Kernels] Two kernels below describe terms present in the mass continuity equation. For fully saturated single component flow mass flux is described by Darcy flow `./massdot` type = *PorousFlowMassTimeDerivativefluid<sub>c</sub>component* = 0 *variable* = *pp* `./` `./massflux` type = *PorousFlowAdvectiveFluxvariable* = *ppfluid<sub>c</sub>component* = 0 `./` `./energydot` type = *PorousFlowEnergyTimeDerivativevariable* = *temp* `./` `./heatflux` type = *PorousFlowHeatAdvectionvariable* = *temp* `./` `./heatconduction` type = *PorousFlowHeatConductionvariable* = *temp* `./` []

[AuxVariables] `./darcyx` order = *CONSTANTfamily* = *MONOMIAL* `./` `./darcyy` order = *CONSTANTfamily* = *MONOMIAL* `./` `./darcyz` order = *CONSTANTfamily* = *MONOMIAL* `./` []

[AuxKernels] `./darcyx` type = *PorousFlowDarcyVelocityComponentvariable* = *darcy<sub>x</sub>component* = *x* `./` `./darcyy` type = *PorousFlowDarcyVelocityComponentvariable* = *darcy<sub>y</sub>component* = *y* `./` `./darcyz` type = *PorousFlowDarcyVelocityComponentvariable* = *darcy<sub>z</sub>component* = *z* `./` []

[Materials] Materials section acts either as a container for all materials used by kernels in the code, transforms output of one kernel into a form required by another one. `co2nodal` type = *PorousFlowSingleComponentFluidat<sub>n</sub>odes* = *true<sub>fp</sub>* = *tabulated<sub>c</sub>o<sub>2</sub>phase* = 0 `co2qp` type = *PorousFlowSingleComponentFluidat<sub>n</sub>odes* = *false<sub>fp</sub>* = *tabulated<sub>c</sub>o<sub>2</sub>phase* = 0 `rockheat` type = *PorousFlowMatrixInternalEnergy<sub>specific</sub>heat<sub>capacity</sub>* = 1000 `J/kg/K` *density* = 2650 `kg/m3` `rockthermalconductivity` type = *PorousFlowThermalConductivityIdeal<sub>dry</sub>thermal<sub>conductivity</sub>* = 2.10002.10002.1' `massfrac` type = *PorousFlowMassFraction* `PS` type = *PorousFlow1PhaseFullySaturatedporepressure* = 'pp' `temperature` type = *PorousFlowTemperaturetemperature* = 'temp' `permeability` type = *PorousFlowPermeabilityConstpermeability* = '5E - 140005E - 140005E - 14' `krnodal` type = *PorousFlowRelativePermeabilityConstat<sub>n</sub>odes* = *true<sub>kr</sub>* = *1phase* = 0 `krqp` type = *PorousFlowRelativePermeabilityConstat<sub>n</sub>odes* = *false<sub>kr</sub>* = *1phase* = 0 `porosity` type = *PorousFlowPorosityConstporosity* = '0.2' []

[DiracKernels] Boreholes are modelled by *PorousFlowPeacemanBorehole* function. Each borehole file has the following form: 'C X Y Z', where C is weight assigned to each point and is used for scaling. In present case each borehole is assigned one XYZ coordinate and extruded in Z direction via `linedirection` = '001' `by305m` `via` `linelength` =

```

305'.Eachpointisassignedaweightof0.05(innerradiusofaproductontubing).[./fluid;injection]type =
PorousFlowPeacemanBoreholevariable = ppSumQuantityUO = injected_mmasspoint_file =
injection_80z.bhfunction_of = pressurefluid_pphase = 0bottom_por_t =
28.9E6re_constant = 0.564forverticals,localmassconservunit_weight = ' 00 -
9.5E + 03'grav.accel * densityofCO2(974kg/m3)use_mobility = truecharacter =
-1[.][./fluid_production]type = PorousFlowPeacemanBoreholevariable =
ppSumQuantityUO = produced_mmasspoint_file = production_80z.bhfunction_of =
pressurefluid_pphase = 0bottom_por_t = 25.7E6re_constant =
0.564forverticals,localmassconservunit_weight = ' 00 - 5.72E + 03'grav.accel *
avgdensityofCO2(temprange : 370 - 373K, 24to26MPA)use_mobility = truecharacter =
1[.][./remove_heat_at_production_well]type = PorousFlowPeacemanBoreholevariable =
tempSumQuantityUO = heat_outpoint_file = production_80z.bhfunction_of =
pressurefluid_pphase = 0bottom_por_t = 25.7E6re_constant =
0.564forverticals,localmassconservunit_weight = ' 00 - 5.72E + 03'grav.accel *
avgdensityofCO2(temprange : 370 - 373K, 24to26MPA)use_mobility = trueuse_enthapy =
truecharacter = 1[.][.]

```

[Postprocessors] Postprocessor are used to extract or compute certain data from the model and output it specified format. Postprocessors below are used to extract data on total mass flow and mass flow and pressure at the borehole nodes.

```

[mass_prod]type = PorousFlowPlotQuantityuo =
produced_masseexecute_on = ' timestep_end'[][mass;injection]type = PorousFlowPlotQuantityuo =
injected_masseexecute_on = ' timestep_end'[][heat_extracted]type =
PorousFlowPlotQuantityuo = heat_out[][fluid_mass0]type =
PorousFlowFluidMasseexecute_on = ' timestep_begin'[][fluid_mass1]type =
PorousFlowFluidMasseexecute_on = ' timestep_end'[][zmass_error_abs]type =
FunctionValuePostprocessorfunction = mass_balance_execute_on = '
timestep_end'[][zmass_error_rel]type = FunctionValuePostprocessorfunction =
mass_balance_new_execute_on = ' timestep_end'[][heat_production]type =
FunctionValuePostprocessorfunction = heat_produced_execute_on = '
timestep_end'[][heat_production_T2style]Tocomparetheeffectofneglectingvariationindensityb/ntheinjectortype
FunctionValuePostprocessorandtheproducernodesfunction =
heat_produced_T2style_execute_on = ' timestep_end'[][timestep]type =
TimestepSizeexecute_on = ' timestep_end'[][mass_flow_rate_p_rod_Bh]type =
FunctionValuePostprocessorfunction = mass_flow_rate_p_rod_Bh[][thermal_power_p_rod_Bh]type =
FunctionValuePostprocessorfunction = thermal_power_p_rod_Bh[][HeatRate_Entire_5spot]type =
FunctionValuePostprocessorfunction = HeatRate_5spot[]

```

```

[VectorPostprocessors] [DarcyVel_x]type = ElementValueSamplervariable = '
darcy'_sort_by = idexecute_on = ' timestep_end'[][DarcyVel_y]type =
ElementValueSamplervariable = ' darcy'_sort_by = idexecute_on = '
timestep_end'[][DarcyVel_z]type = ElementValueSamplervariable = ' darcy'_sort_by =
idexecute_on = ' timestep_end'[]

```

[Functions] Mass balance function to test if model boundaries have significant effect on the mass flow. [./temp;c]type = ParsedFunctionvalue = ' 293 + a \* sqrt(x<sup>2</sup> + y<sup>2</sup>)'vars = ' a'vals = ' 1.81'[.][heat\_produced]mass;injectionisanegativenumber,therefore" + "type = ParsedFunctionvalue = ' (c + a \* b)'vars = ' abc'vals = ' -274592mass;injectionheat\_extracted'Variable'a'isCO2enthalpyattheinjectionconditions[][heat\_produced\_T2style]I

```

m_dot * hattheproductionbh$type = ParsedFunctionvalue = (c - a * b)'vars = abc'vals =
-274592mass_prodheat_extracted'Variable'a'isCO2enthalpyattheinjectionconditions[./massflowrate_prod
ParsedFunctionvalue = a/b'vars = ab'vals = mass_prodtimestep'[./]/thermal_power_prodh$type =
ParsedFunctionvalue = a/b'vars = ab'vals = heatproductiontimestep'[./]/HeatRate5spot$type =
ParsedFunctionvalue = 4 * a'vars = a'vals =
thermal_power_prodh'[./]massbalance'new$type = ParsedFunctionvalue =
abs((a - c + d + f)/(2 * (a - c)))'vars = acdf'vals =
fluid_mass1fluid_mass0mass_injmass_prod'[./]/massbalance$type =
ParsedFunctionvalue = abs((a - c + d + f))'vars = acdf'vals =
fluid_mass1fluid_mass0mass_injmass_prod'[./]
[]

[Preconditioning] [preferred] type = SMP full = true petsc_options_name = -pc_type -
pc_factor_mat_solver_package'petsc_options_value = lumumps'[][]

[Executioner] type = Transient scheme = implicit-euler solve_type = Newtonnl_rtol =
1E-9end_time = 7.8894E825yearsend_time = 2849400endtimeinseconds(33days)dtmax =
1e7[./TimeStepper]type = SolutionTimeAdaptiveDTdt = 10E3initial_direction =
1percent_change = 0.3[./]

[Outputs] execute_on = timestep_end'print_linear_residuals =
trueperfgaph = truexda = false[./exodus1]type = Exodusinterval =
1file_base = co2upwgrvert32x32x20ref2xe0564[./][./csv1]type =
CSVinterval = 1execute_vector_postprocessors_on = NONE'file_base =
co2upwgrvert32x32x20ref2xe0564[./]

[Debug] show_var_residual_norms = true[]

```

---

# Bibliography

- [B. M. Adams, 2014a] B. M. Adams, e. a. (2014a). On the importance of the thermosiphon effect in cpg (co2 plume geothermal) power systems.
- [B. M. Adams, 2014b] B. M. Adams, Thomas H. Kuehn, J. M. B. J. B. R. M. O. S. (2014b). A comparison of electric power output of co2 plume geothermal (cpg) and brine geothermal systems for varying reservoir conditions.
- [Bear, 2010] Bear, J. (2010). *Modeling Groundwater Flow and Contaminant Transport*. Springer.
- [Chen and Zhang., 2009] Chen, Z. and Zhang., Y. (2009). Well flow models for various numerical methods.
- [Diersch, 2018] Diersch, H. (2018). *Finite Element Modeling of Flow, Mass and Heat Transport in Porous and Fractured Media*. Springer.
- [INL, 2018] INL (2018). Moose online documentation.
- [Kukreti and Rajapaksa, 1989] Kukreti, A. R. and Rajapaksa, Y. (1989). A numerical model for simulating two-phase flow through porous media.
- [N. Garapati, 2015] N. Garapati, J. B. Randolph, M. O. S. (2015). Brine displacement by co2 , energy extraction rates, and lifespan of a co2 -limited co2-plume geothermal (cpg) system with a horizontal production well.
- [Peaceman., 1983] Peaceman., D. W. (1983). Interpretation of well-block pressures in numerical reservoir simulation with nonsquare grid blocks and anisotropic permeability.
- [Randolph and Saar, 2010] Randolph, J. B. and Saar, M. O. (2010). Coupling geothermal energy capture with carbon dioxide sequestration in naturally permeable, porous geologic formations: A comparison with enhanced geothermal systems.
- [Randolph and Saar, 2011a] Randolph, J. B. and Saar, M. O. (2011a). Coupling carbon dioxide sequestration with geothermal energy capture in naturally permeable, porous geologic formations: Implications for co2 sequestration.

[Randolph and Saar, 2011b] Randolph, J. B. and Saar, M. O. (2011b). Impact of reservoir permeability on the choice of subsurface geothermal heat exchange fluid: Co<sub>2</sub> versus water and native brine.