NTNU – Trondheim
Norwegian University of
Science and Technology

Generating a Regression Model Proxy for CO_2 storage

Jørgen Stausland

Petroleum Geoscience and Engineering
Submission date: June 2014
Supervisor: Ole Torsæter, IPT

Norwegian University of Science and Technology
Department of Petroleum Engineering and Applied Geophysics

ABSTRACT

CO_2 storage is regarded an important asset in reducing total CO_2 emissions to the atmosphere. Several methods for storing CO_2 have been proposed, but underground storage in saline aquifers are among the most promising. Storing CO_2 underground is a comprehensive process that requires thorough understanding of aquifer behavior which is acquired through reservoir simulations, which are time consuming and data demanding. Injection is an expensive process and to save cost it is desirable to optimize the injection process. Optimization of injection scenarios require many reservoir simulations. It is desirable to save time on simulating different injection scenarios, and proxy can be created to take over for the simulator. In this thesis a regression model proxy is being built to replace the need for reservoir simulations and to help optimize the injection scenario.

Creating a proxy requires a thorough understanding of the injection process and many simulations has to be conducted. To reduce the amount of simulations required the input parameters can be scaled dimensionless. Still there are many simulations required to generate enough data for the proxy to use. The process can be simplified by writing computer scripts to automate simulations and generation of the proxy. Results prove that it is possible to create a regression model proxy for CO_2 injection scenarios and to use it to find an optimal injection scenario.

SAMMENDRAG

CO_2 lagring er regnet som en viktig del i å redusere CO_2 utslipp til atmosfæren. Flere metoder for CO_2 lagring har blitt foreslått, men få metoder er modne og godt testet. Lagring i vannførende lag i undergrunnen er ansett som en av de mest lovende metodene for CO_2 lagring. Injeksjon av CO_2 i saline akviferer er en omfattende prosess som krever god forståelse av akviferen. Denne forståelsen fås vanligvis gjennom å simulere for det antatte injeksjonsscenarioet. Reservoarsimuleringer er ofte tidkrevende og krever mye datakraft. I tillegg er det dyrt å bore brønner og å injisere CO_2, så det er ønskelig å optimalisere injeksjonsprosessen. Å optimalisere injeksjonsprosessen krever mange simuleringer og det er derfor ønskelig å effektivisere simuleringsprosessen. Dette kan gjøres ved å lage en proxy som kan overta for simuleringene. I denne oppgaven blir det utviklet en proxy basert på regresjon og minste kvadraters metode som kan gi estimater på ulike injeksjonsscenarioer, og brukes til å optimalisere et gitt injeksjonscenario.

For å bygge en injeksjons-proxy kreves det en god forståelse av injeksjonsprosessen og mange simuleringer for ulike scenarioer. Antall simuleringer som kreves kan reduseres ved å skalere input-parameterne gjennom dimensjonsløse likninger. Fortsatt kreves det mange simuleringer for å få tilstrekkelig data til å lage proxyen. Ved å automatisere simuleringene og beregningene som kreves for å lage proxyen går prosessen fortere og blir mer nøyaktig. Resultatene viser at det er mulig å lage en regresjons-basert proxy for CO_2 injeksjon og å bruke den til å finne optimale injeksjonsscenarioer.

ACKNOWLEDGEMENTS

This thesis was made possible by the dedication and knowledge of Dr. Dag Wessel-Berg at SINTEF petroleum. I would therefore like to thank Dr. Dag Wessel-Berg for his tremendous support throughout the semester. I would also like to thank my supervisor Professor Ole Torsæter at NTNU for his guidance through the semester. Finally I would like to thank Håkon Ankervold for proofreading this thesis and for being a good friend through all five years in Trondheim.

TABLE OF CONTENTS

List of figures

List of tables

1 INTRODUCTION

CO_2 storage has gained increased attention as a necessity to reduce CO_2 emissions to the atmosphere. Storing CO_2 requires proper planning to ensure secure storage for sufficient time to ensure it no longer poses a threat to the environment. There exists different methods for CO_2 storage; in deep geological media, by surface mineral carbonation, and in oceans. From these storage methods CO_2 storage in deep geological media is the most mature and promising method (IPCC, 2005).

When injecting CO_2 in deep geological media three different possibilities can be defined; storage in coal seams, storage in depleted hydrocarbon reservoirs either by injection as a part of enhanced oil recovery or for storage purposes alone, and storage in deep saline aquifers. Saline aquifers are assumed to have the largest storage potential for CO_2, with an estimated storage capacity between 30 Gt and 800 Gt in Europe only (Holt, Lindeberg, & Taber, 2000).

CO_2 injection is an expensive strategy for reducing emissions, therefore it is desirable to optimize the injected volume of CO_2. Optimizing injection strategies requires numerous simulations of different injection scenarios to be conducted. Simulations are time consuming and data demanding. One method to reduce simulation time is to create a proxy. A proxy may be defined as any mathematical or statistical function capable of representing the reservoir behavior for selected input parameters (Azad & Chalaturnyk, 2013). To create the proxy a regression model was chosen.

Creating a proxy requires a thorough understanding of the reservoir behavior for the specific circumstances the proxy is intended for. This requires many simulations for different scenarios to give sufficient data to create the proxy model. To avoid mistakes, and to save time, the procedure for creating simulation files, running and analyzing them is automated through a computer program.

The purpose of this thesis is to create a proxy model for CO_2 injection in a homogenous and non-dipping aquifer with a five-spot well pattern. The purpose of the proxy is to optimize injection for a large field. Either by finding optimal operational parameters for maximal storage and breakthrough time to producers or by having a predetermined amount of CO_2 over a specified time limit with as few wells as possible. This report will first give a short overview of CO_2 storage possibilities and trapping mechanisms in saline aquifers. Next the report will give an overview of different proxy models in existence, before explaining the regression model proxy more thoroughly. Input for the regression model is discussed along with the automation process. After the proxy has been created it is tested on a full scale injection scenario and an optimization case.

2 CO_2 STORAGE POSSIBILITIES

There are currently three forms of CO_2 storage identified; in deep geological media, by surface mineral carbonation, and in oceans. Ocean storage is an immature technology, which will alter the environment it covers, and may endanger organisms living there (IPCC, 2005). Surface mineral carbonation is currently expensive, and has a significant environmental imprint (Stefan Bachu et al., 2007). CO_2 storage in deep geological media on the other hand is available due to experience mainly from the oil and gas industry. This storage form has a large available storage potential, large enough to store captured CO_2 in the foreseeable future. Geological storage has, if stored right, a possibility to retain the CO_2 in the ground for thousands to millions of years (Stefan Bachu et al., 2007).

Geological media mainly involves storage in coal seams, hydrocarbon reservoirs and saline aquifers, and is one of the more promising storage concepts (Pruess, Xu, Apps, & Garcia, 2003).

2.1 COAL SEAMS

Carbon dioxide storage in coal seams may be injected as a part of methane production from underground coal, since the coal has a higher affinity for CO_2 than for methane. This technology is still not well developed, and a better understanding of the storage processes is needed (IPCC, 2005).

2.2 HYDROCARBON RESERVOIRS

Hydrocarbon reservoirs are well understood due to oil and gas production. It is also possible to combine this process with increased oil recovery as a part of an EOR project. Another advantage with CO_2 storage in oil and gas reservoirs is that there is already substantial infrastructure in existence, so costs of developing the necessary infrastructure will be greatly reduced. Storage potential is also large but it requires that the reservoirs are depleted or suitable for EOR with CO_2 injection. Still relatively few hydrocarbon reservoirs are depleted, and CO_2 storage has to be staged to fit reservoir availability (IPCC, 2005). However, before CO_2 sequestration becomes commercialized they have a promising potential early in the CO_2 deposition era (Holt et al., 2000).

2.3 Saline aquifers

Saline aquifers have by far the largest storage potential for CO_2 storage, with a storage capacity estimated to be at least 1.000Gt of CO_2 or even a magnitude larger (IPCC, 2005). The aquifers are well distributed around the globe (Pruess et al., 2003), meaning they are often not very far away from a potential point source of CO_2. A disadvantage with deep saline aquifers is that they are less characterized than petroleum reservoirs, and a comprehensive characterization is needed to ensure the suitability of the aquifer proposed as a storage site (Mo & Akervoll, 2005).

Saline aquifers as storage sites for CO_2 disposal is an emerging technology, with an increasing number of field trials for storage. A common problem for CO_2 disposal in aquifers is pressure maintenance. As CO_2 is injected into the aquifer, pressure increases if the aquifer boundaries are closed. If the pressure increase is severe enough the injection well will have to be shut in, and a new site for injection will have to be found. If the well is not shut in before the pressure has risen to critical levels it may cause a fracture in the cap rock and the injected CO_2 will leak back towards the surface. Such problems have been encountered in the Snøhvit project, where pressure increase is a relevant problem and has to be taken care of before reservoir pressure reaches critical limits (Eiken et al., 2011).

3 TRAPPING MECHANISMS

When injecting CO_2 into saline aquifers, four trapping mechanisms are identified. These mechanisms are structural trapping, residual gas trapping, solubility trapping and mineral trapping. Structural trapping involves CO_2 trapped in a geological structure as a free fluid, allowing free flow of the CO_2. The other three trapping mechanisms immobilizes the CO_2, increasing storage security with respect to leakage prevention (Nghiem, Shrivastava, Kohse, Hassam, & Yang, 2010). Structural trapping is the most immediate trapping mechanism in CO_2 injection. Followed by increasing time consuming trapping mechanisms, where the slowest may take thousands of years to yield contribution to the trapping of CO_2, as seen in Figure 3-1.

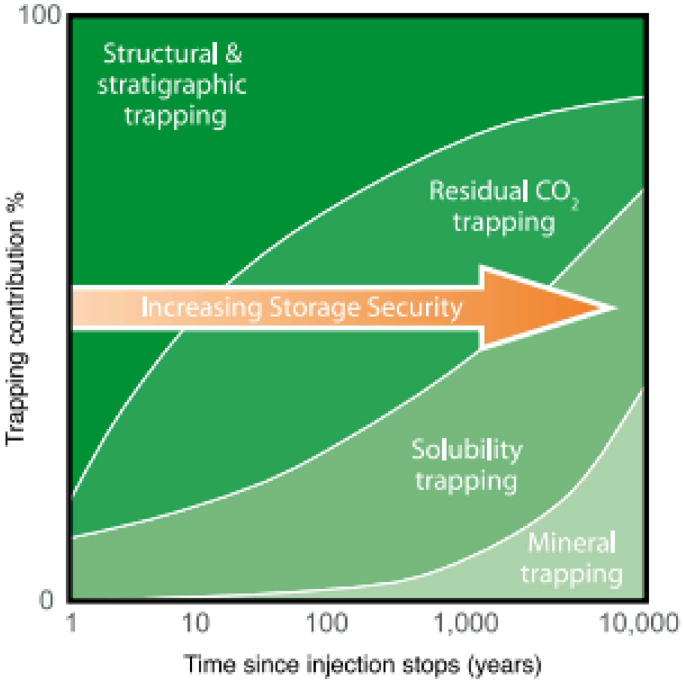


Figure 3-1 Contribution of different trapping scenarios over time (Nghiem et al., 2010).

3.1 Structural trapping

Structural trapping involves storage of CO_2 in free fluid phase. When CO_2 is injected it will rise due to buoyancy forces, until it reaches a cap rock or is diluted enough to become residual trapped. CO_2 below cap rock will be mobile and propagate away from the injection well. If the cap rock seal is intact, the CO_2 will remain trapped in the formation for a very long time and other trapping mechanisms will come into play.

3.2 Residual trapping

As CO_2 propagates further away from the injection spot, it becomes more diluted. With CO_2 being the non-wetting phase for most aquifers the injection will behave as a drainage process. The drainage process will continue after injection stop, since the CO_2 plume propagates away from the injection well, causing further water displacement. At the edge of the CO_2 trail imbibition will occur, snapping off CO_2 from the plume, trapping it with capillary forces (Juanes, Spiteri, Orr, & Blunt, 2006).

3.3 Solubility trapping

When CO_2 is exposed to brine it will begin dissolving into the brine by diffusion, saturating the brine with CO_2. The CO_2 dissolved in brine will stay in solution as long as the brine is trapped in the aquifer. CO_2-saturated brine increases in density, and a convection process will start, forcing the saturated brine down, making room for fresh brine to be exposed to the CO_2 (Soroush, Wessel-Berg, Torsaeter, Taheri, & Kleppe, 2012). Buoyancy forces pushing the heavier CO_2-saturated brine downwards will also act as an extra seal for keeping the CO_2 underground.

3.4 Mineral trapping

When CO_2 is dissolved in aquifer brine it dissociates into H^+ and HCO_3^- ions. These ions can cause precipitation of different calcite minerals, typically calcite

$$CaCO_3 + H^+ = Ca^{2+} + HCO_3^- . \tag{3.1}$$

This conversion of CO_2 into carbonate minerals is known as mineral trapping. Mineralization of CO_2 relies on the presence of minerals in the formation that can provide Ca^{2+}, Mg^{2+} or Fe^{2+} ions for the precipitation of calcite, dolomite and siderite respectively (Nghiem et al., 2010). Mineral trapping is dependent on the mineral composition in the aquifer. Pure sandstone are chemically inert to CO_2 and are dependent on contamination from other minerals to react.

4 FLUID PROPERTIES

Fluid properties are an important factor when considering CO_2 injection and storage. pVT data will affect storage capacity to an extent. It is desirable to have a satisfactory overview of the fluid properties to be able to plan for an optimal injection scheme and storage capacity.

4.1 CO_2 PROPERTIES

CO_2 fluid properties are important parameters when assessing CO_2 storage. Pressure and temperature of the CO_2 will affect several storage parameters which in turn impacts the total storage efficiency. When injecting CO_2 it is possible to control the injection pressure and temperature, to optimize injectivity. Once CO_2 has been injected into the subsurface it will mostly be dependent on formation parameters, which are more difficult to influence.

To assure efficient and secure storage it is desirable to have an aquifer pressure above the critical pressure for CO_2. CO_2 in dense phase have more favorable properties than gaseous or liquid CO_2 with respect to density (Figure 4-1) and viscosity (Figure 4-2). Critical temperature, T_{cr}, and pressure, P_{cr}, of CO2 are 31.1°C and 73.8 bar as seen in Figure 4-3. Above these conditions CO_2 will exist in dense phase (critical phase). To ensure these criteria are achieved a minimum depth of 800m is recommended (Pruess et al., 2003).

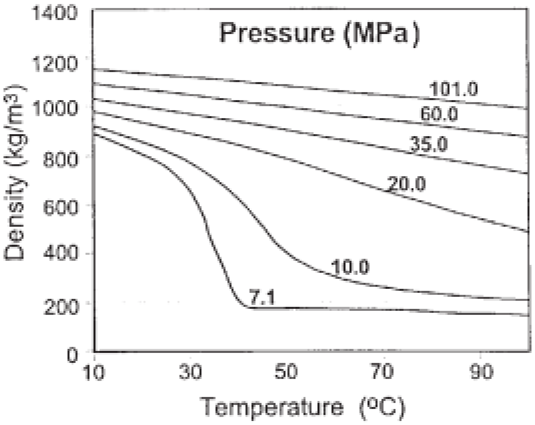


Figure 4-1 CO_2 density diagram, (S. Bachu & Stewart, 2002).

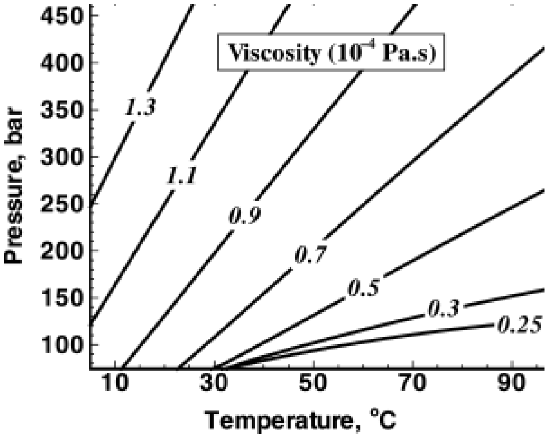


Figure 4-2 CO_2 viscosity diagram, (Pruess et al., 2003).

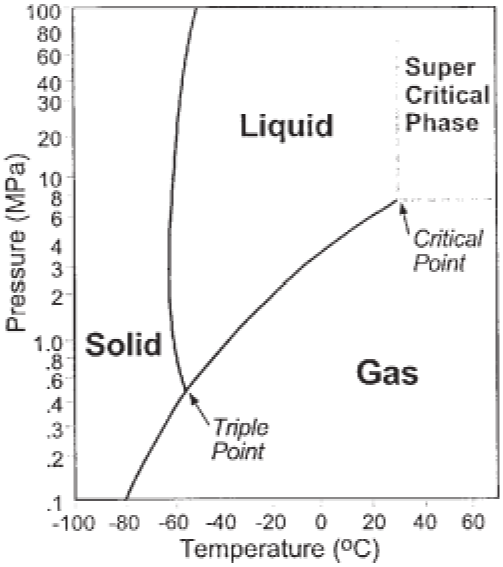


Figure 4-3 CO_2 phase diagram, (S. Bachu & Stewart, 2002).

4.2 BRINE PROPERTIES

Formation brine is dependent on pressure, temperature and salinity. The density of brine is relatively insensitive to pressure, as indicated in Figure 4-4 A. Compared to CO_2 brine is less compressible. Brine density is mostly dependent on formation temperature and salinity, as seen in Figure 4-4 B and C. Density of brine will affect storage potential, as a large density difference between brine and CO_2 enhances gravity segregation, forcing the CO_2 up towards the cap rock earlier. A lower density difference is associated with increased storage, as the front of CO_2 will be larger.

Brine viscosity is mostly dependent on temperature, and increases with decreasing temperatures. Salinity also has a significant effect on the viscosity curve, while the pressure effect is very small. Figure 4-5 illustrates the brine viscosity dependence on Temperature and salinity. A large difference in viscosities between CO_2 and brine will affect storage potential by increasing the mobility ratio. When injecting a low viscous fluid it is desirable to have a low mobility ratio, to decrease the chances for viscous fingering to occur. Viscous fingering will cause early breakthrough and leave un-flooded areas in the aquifer.

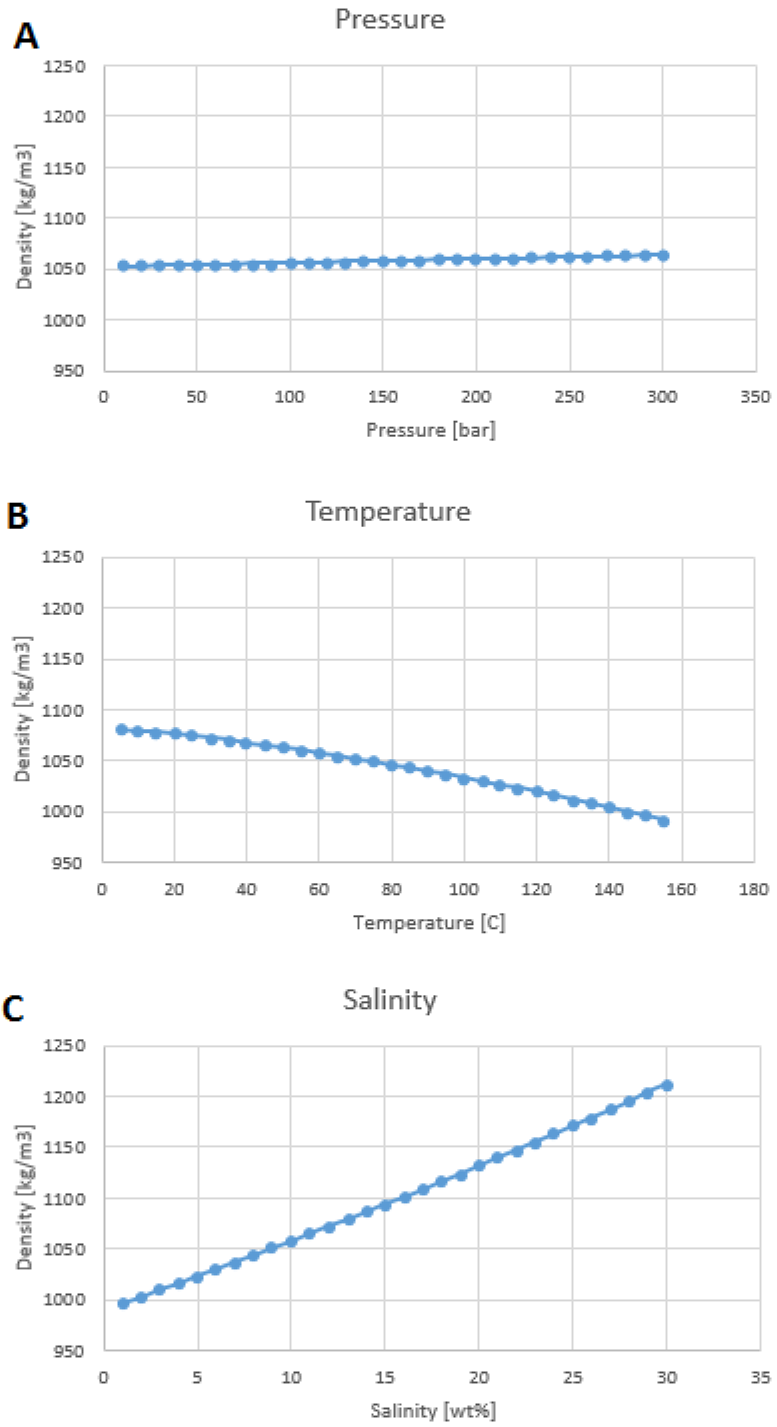


Figure 4-4 Differences in sensitivities for density in brine. A) density sensitivity for pressure changes, B) density sensitivity for temperature changes C) density sensitivity for salinity changes.

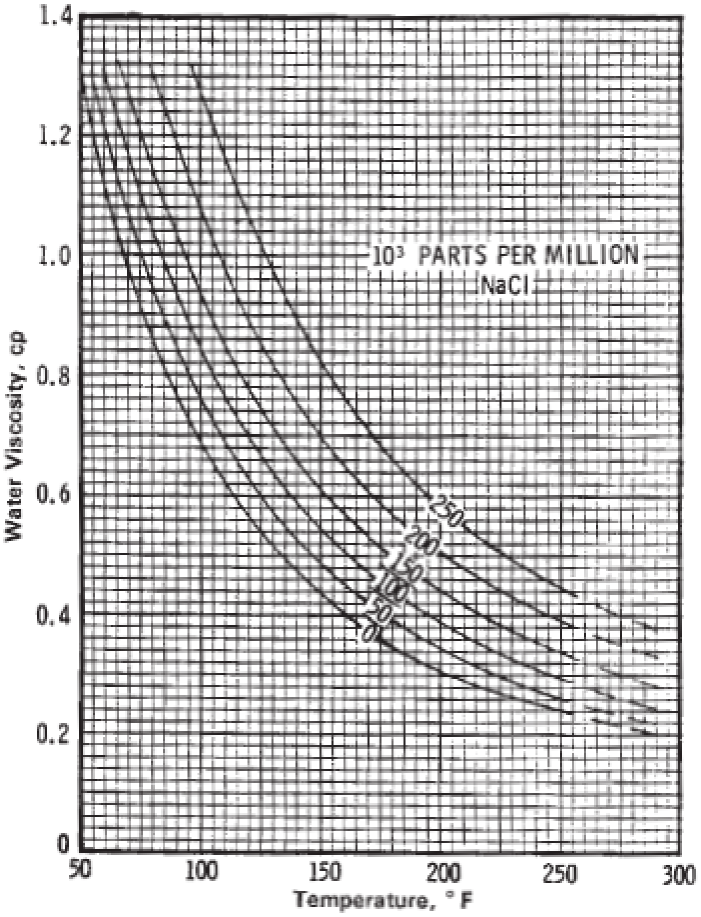


Figure 4-5 NaCl brine as a function of temperature and salinity (Whitson, Brulé, & Engineers, 2000)

4.3 INJECTIVITY PROBLEMS

4.3.1 Hydrates

Formation of hydrates is a well-known problem for hydrocarbon gas production or injection. CO_2 injection may also form hydrates under certain conditions. Temperatures up to 10 °C with pressures greater than 45 bar are conditions where it is possible for CO_2 hydrates to form. These conditions typically occur under depressurization, where the CO_2 cools, such as in valves and chokes. This may also be the case if CO_2 is transported along the sea bed, where hydrates precipitates and clogs the pipe (Stalkup, 1983). Hydrates have been known to form with original reservoir temperatures as high as 27 ºC (Mizenko, 1992), which may become a problem for CO_2 injection.

These are problems that may occur in reality, but were not considered in the simulations in this report.

4.3.2 Corrosion

Corrosion from CO_2 has generally not been a major problem in the petroleum industry, with dry gas being transported along surface pipelines. It is when CO_2 comes in contact with water it becomes corrosive, which may be a problem when CO_2 is injected into the formation. When CO_2 comes in contact with brine it hydrates from CO_2 to H_2CO_3 which is a weak acid, and may corrode the production tubing and casing if measures are not taken to protect these. Normally production tubing and casing are made of black steel, and a way to protect them from corrosion is to add chrome into the steel (Yevtushenko et al., 2014), (Eiken et al., 2011).

5 FIVE SPOT WELL PATTERN

Five-spot well pattern, Figure 5-1, is defined as an injection pattern with four injection wells located in the corners of a square and a production well in the center (David Martin & Colpitts, 1996). For an infinite extent of this type of well pattern the producer/injector ratio will be equal to 1.

A five-spot pattern is a common well configuration in oil and gas production, as it is simple to use, and gives good results. Having a strict well pattern, however, is most common in onshore reservoirs due to well cost. Onshore drilling is usually relatively inexpensive, and it is easy to find a suitable place to drill. If the formation drilled is located offshore it is often expensive to drill a well, and well locations are normally dependent on geological factors to optimize production/injection.

When doing calculations on injection a five-spot pattern is beneficial. By assuming a non-dipping and homogenous reservoir in horizontal extent it is valid to assume that all quadrants are equal, and only one quadrant needs to be analyzed, Figure 5-1. After the one quadrant have been analyzed all quadrants are multiplied, and the reservoir is considered as a whole.

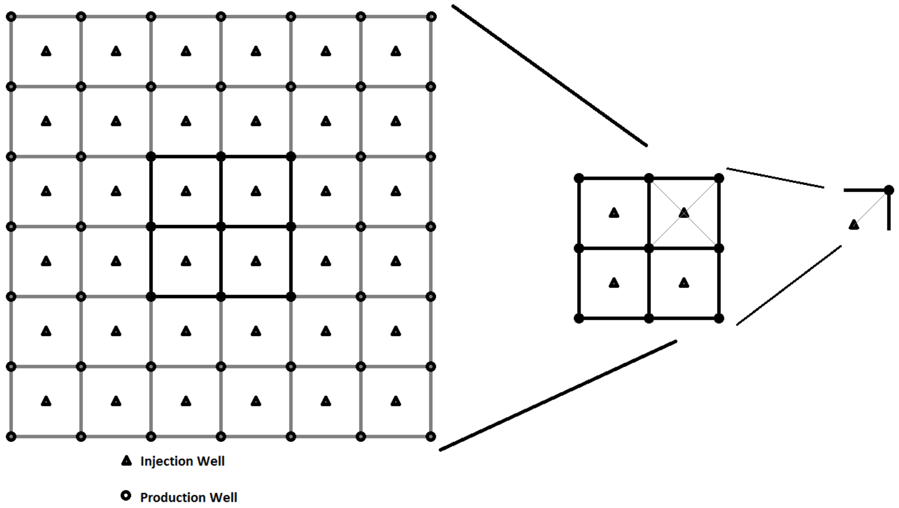


Figure 5-1 Typical well placement in a five spot well pattern, narrowing down to a single quadrant.

6 MODELLING CO_2 INJECTION WITH A BLACK OIL SIMULATOR

To model the injection of CO_2 into a saline aquifer a reservoir simulator is needed. Several reservoir simulators exist and may be applied. Special simulators for CO_2 storage exist in addition to conventional reservoir simulators designed for oil simulations. Shariati Pour, Pickup, Mackay, and Heinemann (2012) found that black - oil simulators such as eclipse 100 and eclipse 300, are of sufficient accuracy for CO_2 – brine simulations and faster than compositional simulators, if accurate pVT tables are used to represent CO_2 and brine.

To adapt black - oil simulators to CO_2 storage the oil phase is used to represent brine, and gas phase presents CO_2 (Mo & Akervoll, 2005). This makes the simulator more flexible than if only water and hydrocarbon gas is present, especially with respect to solution of CO_2 into brine and vice versa.

A black oil simulator, eclipse 100, have been used in the simulations conducted in this report. Oil was given water properties to make the simulator more flexible, while gas was CO_2. This gave a simple two phase model with brine as the oil component and CO_2 as the gas component. The water component was given the same properties as the oil component (brine properties) and was placed far below the relevant depths, to not interfere with the two phase system.

In the simulations done in this report solution of CO_2 into brine, and solution of brine in CO_2 has been neglected since it will have a negligible effect on total storage in the time range injection is reasonable.

7 PROXY MODELLING

A reservoir simulator is a powerful tool used to visualize reservoir behavior over time. The simulation model has a lot of input and is very extensive. Proxy models may be defined as mathematically or statistically defined functions that replicate simulation output for selected input parameters, (Zubarev, 2009).

A proxy is a substitute for a reservoir simulator, and is used to make simplified simulations which is less resource demanding. The accuracy of the applied proxy is not as high as the simulator, but with sufficient data to build the proxy model on, acceptable accuracy may be achieved. When using a proxy it is possible to run through many datasets quickly for sensitivity analyses. Several types of proxy-models exist for reservoir simulation, varying in complexity and accuracy. Common features for all proxy models is the workflow in how they are designed, which is briefly described in Figure 7-1.

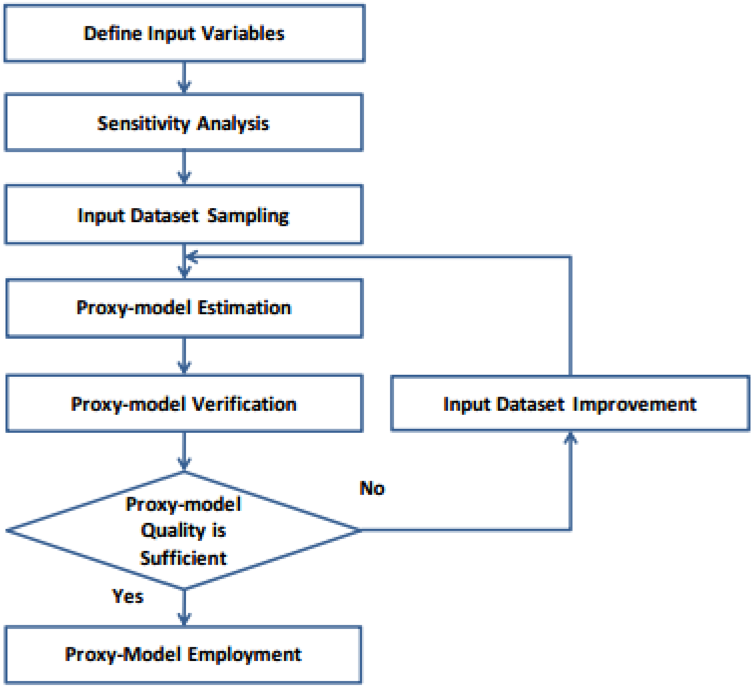


Figure 7-1 Workflow for proxy – modeling (Zubarev, 2009)

Zubarev (2009) described four common types of proxy models in the petroleum industry:

- Polynomial regression
- Multivariate kriging model
- Thin-plate splines model
- Artificial neural network

7.1 POLYNOMIAL REGRESSION MODEL

Polynomial regression models was first used as a tool for analysis of physical experiments, and later adopted into computer experiments. This type of proxy-model does not approximate exactly the experimental data. It has been widely used in the petroleum industry due being easily understandable, flexible and computationally economic.

7.2 KRIGING MODELS

Kriging was originally developed for use in geostatistics, but the method have proven useful in other areas, such as proxy modelling. Krigings proxy-models are based on the geostatical technique for spatial correlation of an arbitrary parameter called kriging. The model is based on interpolation which is modeled from covariances from the last point. When created it exactly replicates the initial data sample, which makes them attractive for computer experiments.

7.3 THIN-PLATE SPLINES MODEL

Thin-plate splines is an interpolation method that finds a minimally bended smooth surface that passes through all given data points This type of proxy-model replicates the input data exactly, but there has to be more experiments than uncertainty parameters to be applicable. Thin plate spline proxy-model involves of two parts: a global approximation regression function and a radial basis function that define a spatial mapping between two points in space.

7.4 ARTIFICIAL NEURAL NETWORK

An artificial neural network is an imitation of a biological neural systems, which may be found in the brain. This proxy-model is built from nodes and each of the nodes receives signals from neighboring nodes and processes them to generate a particular output. A schematic of an artificial neural network work process is illustrated in Figure 7-2. The number of hidden layers and nodes affects the ability of the neural network to produce different degrees of non-linearity. The number of nodes is restricted by the number of experiments.

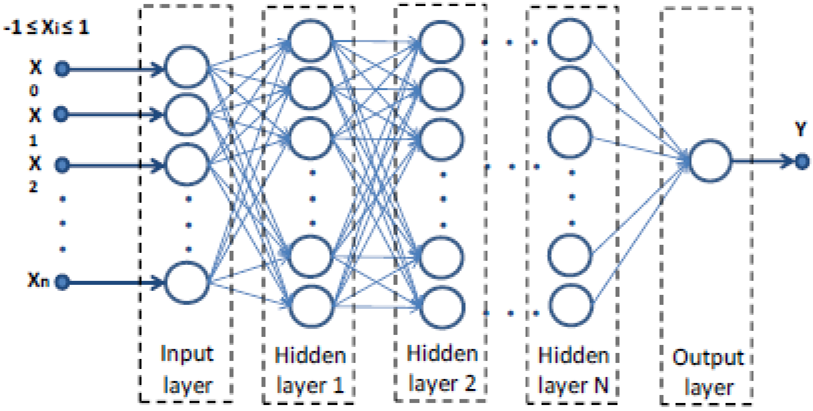


Figure 7-2 Schematic of Artificial neural network (Zubarev, 2009)

8 POLYNOMIAL REGRESSION MODEL

A polynomial regression model was chosen as the base for the proxy in this thesis. Jurecka (2007) explained the set-up of the regression model.

A polynomial regression model is based on fitting free parameters P of function

$$y = \eta(\mathbf{x}, \mathbf{P}) + \varepsilon \tag{8.1}$$

to observed values. ε is the error between observed values and calculated values. This type of error is not possible to rule out, but it may be made small enough to not have a significant impact on the result. The regression function η is typically a linear function, but may be of higher orders, with linear addressing of the regression coefficients $\mathbf{P}$

$$\eta(\mathbf{x}, \mathbf{P}) = \sum_{j=1}^{n_P} \mathbf{P}_j \eta_j = \hat{\mathbf{P}}^T \boldsymbol{\eta}(\mathbf{x}) \tag{8.2}$$

Here the function $\eta(\mathbf{x}, \mathbf{P})$ is the sum of a pre-determined set of η_P linearly independent functions $\eta_j(\mathbf{x})$ called regressors. The regressors are multiplied with a respective scalar P_j. Using matrix notation the regressors can be assembled into a vector,

$$\boldsymbol{\eta}(\mathbf{x}) = [\eta_1(\mathbf{x}), \eta_2(\mathbf{x}), \eta_3(\mathbf{x}), \ldots, \eta_{n_P}(\mathbf{x})]^T \tag{8.3}$$

With the vector $\tilde{\ }$ of size $m \times 1$ containing observed values at sampling points $\mathbf{x}^l$, where l ranging from $l = 1....m$, equation (8.3) inserted into equation (8.2) at $\mathbf{x}^l$ is written as

$$\tilde{\ } \tag{8.4}$$

Where $\mathbf{F}$ is a matrix containing the individual regressors at respective data points.

$$\mathbf{F} = \begin{pmatrix} \eta_1(\mathbf{x}^1) & \cdots & {}^{1}) \\ \vdots & \ddots & \vdots \\ \eta_1(\mathbf{x}^m) & \cdots & {}^{m}) \end{pmatrix} \tag{8.5}$$

Vector $\mathbf{e}$ (size $m \times 1$) is the error term between observed values, $\tilde{\ }$ and calculated values $\eta(\mathbf{x}^l, \mathbf{P})$, which are called residuals.

$$e_l = \tilde{\ } \qquad \mathbf{)}),\ l = 1....m \tag{8.6}$$

By assuming the function $\eta(\mathbf{x}, \mathbf{P})$ describes the observed data points precisely, the residuals $\mathbf{e}$ are from measurement error only, the residuals can be assumed to be normally distributed with a mean of zero, with no correlation and a constant variance σ^2. By using this assumption the least squares method can be used to calculate an estimation for the regression coefficients.

The least squares method estimates the regression parameters $\hat{\mathbf{P}}$ by minimizing the sum of squared residuals

$$\sum_{l=1}^{m} (\tilde{\ } \qquad \mathbf{)}))^2 = \mathbf{e}^T \mathbf{e} \tag{8.7}$$

Equation (8.4) can be transformed to a formulation for linear regression analysis

$$\min_P (\tilde{\ } \qquad \tilde{\ } \tag{8.8}$$

The minimization turns into a linear system of equations

$$-2\mathbf{F}^T \tilde{\ } \tag{8.9}$$

which can be solved for $\tilde{\ }$ if $\mathbf{F}^T \mathbf{F}$ is invertible, if there are at least as many observed data points as there are coefficients to be estimated.

$$\tilde{\ } \qquad {}^{-1}\mathbf{F}^T \tilde{\ } \tag{8.10}$$

Together with the relationship $\eta(\mathbf{x}, \mathbf{P})$, the coefficient $\hat{\mathbf{P}}$ define the global relationship

$$\hat{y} = \hat{f}(\mathbf{x}) = \eta(\mathbf{x}, \tilde{\ } \qquad \underset{j=1}{\overline{}} \ \eta_j(\mathbf{x}) = \hat{\mathbf{P}}^T \boldsymbol{\eta}(\mathbf{x})\,. \tag{8.11}$$

For engineering purposes it is desirable that the function can be represented as a polynomial. Higher order accuracy is obtained by using higher order polynomials. For most applications it is sufficient to use a second order polynomial for the approximation, as represented in equation (8.12)

$$\eta(\mathrm{x},\mathrm{P}) = P_0 + \sum_{i=i}^{n} P_i x_i + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} P_{ij} x_i x_j \ . \qquad (8.12)$$

9 DIMENSIONLESS PARAMETERS

To create input parameters for the regression model dimensionless parameters are used. Dimensionless parameters are beneficial to use because a dimensionless approach generalizes the problem. A single dimensionless solution may define many dimensional solutions. This greatly reduces the amount of simulations required to give a sufficient representation of the problem. Dimensionless parameters are independent of the scale of the system, making it possible to represent many different physical systems, making scaling simpler.

Dimensionless equations also helps deducing importance of variables used. Making it simpler to find variables that are insensitive to changes applied, or highly sensitive variables.

9.1 DIMENSIONLESS EQUATIONS

Input parameters are scaled dimensionless to reduce the number of parameters required to create the regression model. The derivations of the equations can be seen in appendix A.

The different equations used in the polynomial regression model are:

Relative permeability water equation (9.1) and relative permeability gas equation (9.2), where the interesting parameters are α_w and α_g , which define the shape of relative permeability curves for water and gas.

$$k_{rw}\left(S_w\right)=\left(\frac{S_w-S_{wrg}}{1-S_{wrg}}\right)^{\alpha_w} \tag{9.1}$$

$$k_{gr}\left(S_w\right)=k_{rg}^0\left(\frac{1-S_w}{1-S_{wrg}}\right)^{\alpha_g} \tag{9.2}$$

To describe a relationship between aquifer extent and vertical permeabilities the aspect ratio, equation (9.3), is defined.

$$R^2=\frac{L^2k_v}{H^2k_h} \tag{9.3}$$

Mobility ratio, equation (9.4), is the dimensionless ratio describing how easily the different fluids flow relative to each other. For a water-wet drainage scenario the k_{rw}^{0} will be equal to 1.

$$M = \left(\frac{\frac{k_{rg}^{0}}{\mu_g}}{\frac{k_{rw}^{0}}{\mu_w}} \right) \tag{9.4}$$

To define the significance of capillary forces compared to viscous forces, the capillary number, equation (9.5), is used. In this scaling of capillary pressure the entry capillary pressure is used as P_c^*.

$$N_{cv} = \frac{P_c^*}{\Delta p} \tag{9.5}$$

Gravity number is defined as the significance of gravitational forces versus viscous forces and is described in equation (9.6)

$$N_{gv} = \frac{\Delta \rho g H}{\Delta p} \quad . \tag{9.6}$$

The ratio between perforated injector and produced is scaled dimensionless through equation (9.7). Where the injection well is perforated though the entire height, H, of the aquifer.

$$\xi_{prod} = \frac{L_{prod}}{H} \tag{9.7}$$

To describe well radii dimensionless equations (9.8) and (9.9) are used. This can be simplified by setting $r_{inj} = r_{prod}$, making $\eta_{inj} = \eta_{prod}$.

$$\eta_{inj} = \frac{r_{inj}}{L} \tag{9.8}$$

$$\eta_{prod} = \frac{r_{prod}}{L} \tag{9.9}$$

To define a dimensionless gas-oil ratio equation (9.10) can be used.

$$\chi = \left(c_w + c_r\right)\Delta p \tag{9.10}$$

The time it takes for the producer to reach a given gas-oil ratio is scaled dimensionless through equation (9.11)

$$t\left(\vec{C}_{NUM}, \zeta\right) = \frac{\Delta p k_h t^*\left(\zeta\right)}{L^2 \mu_w \varphi\left(1 - S_{wrg}\right)} \ . \tag{9.11}$$

While the injected gas/produced water at given time is scaled dimensionless through equation (9.12)

$$Q_D\left(\vec{C}_{NUM}, \zeta\right) = \frac{Q_g\left(t^*\left(\zeta\right)\right)}{\varphi\left(1 - S_{wrg}\right) L^2 H} \ . \tag{9.12}$$

A square, homogenous and non-dipping aquifer with one injection and one production well can be described through these dimensionless equations.

10 ECLIPSE INPUT

Eclipse does not take the dimensionless groups mentioned in the chapter above as input parameters, and some adjustments to the input have to be made. Since dimensionless scaling is independent on which parameter changed in a dimensionless equation, the number of input variables in eclipse needed to be tested for importance is significantly reduced.

To minimize the number of simulations run it was desirable to use relevant input parameters only. Input variables for the single quarterspot were tested independently to find their significance on total stored CO_2 at breakthrough. Also the validity of using a quarter spot model compared to a full five-spot model had to be proved.

10.1 USE OF A QUARTER SPOT MODEL

In the five spot model it is assumed to be only a quarter of the injection well feeding each quarter of the aquifer, and the injection well is placed exactly in the center between the injectors, between the corners of four grid blocks. In the quarterspot model, however, the injector is placed in the middle of the grid block belonging to that quadrant of the aquifer. This will in principle give an error in both well radii and for the grid blocks. To check if it is valid to use a quarterspot model with equal injection and production well radii the quarterspot model was inverted (producer and injector changed places) and a simulation with a full five spot was conducted. The results is presented in Table 1 for a low rate model (dp =40bar) with breakthrough when 100Sm3/d CO_2 is produced. The difference between the full five spot and quarterspot model, and the quarterspot with different well radii, is small enough to be neglected. Thus the use of a quarterspot model is representative.

Table 10-1 Comparing injected CO_2 at breakthrough for full five spot, a corresponding quarterspot and a quarterspot with an injection well radii divided by four

Comparing Full five spot model and quarterspot model				
	Breakthrough time [days]	Inj [sm3]	Inj 1/4 [Sm3]	Difference [%]
Full five spot	1890	6185055000	1546263750	
Quarterspot equal well radii	1890	1570218000		1.53 %
Quarterspot $r_{rw,inj} = 1/4r_{w,prod}$	1890	1567851000		1.38 %

10.2 CAPILLARY PRESSURE

In reality it is reasonable to assume that relevant rocks for CO_2 storage are mostly water wet, and will have a capillary pressure when injecting CO_2. For the model to be as simple as possible it was important to test the different parameters for their significance on simulations. The capillary pressure was tested for its significance by applying a capillary pressure derived from a Leveret – J curve based equation (10.1) on CO_2/brine and Berea sandstone from Pini and Benson (2013) (Figure 10-1). The capillary pressure has a significant impact on stored CO_2 as seen in Figure 10-2 and in Figure 10-4.

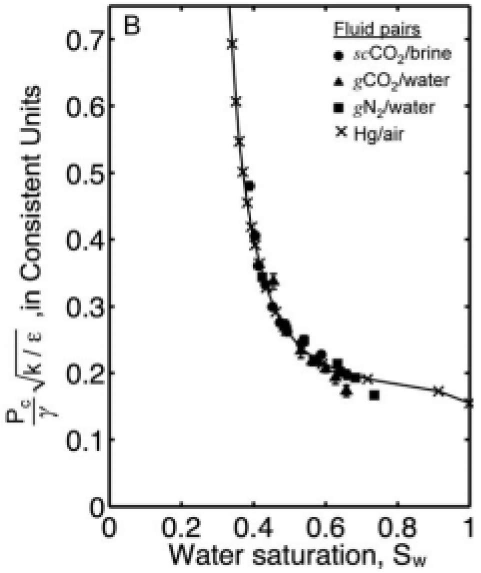


Figure 10-1 Leveret-J function capillary pressure is basesd on. (Pini & Benson, 2013)

The Leveret-J equation

$$P_c(\text{J}) = \frac{P_c}{\gamma}\sqrt{\bar{k}\,/\,\varphi} \tag{10.1}$$

where

$$\bar{k} = \sqrt{k_v\,\text{k}_h} \tag{10.2}$$

required the interfacial tension for different salinities and pVT properties, which was found from Li, Boek, Maitland, and Trusler (2012) and Stefan Bachu and Bennion (2008), while the permeability was averaged as geometrical permeability.

The capillary pressure proved rather insensitive to the interfacial tensions relevant for applied salinities and pressures/temperatures, Figure 10-2, thus all interfacial tensions was averaged to 35mN/m. On the other hand capillary pressure proved rather sensitive to permeabilities, and a geometrical average between vertical and horizontal permeability was used. The capillary pressure curve was then approximated by the Brooks-Corey equation (10.3), to make it dependent on pore size distribution index, λ, and to extrapolate it to desired values of S_{wi}=0.2. The λ parameter was set constant to 1.16 to match the calculated capillary pressure from the Leveret – J curve, while the entry pressure, P_e, varied dependent on permeability. An illustration of capillary pressure can be seen in Figure 10-3

$$P_c = P_e(\frac{S - S_{wr}}{1 - S_{wr}})^{-\frac{1}{\lambda}} \tag{10.3}$$

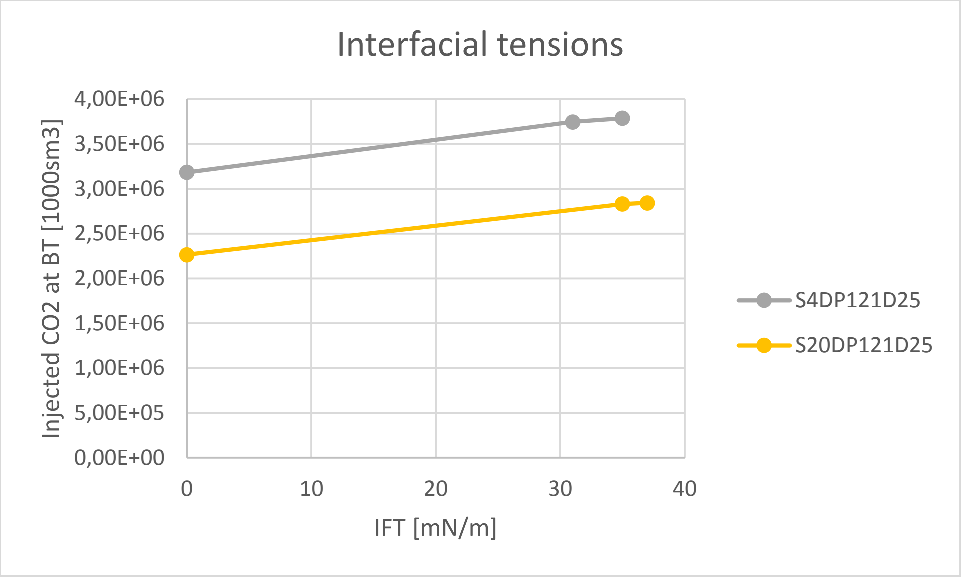


Figure 10-2 The difference in injected CO₂ at breakthrough for different interfacial tensions. IFT = 0 represents zero capillary pressure.
S = salinity [wt%], DP = pressure drop [bar] and D = depth [100m].

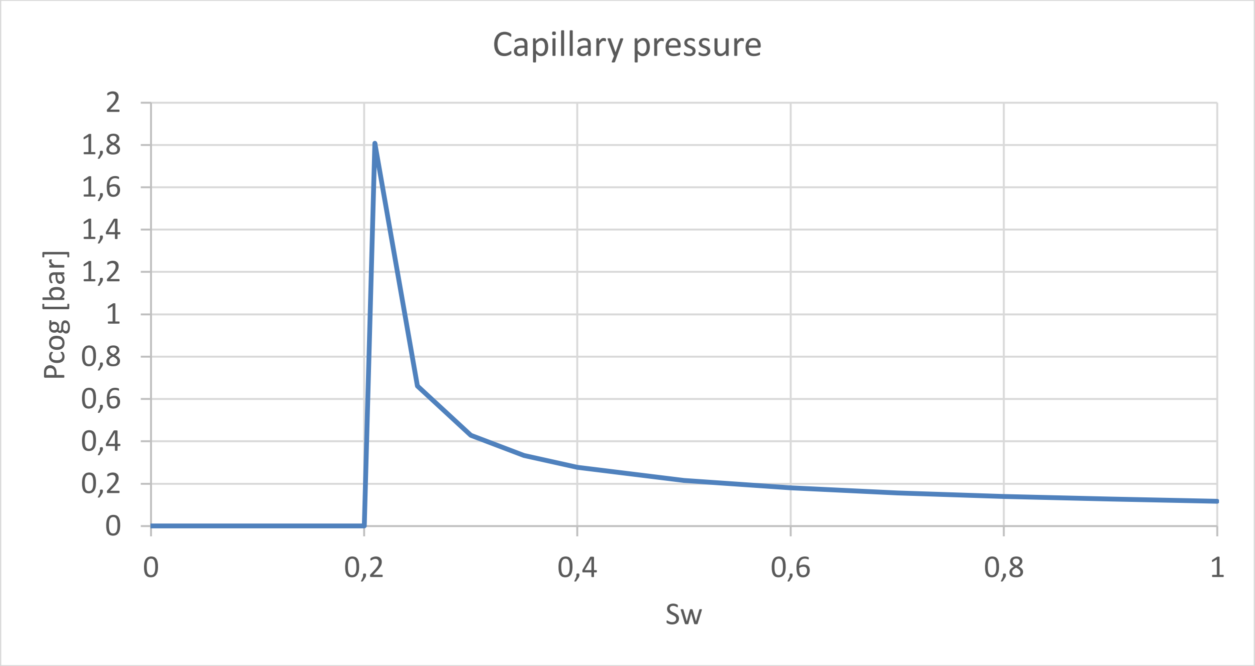


Figure 10-3 Capillary pressure curve for Kv=5mD, λ=1.6 and Pe=0.117bar.

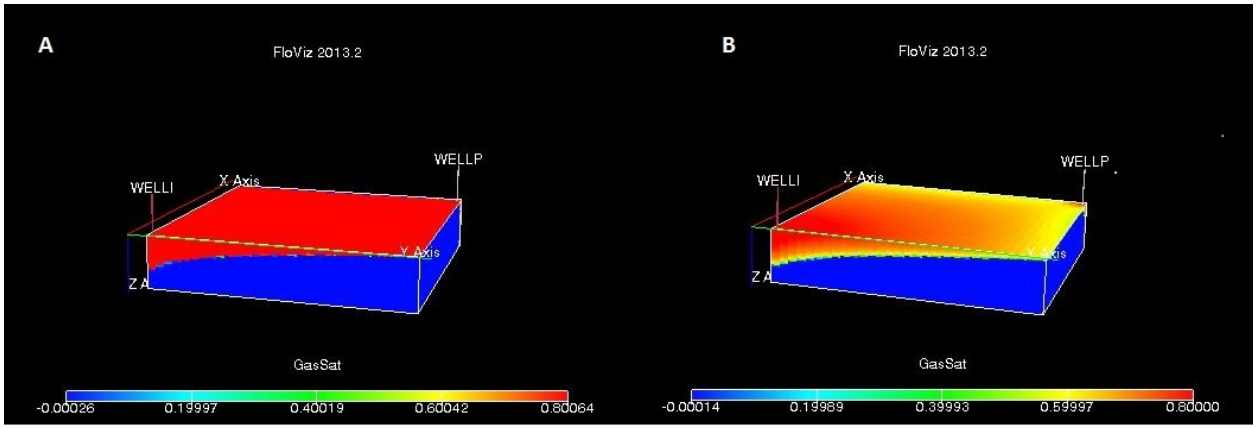


Figure 10-4 The difference between no capillary pressure and with capillary pressure, where A is without cap. pres. and B is with cap. pres. The color scale show gas saturation.

10.3 Relative permeability

Relative permeability curves are based on the corey equations (9.1) and (9.2), and are dependent on the corey exponent, water saturations and residual water saturation. Corey exponents determine the shape of the relative permeability curve, and are important for how CO_2 and brine flow relative to each other.

Since the residual saturations will be scaled, the dimensionless groups will only depend on corey exponents. Therefore to prove the significance of the corey exponents several runs was carried out with different corey exponents, with some of them represented in Figure 10-5. From Figure 10-5 it is clear that the gas exponent, CG, is more important than the water exponent, CW, but still both have significant impact on total amount of stored CO_2. The high dependency of stored CO_2 on Corey gas exponent may be due to the segregation of gas which goes slower when the gas rel. perm. curve gets steeper.

Corey exponents depend on rock properties, and are measured parameters from core samples found in Krevor, Pini, Zuo, and Benson (2012). An illustration of a relative permeability curve used in simulations are presented in Figure 10-6

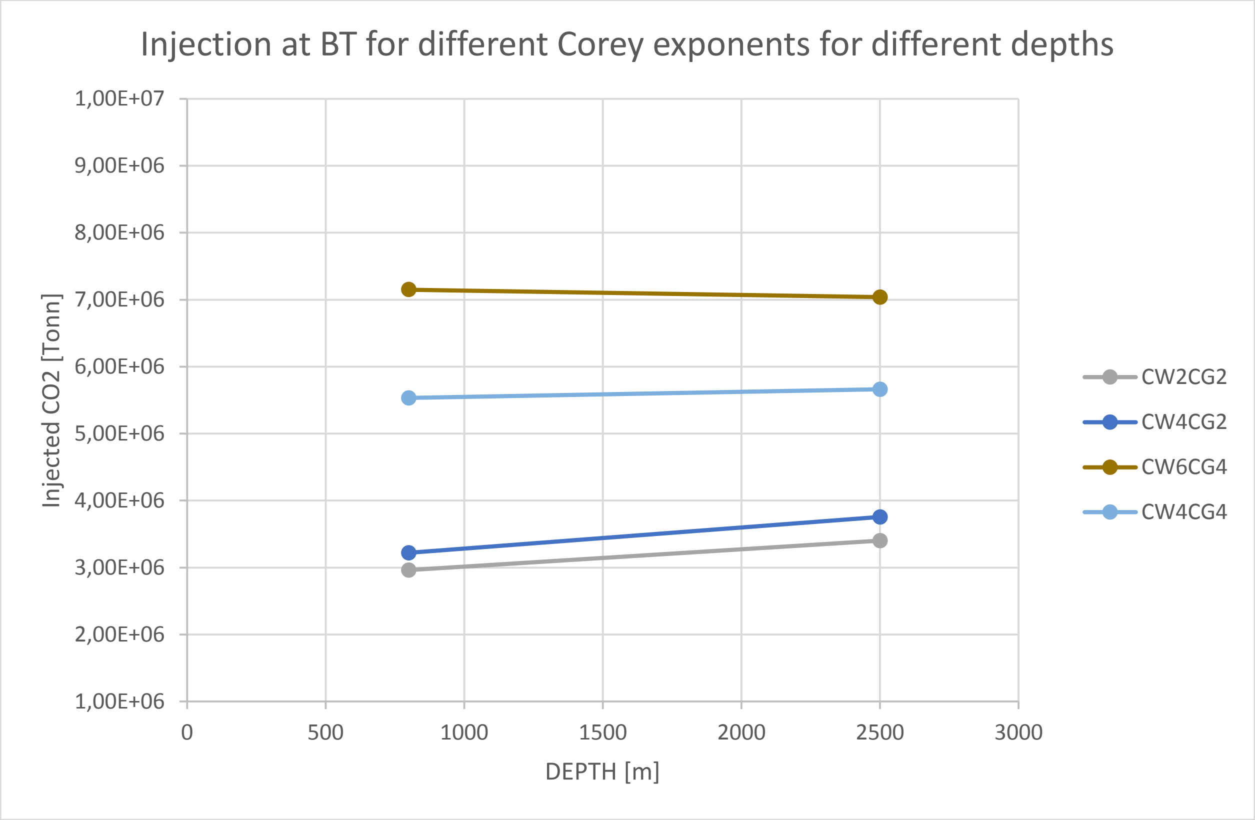


Figure 10-5 Recovery as a function of depth and corey exponents.
CW is water exponent, CG is gas exponent. Lines between points are to illustrate same Corey exponents.

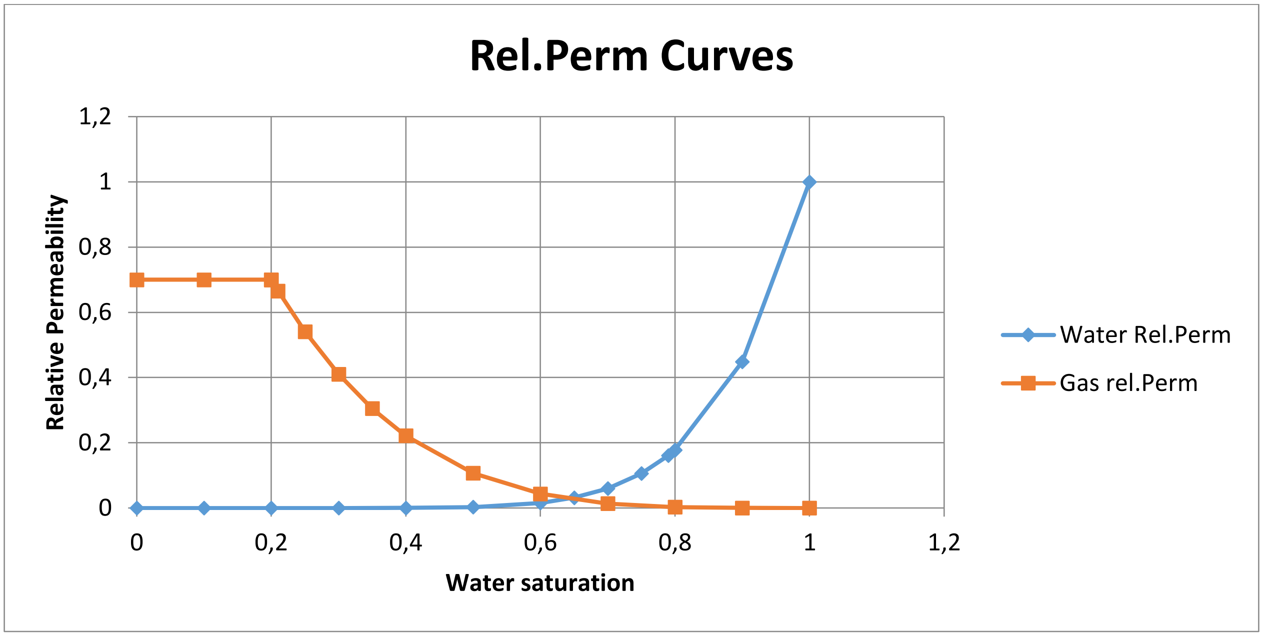


Figure 10-6 Relative permeability curves for CW=2 CG=2

10.4 Gridding

The grid block size is not a physical parameter, but may have impact on the flow pattern in a model. When a grid block is introduced to a new fluid, this fluid instantly distributes homogenously in the whole grid block, contrary to in reality where the fluid is gradually introduced to new rock volumes. This is a non-physical error, and is corrected by increasing the number of grid blocks. Increasing the number of grid blocks however, increases the simulation time.

The model is mainly sensitive to changes in grid z-direction, and barely in x and y-direction. Sensitivities to number of grid blocks in z-direction is illustrated in Figure 10-7. When the model stabilizes it is assumed to be at near-realistic conditions for fluid propagation. To keep the simulation time as short as possible it is reasonable to only increase the most sensitive area of the model, where the CO_2 propagates, which mainly is in the top of the reservoir. Only the top 6m of the aquifer was given finer grid in z-direction, the rest remained constant.

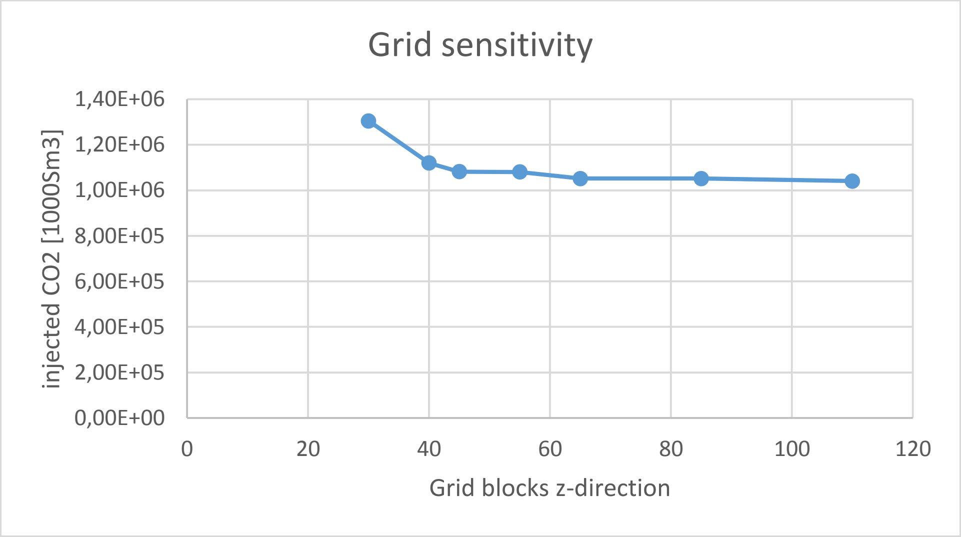


Figure 10-7 Grid sensitivity, with progressively decreasing grid block size the last 10 meters of the aquifer

10.5 DEPTH, TEMPERATURE AND BOTTOM HOLE PRESSURES

Depth is affecting both pressure and temperature in the aquifer. As depth increases both pressure and temperature increases, affecting density and viscosity of the brine and CO_2. Temperature gradient was set constant to 0.03 °C/m, which is a reasonable assumption for parts of the North Sea, (Baird, 1986), and a common assumption in many cases. A lithostatic gradient of 2.5bar/m, with a safety margin of 0.6 was assumed to calculate maximum allowable BHP. Another BHP was calculated by taking the average between lithostatic (with safety margin) and hydrostatic pressure gradient.

The producing well was a passive well with a BHP equal to initial pressure, thus the well only produces when the aquifer is subjected to a higher pressure than initial pressure.

Increasing injection pressure resulted in faster migration of CO_2 and shorter breakthrough times. Contrary to expected an increase in stored CO_2 was observed, as seen in Figure 10-8, even with an earlier breakthrough of CO_2 for increasing pressure.

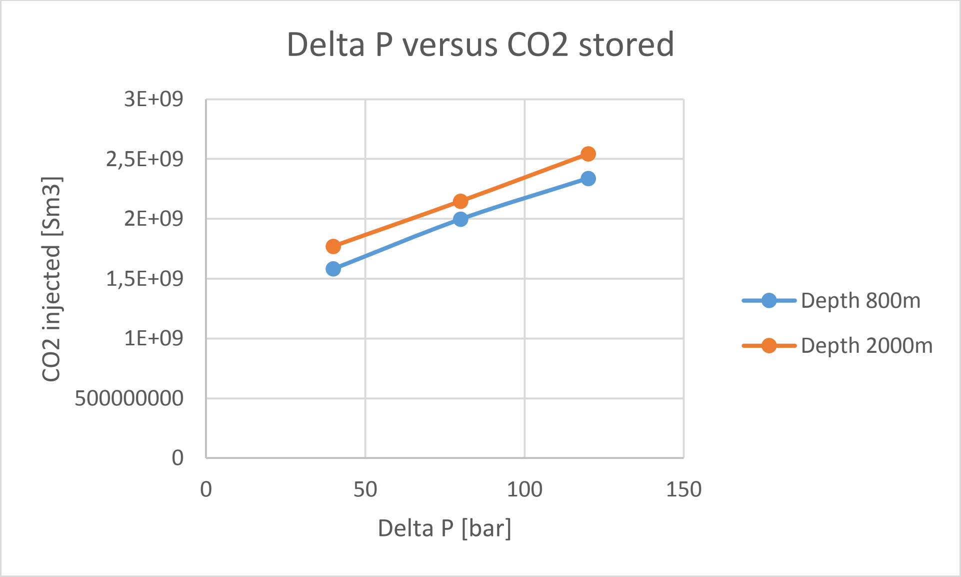


Figure 10-8 Illustrating significance of increased pressure difference between Injection well and production well for two different depths.

10.6 Permeability

Variations in absolute vertical permeability can be proven to have a significant impact on storage capacity as seen in Figure 10-9. Increasing vertical permeability reduces stored CO_2. This may be due to gas segregation happens faster, as it is easier for the CO_2 to flow up towards the aquifer seal.

Absolute permeability will affect the aspect ratio. The aspect ratio, equation (9.3) is dimensionless and indifferent to which of the permeabilities changed. Aspect ratio is also dependent on aquifer height and length. In the model only vertical permeability is changed, to mimic flow restrictions in vertical direction. Aquifer height is also varied, while length of aquifer and distance between wells are kept constant.

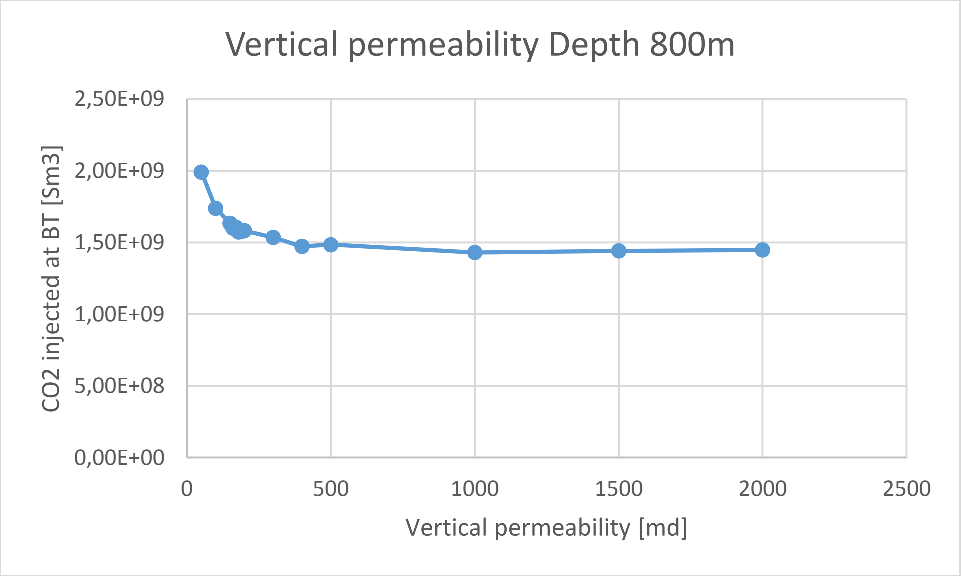


Figure 10-9 Significance of variations in vertical permeability for stored CO_2 at breakthrough at depth 800m and with constant dp = 40bar

10.7 Perforations

Due to dimensionless scaling the model uses the ratio between the length of the perforated intervals between producer and injector equation (9.7). When using a dimensionless parameter as this, it is not important which of the wells that changes in perforated length.

Only the production well perforation intervals have been changed in the simulation runs in this report, and only in horizontal direction, to expose injected CO_2 for volume as large as possible, before breaking through into the production well. To test well perforations significance, several runs with different perforation length in x and y direction were made, as can be seen in Figure 10-10. The length of the perforations have significant impact on total injection, and is an important parameter to consider. An illustration on how the perforations in the horizontal production wells are set up can be seen in Figure 10-11. For horizontal wells both x and y direction are perforated in the specified number of grid blocks.

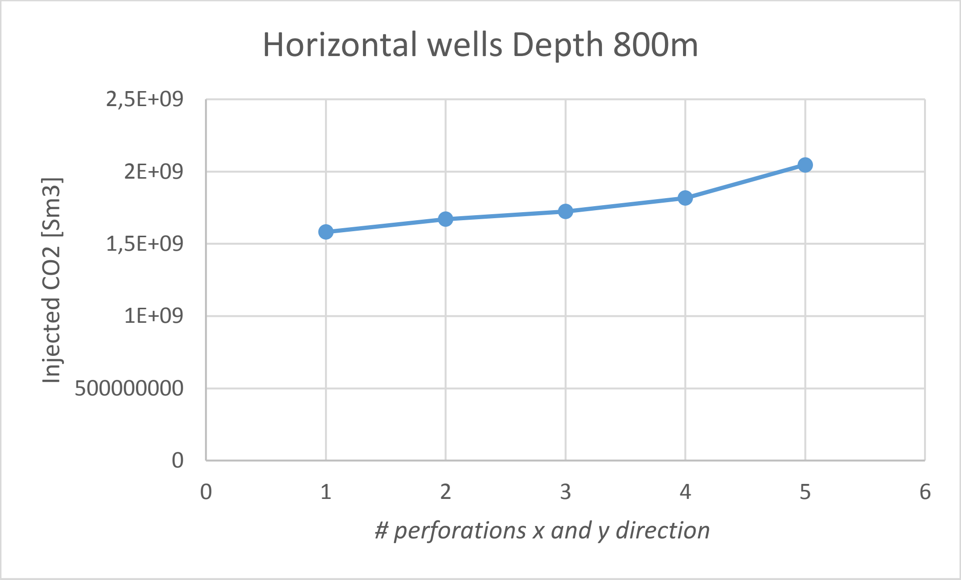


Figure 10-10 Injected CO_2 at breakthrough for different perforation intervals (equal for both x and y direction) at depth 800m and with constant dp = 40bar

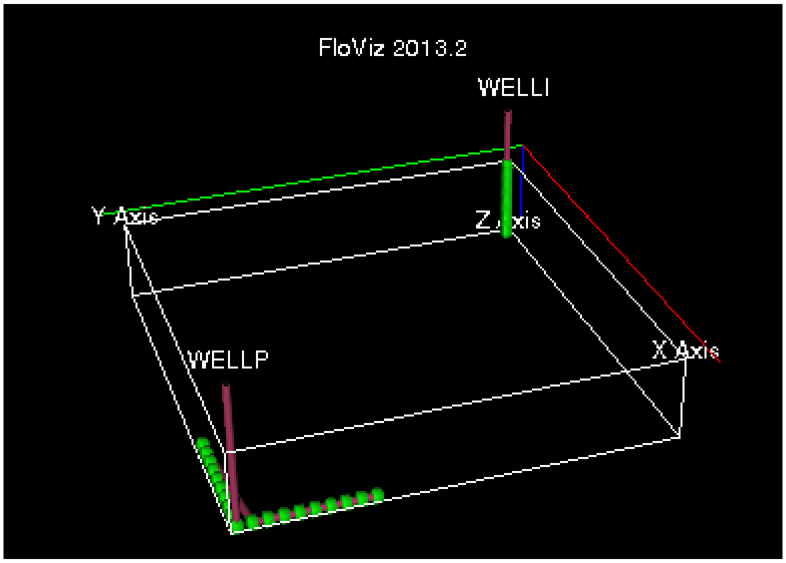


Figure 10-11 Illustration of horizontal perforations. In this figure 10 grid blocks are perforated in x and y direction in the production well. While the injection well is perforated vertical through the entire aquifer length.

10.8 AQUIFER AREA

To describe the size of the aquifer, aspect ratio was used as a dimensionless parameter equation (9.3). Aspect ratio only regards aquifer height and length, and to make simulations simpler it is preferable to keep distance between injection and production wells constant, thus aquifer height was the only parameter changed. Grid block size was kept constant to keep perforation intervals the same, and the increase in grid resolution was kept constant at the top 6m of the aquifer.

Increasing aquifer height has an important effect on injected CO_2, as increasing aquifer height increases stored CO_2 before breakthrough.

10.9 Salinity

Salinity is important for viscosity and density to brine. It is common for brine to become more saline the deeper it is, but this is not necessarily the case. Salinity depends on the formation the brine is located in, and may also vary with depth and location. When brine becomes more saline it increases in density and viscosity which is unfavorable for the storage potential of an aquifer. Figure 10-12 show the total injected CO_2 dependency on salinity to the brine. It was clear that salinity to the brine has a significant impact on total CO_2 storage, and has to be considered.

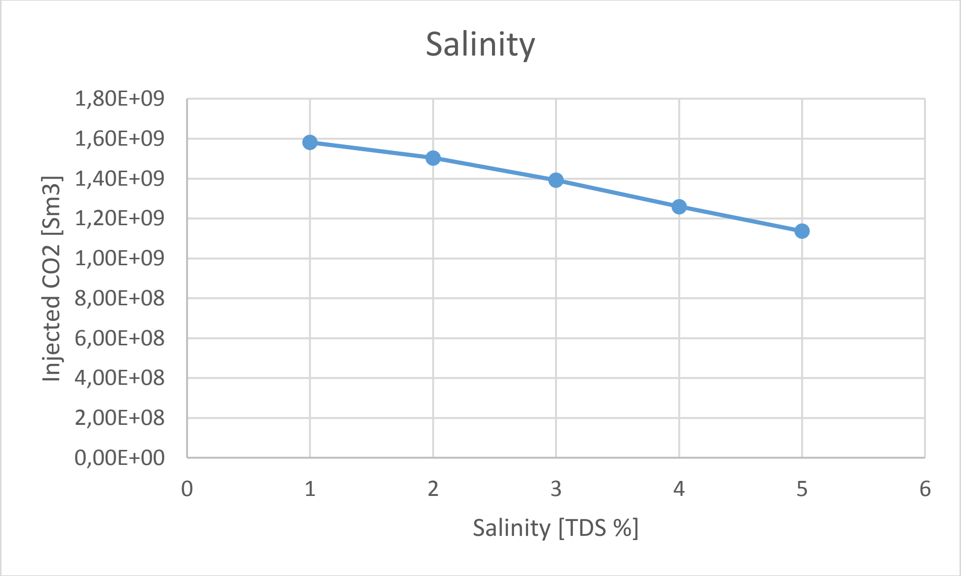


Figure 10-12 Total injected CO_2 at breakthrough dependence on salinity

11 RUNNING SIMULATIONS

When the parameters that had significant impact on simulations, and is present in the dimensionless equations, were found, input parameters had to be chosen. Choices for input parameters was based on the significance each parameter had on injected CO_2, but also on having a large span of variables, to cover everything in between the parameters used.

To generate all eclipse run-files a unix bash script was used (appendix B). The purpose for the unix-script was to automatically generate all .DATA files used in eclipse simulations. Input parameters, given in Table 2, were automatically changed for each case, to make it unique and to give it a unique name. When the .DATA files was generated another unix bash-shell script was used to run all files in eclipse (appendix C). To avoid generating large amount of unnecessary data the script automatically created a copy of the desired file, moved it to another folder, and deleted the output files from each simulation. The eclipse templates used are given in appendix F. Three different eclipse templates were used. One for each aquifer height, to ensure the same vertical grid block length for all three templates.

Table 11-1 Overview of parameters used to generate data files for eclipse simulations. Note that BHP is dependent on depth and Corey exponents are dependent on each other, while the other variables are independent. Every number in each row were used to create combinations for the data files.

Depth	m	1000		1500		2000		2500	
Respective Injection pressure	bar	133	156	195.5	231	258	306	320.5	381
Salinity	wt %	4	8	12	20				
Vertical permeability	md	5	50	250	500				
Corey exponent combinations	Water	2	4	4	6				
	Gas	2	2	4	4				
Aquifer height	m	20	60	120					
Perforated grid blocks	#	1	2	3	7				
Gas oil ratio	Sm3/Sm3	0	50						

11.1 EXTRACTING RELEVANT DATA

To extract relevant data (Time, injected CO_2 and produced water) from each .RSM file a c-script was written (appendix D). The c-script read through each line and stored time, injected CO_2 and produced water for gas production rate = 1Sm^3, which was used as breakthrough for CO_2, and the last time step in each .RSM file, GOR=50. Another unix bash-shell script was written (appendix E) to use the compiled c-script for each case, and save all numbers the c-script printed into a unique text file. Further, the unix script merged all unique text files into one file, producing a text file with all cases.

12 GENERATING A REGRESSION MODEL

The regression function was defined through the dimensionless parameters from chapter 9. The equation (8.12), was calculated for each simulation, resulting in 45 linearly independent equations, that had to be solved with least squares method. To solve for all cases it was easiest to load the matrix of equations into Matlab, and let Matlab solve the equations. Solving linear equations in Matlab using least squares method can be done in a variety of ways. The simpler method is by loading all 45 linearly independent equations as a matrix, $\mathbf{F}$, the results, y , into a [1,n] vector, and use matrix derivation to find the regression coefficients $\mathbf{P} = \mathbf{F} \backslash \mathbf{y}$. Using this type of matrix derivation Matlab will automatically try to solve the equation system. If there are no exact solution to the equation system, Matlab will use least squares method to solve the equations. The Matlab script used is seen in appendix G. The output vector, $\mathbf{P}$, was the regression coefficients. Vector $\mathbf{P}$ can be written in an EXCEL spreadsheet and multiplied with the regressors, to give the estimated result $\tilde{y}$ from the observed values y . This approximation should be relatively precise for the cases above regarding the number of cases and extent of variables used.

To generate input parameters for the matrix $\mathbf{F}$ in equation (8.4) dimensionless parameters are used as input. The regressors, x_i are used to determine the input values. These values are the dimensionless groups from chapter 9. Here the regressors can be listed as follows:

$$x_1 = \alpha_w$$
$$x_2 = \alpha_g$$
$$x_3 = \ln(R^2)$$
$$x_4 = M$$
$$x_5 = N_{gv}$$
$$x_6 = N_{cv}$$
$$x_7 = \xi_{prod}$$
$$x_8 = \chi$$

The natural logarithm, $\ln(R^2)$ was used instead of R^2 as the aspect ratio because this number is often large and will have a dominant effect on the equations. By using the natural logarithm instead the aspect ratio was more in the same range as the rest of the regressors. By applying these regressor values into equation (8.12) for all observed values the matrix $\mathbf{F}$ was made. Output of the equation was the observed values y, which was stored in the solution vector $\mathbf{y}$. Each observed value (time, injected and produced) was stored in a separate solution vector, $\mathbf{y}_i$.

13 FIELD TESTING

After the proxy had been created it needed to be tested. A full field model of an aquifer was created. The field model consisted of 81 five spot well patterns. To avoid problems with injected CO_2 migrating out of the five spot pattern the five spot patterns was designed such that production wells was on the edges of the five spot pattern. Without this configuration it would have resulted in an error in the proxy model since the proxy model was based on mass balance. This resulted in 81 five spots, with 81 injectors and 100 producers with one perforated grid block as seen in Figure 13-1. The distance between production wells was set to 7350m or 21 grid blocks. For simplicity the aquifer was a square, and all of the aquifer area was used for injecting CO_2.

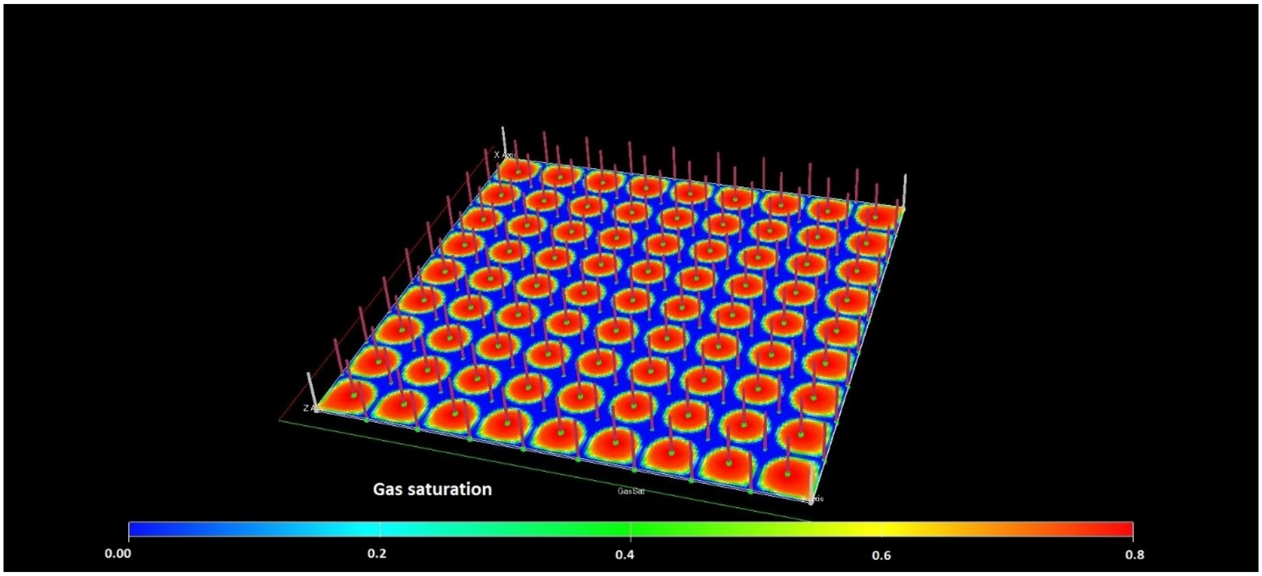


Figure 13-1 Well distribution in the full scale aquifer. The red circles represent high gas saturation around the injector well. Note the corners where gas migration is happening faster than the rest.

14 OPTIMIZATION OF INJECTION SCENARIOS

The main purpose of the proxy model created was to optimize injection strategy. Optimization of the injection strategy was done by changing the parameters that can be controlled from the surface, the operational parameters. These parameters were; bottom hole injection pressure, number of injection and production wells, perforation length for horizontal production wells, distance between wells and gas-oil ratio after breakthrough. To test the proxy model a fictive injection scenario was created. 10.000 tons/year of CO_2 was to be injected over a period of 40 years. CO_2 was converted to Sm^3 and the difference between calculated CO_2 and the CO_2 injection target was used. The optimization procedure was to use the Solver ad-in in Excel with the constraints in Table 14-1. Constraints were that the wells and perforated grid blocks had to be a whole number, injection pressure could not surpass fracture limit or go below production pressure, injection time had to be within certain limits and well distance could not be larger than the length of the aquifer.

Table 14-1 Constraints used in the Solver ad-in in Excel

Limitations for Solver		**MIN**		**MAX**
Injection pressure	bar	161		231
Number of injection wells	# wells			81
Number of production wells	# wells			100
Perforation length producer	# blocks	1		
Well distance	m			63350
GOR	Sm3/Sm3	0		
Time	years	40		60

15 RESULTS

The objective of this thesis was to create a proxy for CO_2 injection in a homogenous non-dipping saline aquifer using a quarter of a five spot well pattern. The objective of the proxy model was to give an indication of storage potential in the saline aquifer, and optimize operational parameters to maximize injection.

15.1 REGRESSION MODEL PROXY

The regression coefficients were estimated from least squares optimization in Matlab while the regressors were determined from the dimensionless equations in chapter 9.1. The regression coefficients were constant while the regressors changed for different injection scenarios.

Three different spans of linearly independent equations were created to cover all perforation lengths with a minimal error (perforation length 1, 2 and 3 & 7 grid blocks). With the regression coefficients given in Table 3, Table 4 and Table 5.

By comparing estimated values and observed values the error, ε, from equation (8.1) could be found. This error is dependent on the size of the values used, and is in itself not a good measure of the error. By dividing the error on the observed value the relative error is found. This error has the same range independently on the size of values used. The relative error plots for time, CO_2 injected and brine produced for each perforation can be seen in Figure 14-1 for perforation length 1 grid block, Figure 14-2 for perforation length 2 grid blocks in x and y direction and Figure 14-3 for perforation length 3 and 7 grid blocks in x and y direction. To find the average relative error values for each plot in the figures below the absolute values are summed together and divided on total number of cases.

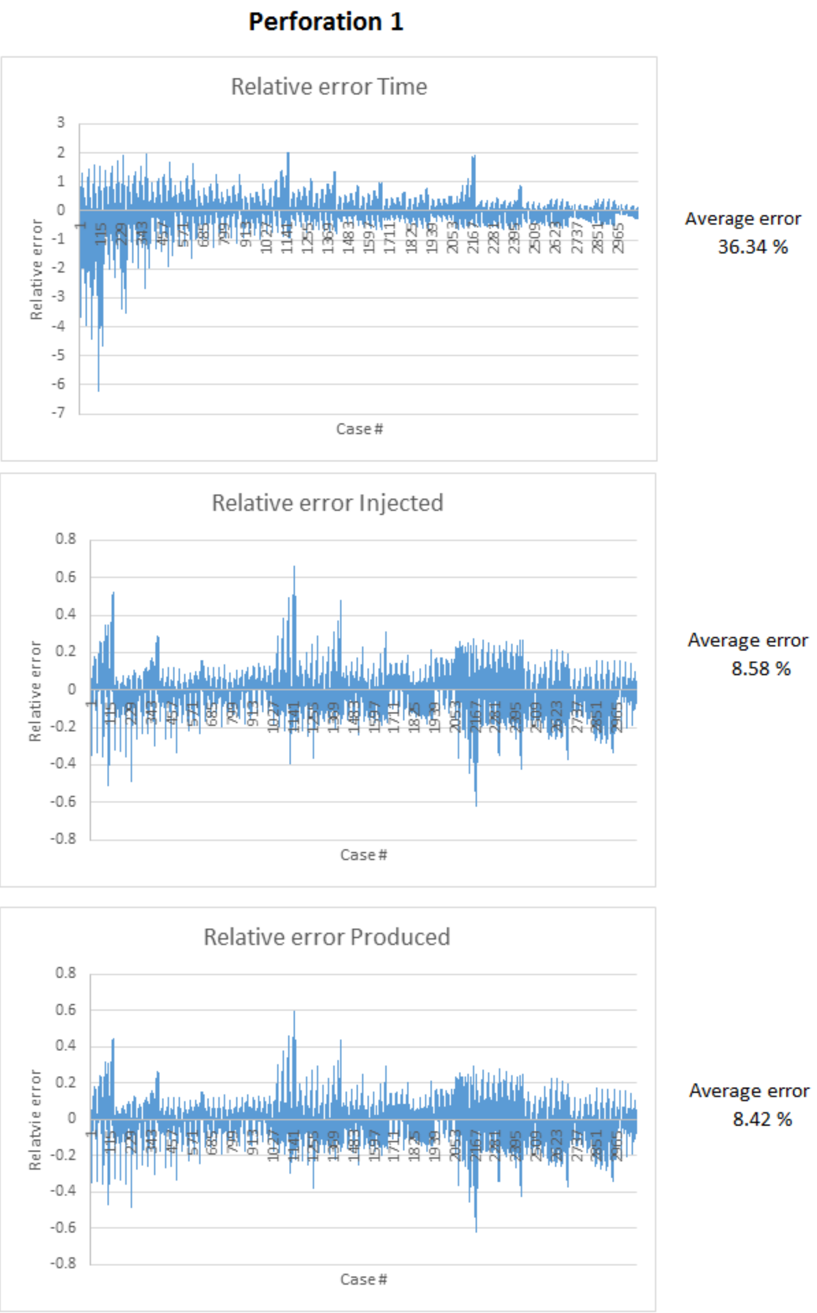
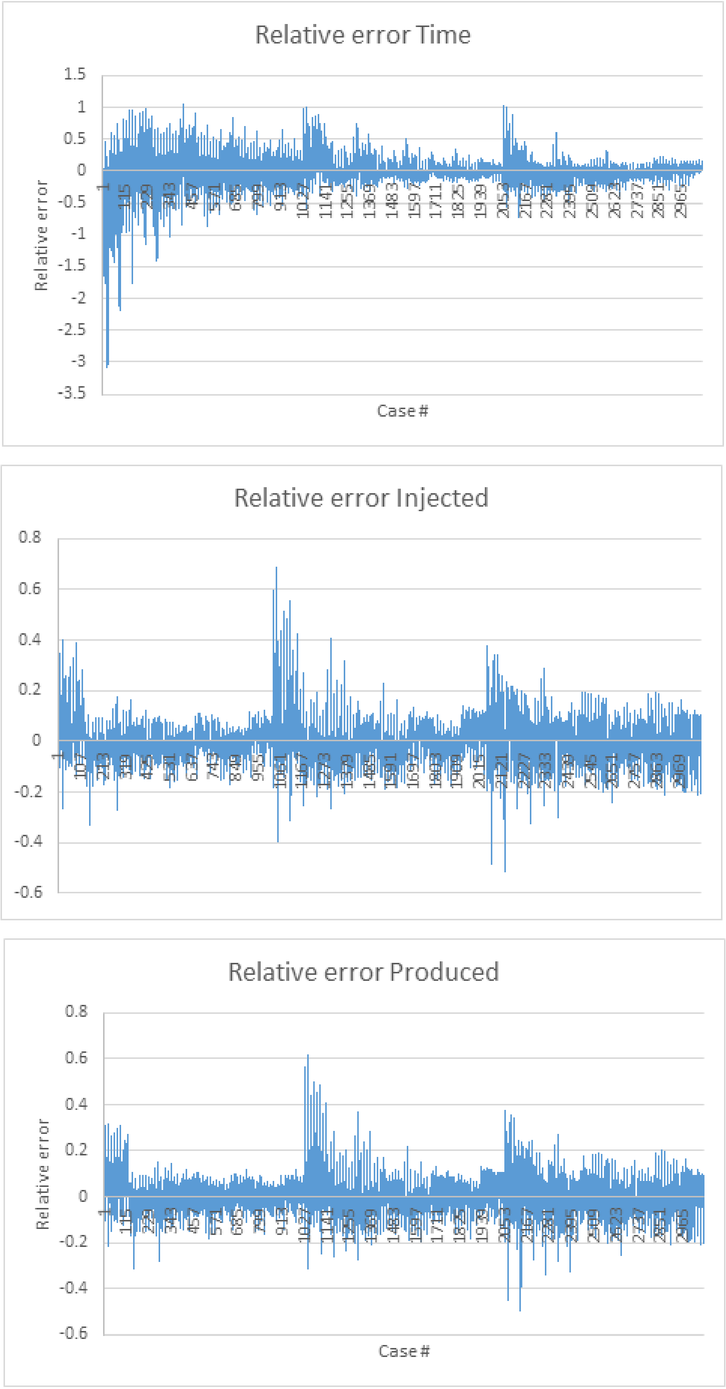


Figure 15-1 Relative error for time, injected CO_2 and produced brine with perforation length 1 grid block

Perforation 2


Relative error Time

Relative error Injected

Relative error Produced

Average error
18.31 %

Average error
7.03 %

Average error
6.75 %

Figure 15-2 Relative error for time, injected CO_2 and produced brine with perforation length 2 grid blocks

Perforation 3 & 7

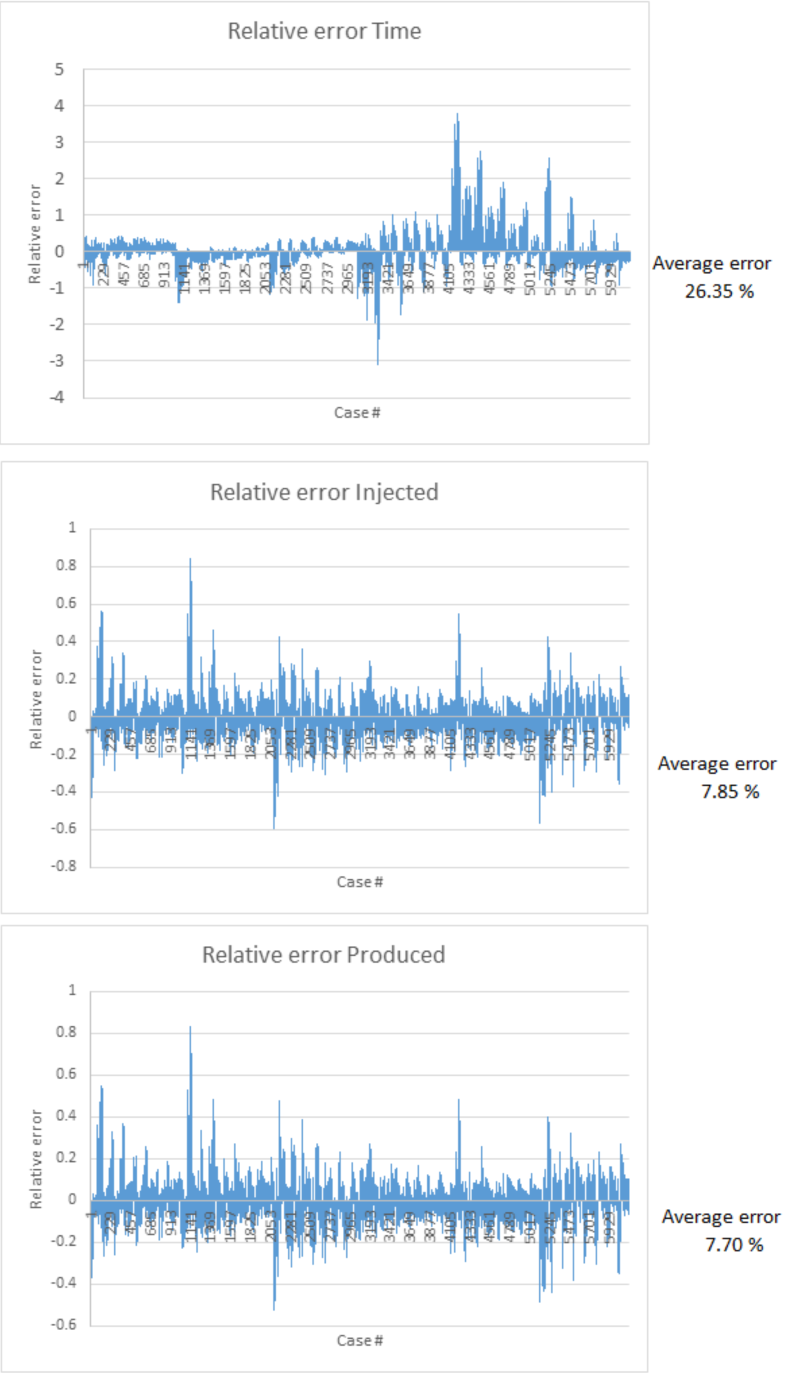


Figure 15-3 Relative error for time, injected CO_2 and produced brine with perforation length 3 and 7 grid blocks

Table 15-1 Regression coefficients for time, injected CO_2 and produced brine with a production well perforated in 1 grid block

Perforation length 1 grid block			
Regression coefficient	Time	Injected	Produced
P0	0	0	0
P1	-1.85875319	-16.1070592	-0.03697028
P2	31.4502164	171.304242	0.36966154
P3	-6.88774502	-36.2714872	-0.07466597
P4	-1.40456584	-16.2887917	-0.03933193
P5	-44.841376	277.672817	0.83339804
P6	-1441.4589	-17002.2414	-22.6645486
P7	-14.4301284	102.30573	0.2309172
P8	0	0	0
P1P1	0.23292327	1.51185832	0.00352458
P2P1	0	0	0
P3P1	0.25386392	2.38495367	0.00545152
P4P1	-0.07078647	0.03635543	3.5874E-05
P5P1	2.40033863	-39.8820348	-0.09072139
P6P1	-5.73465121	1137.98409	2.57589673
P7P1	-0.55414937	-4.86279032	-0.01094031
P8P1	0.01041867	0.0171457	3.6885E-05
P2P2	-4.35876397	-22.2050443	-0.04708691
P3P2	-0.61701427	-4.75552451	-0.01053584
P4P2	-0.12179068	-0.22852353	-0.00102138
P5P2	25.7741444	15.6076698	0.05079947
P6P2	-897.969049	-3276.37855	-6.37303635
P7P2	0.17952222	12.0508161	0.02760942
P8P2	0.03697127	0.07117278	0.00016331
P3P3	0.63776642	2.37998469	0.00475073
P4P3	0.04536747	-0.00752339	-0.00032784
P5P3	-1.94922244	91.2942285	0.17443949
P6P3	628.783645	2305.88607	2.28647762
P7P3	-0.68511359	0.51285497	0.00249496
P8P3	-0.00160411	-0.00476757	-6.3544E-06
P4P4	0.08966014	0.80779948	0.00222444
P5P4	-6.5276232	-27.2443993	-0.09262739
P6P4	128.24058	1327.4703	1.16000172
P7P4	0.18424264	0.13025515	-0.00020956
P8P4	-0.01246215	-0.04701823	-0.00011475
P5P5	48.0831246	290.409206	1.25931553
P6P5	71.464815	123023.952	235.299792
P7P5	-50.7001164	-1738.3489	-2.86463506
P8P5	2.26393932	10.6661841	0.02505215
P6P6	253194.54	-128018.063	-973.959978
P7P6	-148.298866	3870.65947	12.413417
P8P6	-32.8738409	-120.424581	-0.26747957
P7P7	5.45487243	-28.3532962	-0.06614355
P8P7	-0.00729289	0.03886406	8.2824E-05
P8P8	-0.00052626	0.00200444	4.6376E-06

Table 15-2 Regression coefficients for time, injected CO_2 and produced brine with a production well perforated in 2 grid blocks

Perforation length 2 grid blocks			
Regression coefficient	Time	Injected	Produced
P0	0	0	0
P1	10.9162724	137.202265	0.29268556
P2	0	0	0
P3	-2.14023313	-16.3331008	-0.02883958
P4	-0.85670789	-18.5930417	-0.04511433
P5	0.18377072	264.593188	0.78990956
P6	-37.6846406	1128.17949	18.9133856
P7	-2.67490157	34.4592862	0.07851825
P8	0	0	0
P1P1	-5.75439319	-77.9176497	-0.16771666
P2P1	11.7101659	159.228144	0.34350664
P3P1	0.09568242	3.20099899	0.00731587
P4P1	-0.02327701	0.08305607	0.00011094
P5P1	0.15647652	-4.78934172	-0.00927742
P6P1	21.7043016	1164.93889	2.54773056
P7P1	-0.08397136	-2.91717371	-0.00650043
P8P1	0.00121096	0.01258086	2.5192E-05
P2P2	-7.34471167	-98.0639064	-0.21041085
P3P2	-0.32356957	-4.86329204	-0.01077884
P4P2	-0.02511849	-0.19119338	-0.00098931
P5P2	-1.97576396	-116.846236	-0.25549976
P6P2	-161.839285	-292.483358	0.63774332
P7P2	0.06796028	4.65266539	0.01071143
P8P2	-0.00032439	-0.05273876	-0.00012563
P3P3	0.23758984	0.59238557	0.00065158
P4P3	0.05815969	0.2851787	0.00028686
P5P3	-8.25259101	-38.7543998	-0.12428858
P6P3	66.8333102	-2215.20166	-8.04813279
P7P3	-0.18074977	0.93403107	0.00278832
P8P3	-0.00286419	-0.03406126	-7.5967E-05
P4P4	0.02906438	0.63886172	0.00186305
P5P4	0.21612919	23.3858791	0.03072601
P6P4	61.1624981	1295.91787	0.88455537
P7P4	-0.01028713	0.04615514	-7.942E-05
P8P4	-0.00078586	-0.00958872	-2.1588E-05
P5P5	89.5420831	212.418495	0.84117881
P6P5	-7358.50814	-6887.96853	-59.7754812
P7P5	18.2196374	-336.120112	-0.19266096
P8P5	0.10213471	1.27058913	0.00314079
P6P6	13092.4877	-1741596.49	-4529.34306
P7P6	-110.462156	1384.22985	4.92036587
P8P6	-4.14298719	-36.5437344	-0.07381233
P7P7	0.52553164	-5.25968794	-0.01258691
P8P7	0.00037258	0.01519763	3.1166E-05
P8P8	0.00048794	0.00932445	2.0938E-05

Table 15-3 Regression coefficients for time, injected CO_2 and produced brine with a production well perforated in 3 and 7 grid blocks

Perforation length 3 grid blocks and up			
Regression coefficient	Time	Injected	Produced
P0	0	0	0
P1	5.20091242	122.318373	0.26965842
P2	0	0	0
P3	-0.5520349	2.95532699	0.01172552
P4	-0.66561591	-17.5304497	-0.04629403
P5	-3.55322299	-22.2475571	0.28054166
P6	561.929347	9639.70029	42.2604257
P7	-0.52973838	4.05994619	0.00965116
P8	0	0	0
P1P1	-2.72840982	-70.5792892	-0.15645719
P2P1	5.59946271	143.999657	0.31993516
P3P1	0.02555203	2.46555913	0.00570382
P4P1	-0.01537094	0.36593908	0.00072311
P5P1	0.22061214	44.1113297	0.10263041
P6P1	-22.0552771	-1178.84209	-2.9036411
P7P1	-0.00548387	-0.53655527	-0.00117123
P8P1	0.00038946	0.00963443	1.5995E-05
P2P2	-3.46133908	-87.4067906	-0.19339133
P3P2	-0.14279373	-3.32329766	-0.00729877
P4P2	-0.02414071	-0.18242077	-0.00101486
P5P2	-2.66193366	-147.672304	-0.32483601
P6P2	-21.1811939	1503.74455	4.71730418
P7P2	-0.0074985	0.65251603	0.00156495
P8P2	-0.00091842	-0.06851177	-0.00016555
P3P3	0.0314654	-0.95845983	-0.00248911
P4P3	0.04155246	-0.23154811	-0.00040802
P5P3	-0.98070404	-64.6475011	-0.18456916
P6P3	-49.5402273	-2290.84584	-5.60119934
P7P3	0.03186776	0.49300586	0.00107722
P8P3	-0.00166902	-0.04859971	-0.00010877
P4P4	0.01602875	0.77727663	0.00216848
P5P4	2.66348044	-4.14793815	-0.01368396
P6P4	41.1213378	278.637674	-0.15357841
P7P4	0.01151585	-0.00975411	-7.6623E-05
P8P4	-0.00022792	-0.0022813	-2.653E-06
P5P5	-65.5038071	1750.41285	3.52964098
P6P5	1477.64559	-24928.1268	-102.706578
P7P5	-6.25124752	-1.98484198	0.02745449
P8P5	0.00454354	0.04314148	0.00026725
P6P6	-119331.937	-442566.258	-2583.47874
P7P6	44.0996757	119.921453	0.174769
P8P6	-1.28286136	-22.4265548	-0.03410281
P7P7	0.01058808	-0.26373491	-0.00060364
P8P7	-2.5147E-05	0.00416234	7.4343E-06
P8P8	0.00033347	0.01199342	2.6596E-05

15.2 FULL SCALE TESTING

Results from full scale field testing is seen in Table 14-4. From the table it was clear that injection time was not representable in the proxy, while CO_2 injected and brine produced was in the same order of magnitude. Injection time calculated in the proxy was above 50 times longer than the simulated injection time. Injected CO_2 was about 23% more in the proxy scenario than it was from simulations, while brine produced from the proxy was roughly 40% of simulated produced brine.

Table 15-4 Comparison between a simulated injection scenario and a calculated using the proxy model

	Simulated	**Calculated**
Time [days]	9800	684530
Injected CO_2 [Sm3]	3.45*1e12	4.25*1e12
Produced brine [Sm3]	7.48*1e9	3.04*1e9

15.3 OPTIMIZATION

The results from optimization proved to be inconclusive. The Solver function found several different answers depending on the initial values of operational parameters. It is desirable to find the answer with the least possible wells. Because these are the biggest expensive in CO_2 injection, the answer with the least wells is most relevant. Operational parameters found with lowest number of wells can be seen in Table 15-5, and the well output is listed in Table 15-6.

Table 15-5 Operational parameters for the optimization exercise

Operational parameters		
Injection pressure	bar	205.15
Number of injection wells	# wells	1
Number of production wells	# wells	4
Perforation length producer	# blocks	6
Well distance	m	10190.65
GOR	Sm3/Sm3	1.20E+02

Table 15-6 Aquifer output from operational parameters

Entire reservoir		
Time	Days	**15186**
Injected	Sm3	**2.136924E+11**
Produced	Sm3	**2.018972E+09**

16 Discussion

Using Eclipse 100 to create a regression model proxy for CO_2 storage with a five spot pattern is possible. A quarter of the five spot is sufficient to simulate how the entire reservoir will behave because in a five spot it may be assumed that each well is a flow barrier and no fluid will move past the production wells. Up-scaling is done by multiplying injected CO_2 and produced brine with number of total injection and production wells in the aquifer. The time required to reach the total injected and produced was not dependent on the number of wells. Time was instead dependent on the distance between wells as the CO_2 was assumed to migrate equally between wells in each five spot.

Initially, it was expected that no pressure increase in the aquifer would be observed, since brine was assumed incompressible, and there was no diffusion between CO_2 and brine. This proved to be wrong as the aquifer pressure rose to injection pressure shortly after start of injection. Pressure increase proved to be quickest and most severe in low perforation lengths in the production well. For higher perforation lengths, in the producer, the pressure increase was slower, and for short breakthrough times aquifer pressure did not reach injection pressure. Increase in pressure may have caused problems for injection rate, as injection rate drops when pressure difference between injector and aquifer is reduced.

Differences are large for the same regression coefficients between perforation lengths, example P6 in Table 15-1, Table 15-2 and Table 15-3. A certain difference is to be expected, but the differences seen was larger than anticipated. This may cause discontinuities when changing between perforation lengths, and may contribute to inaccuracy when using the proxy model.

The regression model was not as accurate as desired, especially for calculating time to reach the injected amount of CO_2. A certain level of error is to be expected, but the calculated time was above the expected magnitude of error. For injection and production the error was not as severe, but there was also a periodically large error. Ideally the error is roughly equal for and independent of the specified simulation parameters. This was not the case for the result found in the model created in this thesis. Here the error indicated dependency on injection pressure and depth.

When the proxy was made it was first attempted to create one equation for all perforation lengths. This equation gave high error estimates, and large variations in the dimensionless parameters used as input to calculate the regression coefficients. The equations proved to be highly sensitive to production well perforation length at low perforations lengths. Low perforation lengths gave long injection times, and high dimensionless time. While for longer perforation lengths the injection time before breakthrough and GOR=50 was shorter, making dimensionless time smaller. This difference in dimensionless injection time and pore volume injected caused a large discontinuity between 1 and 3 perforated grid blocks. The discontinuity was smaller for the transition between 3 and 7 perforated grid blocks. To make up for this discontinuity a separate equation was made for one perforated grid block, two perforated blocks and another for three and seven perforated grid blocks. Making three separate equations made the proxy more accurate.

The dimensionless equations from chapter 9.1 is initially designed to operate with reservoir cubic meters (Rm^3). In the calculations in this report, standard cubic meters (Sm^3) was used as parameters for the dimensionless equations. This was partly because eclipse 100 does not allow for measuring different fluids when injecting or producing. It only measures fluid production/injection combined for each phase. Using only produced fluid total would have generated an error in the calculations for produced brine. The dimensionless equations should however be able to handle Sm^3 instead of Rm^3 since they are dependent on each other through a formation volume factor, thus it should not be a significant source for inaccuracy.

Making scripts to create and run simulations, extracting desired data from simulation files and solving for least squared method saves large amount of time. If the scripts are correctly written there is no error in the process.

16.1 FIELD TESTING

For the full scale field testing the results were relatively far off between simulations and calculations. Injection time was not comparable at all, while produced brine and injected CO_2 was in the same range. The error in injected CO_2 and produced brine was mainly believed to be caused by up-scaling of the grid. In the simulations used to create the proxy model the grid blocks were small compared to normal grid size, especially in the vertical direction. Small grid blocks have the advantage of having a more accurate capillary pressure, since it is a local phenomenon which is dependent on grid block size. Another benefit for smaller grid blocks is the fluid travels slower through smaller grid blocks because when a new fluid is introduced to a grid block in eclipse it is instantaneously distributed homogenously in the grid. This effect allows the fluid to come in contact with the next grid block almost immediately, compared to in real life where it is gradually introduced to new formation rock.

Fluid migration in a field case proved to be not entirely equal for all five spots. The five spots located at the edges had faster fluid breakthrough than the five spots that were surrounded by other five spots. Gas breakthrough happened sooner at the production wells located in the corners than in the middle.

16.2 OPTIMIZATION

Optimization of injection strategy gave an estimate of how much CO_2 the aquifer can store over a given amount of time. One of the challenges by using the Solver ad-in to optimize aquifer behavior is that there exist several different solutions for one injection scenario. Due to the existence of several possible solutions it may be necessary to try several times with the solver function using different initial values. It may also be necessary to try with different initial values, as not all initial values lead to a solution.

17 CONCLUSION

From the thesis following conclusions can be made:

- It is possible to create a regression model proxy by using a quarter spot model and scaling input parameters dimensionless.
- Error in time is large (18-36%), but there are large uncertainties linked to injection time in simulations, thus the result is acceptable.
- Error in injection and production are acceptable, between 6-8%.
- Errors from assumptions and simplifications should be negligible, making the inaccuracy come from the unexpected pressure increase in the aquifer, from eclipse, or from choice of model to create the proxy.
- The proxy model can be used to optimize injection scenarios.
- For field simulations with a long well distance the proxy has problems with injection time.

18 REFERENCES

Azad, A., & Chalaturnyk, R. (2013). Application of Analytical Proxy Models in Reservoir Estimation for SAGD Process: UTF-Project Case Study. doi: 10.2118/165576-PA

Bachu, S., & Bennion, D. B. (2008). Interfacial Tension between CO2, Freshwater, and Brine in the Range of Pressure from (2 to 27) MPa, Temperature from (20 to 125) °C, and Water Salinity from (0 to 334 000) mg·L−1. *Journal of Chemical & Engineering Data, 54*(3), 765-775. doi: 10.1021/je800529x

Bachu, S., Bonijoly, D., Bradshaw, J., Burruss, R., Holloway, S., Christensen, N. P., & Mathiassen, O. M. (2007). CO2 storage capacity estimation: Methodology and gaps. *International Journal of Greenhouse Gas Control, 1*(4), 430-443. doi: http://dx.doi.org/10.1016/S1750-5836(07)00086-2

Bachu, S., & Stewart, S. (2002). Geological Sequestration of Anthropogenic Carbon Dioxide in the Western Canada Sedimentary Basin: Suitability Analysis. doi: 10.2118/02-02-01

Baird, R. A. (1986). Maturation and source rock evaluation of Kimmeridge clay, Norwegian North Sea. *AAPG Bulletin, 70*(1), 1-11.

David Martin, F., & Colpitts, R. M. (1996). 5 - Reservoir Engineering. In W. C. Lyons (Ed.), *Standard Handbook of Petroleum and Natural Gas Engineering* (pp. 1-362). Houston: Gulf Professional Publishing.

Eiken, O., Ringrose, P., Hermanrud, C., Nazarian, B., Torp, T. A., & Høier, L. (2011). Lessons learned from 14 years of CCS operations: Sleipner, In Salah and Snøhvit. *Energy Procedia, 4*(0), 5541-5548. doi: http://dx.doi.org/10.1016/j.egypro.2011.02.541

Holt, T., Lindeberg, E. G. B., & Taber, J. J. (2000). *Technologies and Possibilities for Larger-Scale CO2 Separation and Underground Storage*.

IPCC. (2005). Special Report on Carbon Dioxide Capture and Storage. In B. Metz, Davidson, O., de Coninck, H.C., Loos, M., Meyer (Ed.), (pp. Chapter 5, pp. 195–276.): IPCC—Intergovernmental Panel on Climate Change.

Juanes, R., Spiteri, E. J., Orr, F. M., & Blunt, M. J. (2006). Impact of relative permeability hysteresis on geological CO2 storage. *Water Resources Research, 42*(12), W12418. doi: 10.1029/2005WR004806

Jurecka, F. (2007). *Robust design optimization based on metamodeling techniques*: Shaker Verlag.

Krevor, S. C. M., Pini, R., Zuo, L., & Benson, S. M. (2012). Relative permeability and trapping of CO2 and water in sandstone rocks at reservoir conditions. *Water Resources Research, 48*(2), W02532. doi: 10.1029/2011WR010859

Li, X., Boek, E., Maitland, G. C., & Trusler, J. M. (2012). Interfacial Tension of (Brines+ CO2):(0.864 NaCl+ 0.136 KCl) at Temperatures between (298 and 448) K, Pressures between (2 and 50) MPa, and Total Molalities of (1 to 5) mol· kg–1. *Journal of Chemical & Engineering Data, 57*(4), 1078-1088.

Mizenko, G. J. (1992). *North Cross (Devonian) Unit CO2 Flood: Status Report*.

Mo, S., & Akervoll, I. (2005). *Modeling Long-Term CO2 Storage in Aquifer With a Black-Oil Reservoir Simulator*.

Nghiem, L., Shrivastava, V., Kohse, B., Hassam, M., & Yang, C. (2010). Simulation and Optimization of Trapping Processes for CO2 Storage in Saline Aquifers. doi: 10.2118/139429-PA

Pini, R., & Benson, S. M. (2013). Simultaneous determination of capillary pressure and relative permeability curves from core-flooding experiments with various fluid pairs. *Water Resources Research, 49*(6), 3516-3530. doi: 10.1002/wrcr.20274

Pruess, K., Xu, T., Apps, J., & Garcia, J. (2003). Numerical Modeling of Aquifer Disposal of CO2. doi: 10.2118/83695-PA

Shariati Pour, S., Pickup, G. E., Mackay, E. J., & Heinemann, N. (2012). *Flow Simulation of CO2 Storage in Saline Aquifers Using Black Oil Simulator*.

Soroush, M., Wessel-Berg, D., Torsaeter, O., Taheri, A., & Kleppe, J. (2012). *Affecting Parameters in Density Driven Convection Mixing in CO2 Storage in Brine*.

Stalkup, F. I. (1983). *Miscible displacement* (Vol. 8). New York: SPE.

Whitson, C. H., Brulé, M. R., & Engineers, S. o. P. (2000). *Phase Behavior*: Henry L. Doherty Memorial Fund of AIME, Society of Petroleum Engineers.

Yevtushenko, O., Bettge, D., Bohraus, S., Bäßler, R., Pfennig, A., & Kranzmann, A. (2014). Corrosion behavior of steels for CO2 injection. *Process Safety and Environmental Protection, 92*(1), 108-118. doi: http://dx.doi.org/10.1016/j.psep.2013.07.002

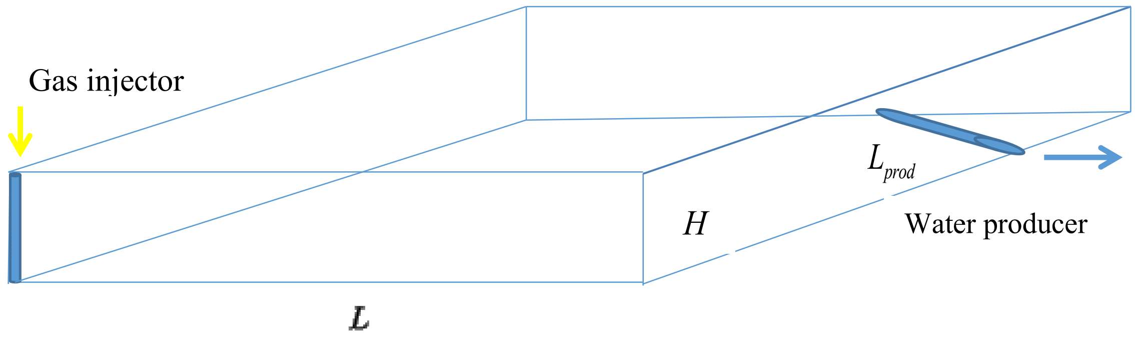
Zubarev, D. I. (2009). *Pros and cons of applying proxy-models as a substitute for full reservoir simulations.* Paper presented at the SPE Annual Technical Conference and Exhibition.

APPENDIX A: DERIVATION OF DIMENSIONLESS EQUATIONS

Free after note from Dr.Dag Wessel-Berg.

Scaling CO_2 injection with water production

Consider the following "quarter-spot model" for CO_2 injection through a vertical well in one corner and water production a horizontal well in the opposite corner:



Scaling gas injection

Assume the aquifer has horizontal extension L in each direction and height H, and that the injector has perforated interval of length H, and that the horizontal water producer has a perforated interval of length L_{prod}. The water producer is controlled by bottom hole pressure equal to the initial pressure. The injector well is also controlled by a given pressure not exceeding the fracturing pressure. We assume a immiscible incompressible model where the standard petrophysical parameters are constant. Let φ be porosity, S_j saturation, and $\vec{q}_j$ flux of phase $j = w, g$.

Mass conservation for each phase reads

$$\varphi \frac{\partial S_j}{\partial t} + \frac{\partial}{\partial x_1} q_{j1} + \frac{\partial}{\partial x_2} q_{j2} + \frac{\partial}{\partial x_3} q_{j3} = 0,\ j = w, g. \tag{1}$$

Do the following scaling, where dimensionless variable is on the right hand side: First define

$$q_0 = \frac{k_h}{\mu_w} \frac{\Delta p}{L},$$

and scale

$$x \mapsto \qquad \mapsto \qquad \mapsto \qquad \mapsto \qquad\qquad \mapsto \qquad ,$$

$$t \mapsto \frac{L\varphi(1 \quad S \quad)}{q_0} \qquad \mapsto \qquad \mapsto \qquad \mapsto \frac{U}{L} \qquad .$$

Here Δp is the initial pressure difference between injector and producer, k_h horizontal permeability, μ_w water viscosity, and S_{wrg} irreducible water saturation to gas. Then the dimensionless form of (1) is

$$\frac{\partial S_j}{\partial t} + \frac{\partial}{\partial x_1} q_{j1} + \frac{\partial}{\partial x_2} q_{j2} + \frac{\partial}{\partial x_3} q_{j3} = 0,\ j = w, g. \tag{2}$$

In physical units the phase fluxes are given by

$$q_{j1} = -\frac{k_{rj}(S_w)}{\mu_j} k_h \frac{\partial \psi_j}{\partial x_1},$$

$$q_{j2} = -\frac{k_{rj}(S_w)}{\mu_j} k_h \frac{\partial \psi_j}{\partial x_2},$$

$$q_{j3} = -\frac{k_{rj}(S_w)}{\mu_j} k_v \frac{\partial \psi_j}{\partial x_3},$$

where $k_{rj}\left(S_w\right)$ is relative permeability to phase j, and where ψ_j is the phase potential of phase j. We assume the relative permeabilities are of Corey type, i.e.

$$k_{rw}\left(S_w\right)=\left(\frac{S_w-S_{wrg}}{1-S_{wrg}}\right)^{\alpha_w}, k_{gr}\left(S_w\right)=k_{rg}^0\left(\frac{1-S_w}{1-S_{wrg}}\right)^{\alpha_g}.$$

Thus

$$\alpha_w, \alpha_g$$

(3)

are the two first dimensionless numbers for the problem.

Scaling the phase potential as $\psi_j \mapsto$, we obtain in dimensionless variables:

$$q_0 q_{wk}=-\frac{S^{\alpha_w}}{\mu_w}k_h\frac{\Delta p}{L}\frac{\partial\psi_w}{\partial x_k}\Leftrightarrow$$

$$q_{wk}=-S^{\alpha_w}\frac{\partial\psi_w}{\partial x_k}, \quad k=1,2, \qquad (4)$$

where $S=S_w\in\left[0,1\right]$. Note that $\frac{\partial\psi_w}{\partial x_k}\approx 1$, thus q_{wk} is of order 1, decreasing to zero when $S\to 0$

.

Now, the dimensionless equation for q_{w3} introduces the next dimensionless group, the aspect ratio R^2, which is the dimensionless vertical permeability. We have

$$q_0\frac{H}{L}q_{w3}=-\frac{S^{\alpha_w}}{\mu_w}k_v\frac{\Delta p}{H}\frac{\partial\psi_w}{\partial x_3}\Leftrightarrow$$

$$q_{w3}=S^{\alpha_w}R^2\frac{\partial\psi_w}{\partial x_3}, \qquad (5)$$

where

$$R^2 = \frac{L^2 k_v}{H^2 k_h} \tag{6}$$

is the aspect ratio. We observe that for long thin domains we generally have $R^2 \gg$ implying in general that vertical equilibrium is quickly achieved when buoyancy forces are present. The next dimensionless group entering is the mobility ratio:

For $k = 1,2$ we have

$$q_0 q_{gk} = -\frac{k_{rg}^0 \left(1-S\right)^{\alpha_g}}{\mu_g} k_h \frac{\Delta p}{L} \frac{\partial \psi_g}{\partial x_k} \Leftrightarrow$$

$$q_{gk} = -M\left(1-S\right)^{\alpha_g} \frac{\partial \psi_g}{\partial x_k}, \tag{7}$$

where

$$M = \left(\frac{\frac{k_{rg}^0}{\mu_g}}{\frac{1}{\mu_w}} \right) \tag{8}$$

is the (gas to water-) endpoint mobility ratio. The dimensionless equation for the vertical component is analogously

$$q_{g3} = -MR^2\left(1-S\right)^{\alpha_g} \frac{\partial \psi_g}{\partial x_3}. \tag{9}$$

In physical units we have

$$\psi_g - \psi_w = p_g - p_w - \rho_g g d + \rho_w g d = P_c + \Delta\rho g d,$$

where p_j are phase pressures, P_c capillary pressure, ρ_j phase mass densities, $\Delta\rho$ mass density difference, g acceleration of gravity, and d depth.

Scaling depth as $d \mapsto$, the dimensionless version of the potential difference is

$$\Delta p\left(\psi_g - \psi_w\right) = P_c^* P_{cD}\left(S\right) + \Delta\rho g H d \Leftrightarrow$$

$$\psi_g - \psi_w = N_{cv} P_{cD}\left(S\right) + N_{gv} d \quad , \tag{10}$$

where

$$N_{cv} = \frac{P_c^*}{\Delta p} \tag{11}$$

and

$$N_{gv} = \frac{\Delta \rho g H}{\Delta p} \tag{12}$$

are the capillary number and the gravity number respectively.

The shape of the dimensionless capillary pressure curve is often modelled using the pore size distribution index λ_{pore}, and λ_{pore} can also be considered a dimensionless number for the problem.

An additional dimensionless group for the initial value problem are

$$\xi_{prod} = \frac{L_{prod}}{H} , \tag{13}$$

the ratio between the length of the perforated intervals of the producer and the injector.

Finally, the well radii r_{inj} and r_{prod} are part of the initial value problem (they define the "inner boundary" of the solution domain), and give rise to two dimensional groups

$$\eta_{inj} = \frac{r_{inj}}{L} , \tag{14}$$

$$\eta_{prod} = \frac{r_{prod}}{L} , \tag{15}$$

Now, any set of physical input parameters defining the physical initial value problem can be scaled to a dimensionless initial value problem on the unit cube with wells defined by (13)-(15). The two unknown functions to be solved for are $S\left(x, y, z, t\right)$ and $\psi_w\left(x, y, z, t\right)$. The boundary conditions on the outer boundary are $\frac{d\psi_j}{dn} = 0$, $j = w, g$. At the injector we have $\psi_g = 1$ and $S = 0$, and at the producer $\psi_w = 0$. Initial conditions are $S = 1$ and $\psi_w = 0$.

Thus the dimensionless solution to the initial value problem depends on the groups

$$\vec{G}_{IVP} = \left[\alpha_w, \alpha_g, R^2, M, N_{cv}, N_{gv}, \lambda_{pore}, \xi_{prod}, \eta_{inj,} \eta_{prod}\right].$$

Numerical realization of the initial value problem introduces the additional groups

$$\partial x = \frac{\Delta x}{L}, \partial y = \frac{\Delta y}{L}, \partial z = \frac{\Delta z}{H}. \tag{16}$$

Also setting maximal time step length $\Delta t_{\max}$ is known to affect results, giving

$$\partial t_{\max} = \frac{\Delta t_{\max} q_0}{L\varphi\left(1 - S_{wrg}\right)} = \frac{\Delta t_{\max} \Delta p k_h}{L^2 \mu_w \varphi\left(1 - S_{wrg}\right)}. \tag{17}$$

Note that assuming the existence of an approximate optimal maximal dimensionless time step $\partial t_{\max}$ gives the physical maximal time step as

$$\Delta t_{\max} = \partial t_{\max} \frac{L^2 \mu_w \varphi\left(1 - S_{wrg}\right)}{\Delta p k_h} \tag{18}$$

Compressibility has been disregarded in the discussion, and for the presented incompressible model the produced reservoir volume must equal the injected reservoir volume. Running simulations with positive total compressibility will facilitate a certain aquifer storage capacity without producing the corresponding reservoir volume, and delay the onset of water production. It seems most relevant to use the sum of water and rock compressibility for total compressibility, since the pressure propagation to the producer does not sense much of the gas.

It is therefore relevant to include an additional dimensionless group for the rock-water compressibility, e.g.

$$\chi = \left(c_w + c_r\right) \Delta p.$$

Note that χ is more or less the ratio between the time scale $T_p = \dfrac{L^2\varphi\left(c_r + c_w\right)\mu_w}{k_h}$ for pressure propagation over the length L and the time scale used in the above scaling exercise.

Thus, results from numerical simulations of the dimensionless problem are defined by the vector

$$\vec{G}_{NUM} = \left[\alpha_w, \alpha_g, R^2, M, N_{cv}, N_{gv}, \lambda_{pore}, \xi_{prod}, \eta_{inj,} \eta_{prod}, \chi, \partial x, \partial y, \partial z, \partial t_{\max}\right].$$

Thus, dimensionless breakthrough time is

$$t_{break} = t_{break}\left(\vec{G}_{NUM}\right),$$

or more generally, the time until the producer has a given gas-water ratio $GWR = \zeta > 0$:

$$t\left(\zeta\right) = t\left(\vec{G}_{NUM}, \zeta\right).$$

The corresponding physical time is

$$t^*\left(\zeta\right) = \frac{L^2\mu_w\varphi\left(1 - S_{wrg}\right)}{\Delta p k_h} t\left(\vec{G}_{NUM}, \zeta\right). \tag{19}$$

In physical units, the amount of stored gas at this time is

$$Q_g\left(t^*\left(\zeta\right)\right) = \varphi\left(1 - S_{wrg}\right)L^2 H \int_0^1\int_0^1\int_0^1 \left(1 - S\left(x, y, z, t\left(\vec{G}_{NUM}, \zeta\right)\right)\right) dxdydz \Leftrightarrow$$

$$Q_g\left(t^*\left(\zeta\right)\right) = \varphi\left(1 - S_{wrg}\right)L^2 H Q_D\left(\vec{G}_{NUM}, \zeta\right). \tag{20}$$

The average injection rate (or actually storage rate) before time $t^*\left(\zeta\right)$ is consequently

$$\bar{J}_g^* = \frac{Q_g\left(t^*\left(\zeta\right)\right)}{t^*\left(\zeta\right)} = \frac{\Delta p k_h H}{\mu_w} \frac{Q_D\left(\vec{G}_{NUM}, \zeta\right)}{t\left(\vec{G}_{NUM}, \zeta\right)}. \tag{21}$$

From numerical simulations we can estimate $t\left(\vec{G}_{NUM},\zeta\right)$ and $Q_D\left(\vec{G}_{NUM},\zeta\right)$ from (19) and (20), giving

$$t\left(\vec{G}_{NUM},\zeta\right) \doteq \frac{\Delta p k_h t^*\left(\zeta\right)}{L^2\mu_w\varphi\left(1-S_{wrg}\right)},$$

and

$$Q_D\left(\vec{G}_{NUM},\zeta\right) \doteq \frac{Q_g\left(t^*\left(\zeta\right)\right)}{\varphi\left(1-S_{wrg}\right)L^2 H}.$$

Estimation of these two dimensionless functions for relevant values of the dimensionless groups enables one then to evaluate breakthrough times and the corresponding stored amount of gas for any input to the models. The number of degrees of freedom is relatively large. However, some of the components of $\vec{G}_{NUM}$ are reasonably constant for CO2 injection, and the values for t and Q_D could be relatively insensitive functions of other components, at least in the regime relevant for practical considerations of CO2 storage.

Discussion of dimensional groups

The set of dimensional groups

$$\vec{G}_{NUM} = \left[\alpha_w,\alpha_g,R^2,M,N_{cv},N_{gv},\lambda_{pore},\xi_{prod},\eta_{inj,}\eta_{prod},\chi,\partial x,\partial y,\partial z,\partial t_{\max}\right]$$

along with the maximal gas-water ratio in the producer, ζ , is 15-dimensional. To simplify maters, it is ok to assume $\partial x = \partial y$. Also the shapes of the relative permeabilities for a water-CO2 system are reasonably invariant since most (all?) rocks are water wet. Thus α_w and α_g can be assumed constant to given values (estimated from core flooding experiments). For the study in question it would probably also be ok to use a single shape for the capillary pressure curve, corresponding to a single value for the pore size distribution index λ_{pore}. If not for technical drilling reasons, one could probably also assume that the dimensionless well radii are equal, i.e. $\eta_{inj} = \eta_{prod}$. Disregarding the numerical dimensionless groups, and assuming the maximal GWR ζ is fixed, we are now left with a 7-dimesional space of freedom:

$$\vec{G}_{RED} = \left[R^2, M, N_{cv}, N_{gv}, \xi_{prod}, \eta_{inj}, \chi \right].$$

For well distances in the order of kilometres, say $L = 5km$, and with aquifer height $H = 100m$, the ratio $\frac{L^2}{H^2} = 2500$ means that the aspect number R^2 is very large for most realistic permeability ratios (except in the case of impermeable layers stopping vertical flow completely) . The strength of the dimensionless vertical flow of gas is essentially dictated by $R^2 MN_{gv}$, and $R^2 MN_{gv} \gg$ (recall the horizontal flux of water is of order unity) would mean that the assumption of vertical equilibrium is appropriate. Consequently results should be relatively insensitive to specific values of R^2.

The endpoint mobility ratio M depends essentially on aquifer temperature since water viscosity is a relatively strong function of this parameter. However, the value of M will not change very much for the different cases considered.

APPENDIX B: UNIX-SCRIPT TO GENERATE .DATA FILES FOR ECLIPSE 100

This Unix-script generates .DATA files for eclipse 100 from templates and puts the .DATA files in a folder named DATAGEN.

```
#This script generates .DATA files for eclipse 100

#!/bin/bash
# Put in dummy zerot'h element

# Parameters changed are:
# Z  =  Number of grid blocks length in z-direction
# Ai = pVT table include file for a certain depth, with different salinities (PVDO,
PVDG)
# Bi = BHP for certain depths
# C  = COREY, Relative permeability include file (SOF2, SGFN)
# D  = Kv, Permeability in Z-direction
# E  = DEPTH, fixed for each loop (1000, 1500, 2000, 2500m)
# F  = POREP, reservoir pressure, fixed for each loop,(110, 160, 210, 260 bar)
# W  = WELL PERFORATIONS, defines how many perforations in x and y direction the
production well will have
# X  = names the different perforations

# Z decides which .DATA file to be used for generating new files
Z=( " " "10" "30" "75" )

# W define number of perforation in horizontal direction for producer
W=( "'WELLP'" "--@3" "--@7" )

# X write number of perforation in file name
X=( "1" "3" "7" )

# A defines pVT table include file. Integer define depth used
A1=( " " "D1000S4" "D1000S8" "D1000S12" "D1000S20" )
A2=( " " "D1500S4" "D1500S8" "D1500S12" "D1500S20" )
A3=( " " "D2000S4" "D2000S8" "D2000S12" "D2000S20" )
A4=( " " "D2500S4" "D2500S8" "D2500S12" "D2500S20" )

# B is BHP values 1st integer is 60% lithostatic pressure. 2nd integer is middle value
between lithostatic and reservoir pressure. Integer define depth used
B1=( " " "156" "133" )
B2=( " " "231" "195.5" )
B3=( " " "306" "258" )
B4=( " " "381" "320.5" )

# C represents inputfile for relative permeability curves dependent on COREY exponents
Nw and Ng
C=( " " "NW2NG2K" "NW4NG2K" "NW4NG4K" "NW6NG4K" )

# D defines Vertical permeability, Kv and capillary pressure
D=( " " "5" "50" "250" "500" )

#echo "DEPTH 1000"
```

```
# E is current depth
E=1000

# F is current Reservoir pressure
F=110


#Runs through all CO2_GRIDZ.DATA files for different grid block numbers
for z in ${Z[@]}
do

# Run through all horizontal perforations
for ((w=0;w<${#W[@]};++w))
do

#Run through all pVT tables for specific depth
for a in ${A1[@]}
do

# Run through all BHP
for b in ${B1[@]}
do

# Run though  all COREY include files
for c in ${C[@]}
do

# Run through all vertical permeabilities
for d in ${D[@]}
do
echo "${X[w]} $z $a $b $c $d $E $F"

# Exhanges all keywords builds new .DATAfile in DATAGEN folder
sed -e "s/@DEPTH/${E}/g" -e "s/@POREP/${F}/g" -e "s/@COREY/${c}${d}/g" -e
"s/@pVT/${a}/g" -e "s/@BHP/${b}/g" -e "s/@Kv/${d}/g" -e "s/${W[w]}/'WELLP'/g"
CO2_GRIDZ${z}.DATA > ./DATAGEN/CO2_Z${z}A${a}B${b}C${c}${d}D${d}E${E}F${F}W${X[w]}.DATA
done
done
done
done
done
done

#Start the same process for new depth

#echo "DEPTH 1500"

# E is current depth
E=1500

# F is current Reservoir pressure
F=160


#Runs through all CO2_GRIDZ.DATA files for different grid block numbers
for z in ${Z[@]}
do

# Run through all horizontal perforations
for ((w=0;w<${#W[@]};++w))
do

#Run through all pVT tables for specific depth
```

```
for a in ${A2[@]}
do

# Run through all BHP
for b in ${B2[@]}
do

# Run though  all COREY include files
for c in ${C[@]}
do

# Run through all vertical permeabilities
for d in ${D[@]}
do
echo "${X[w]} $z $a $b $c $d $E $F"

# Exhanges all keywords builds new .DATAfile in DATAGEN folder
sed -e "s/@DEPTH/${E}/g" -e "s/@POREP/${F}/g" -e "s/@COREY/${c}${d}/g" -e
"s/@pVT/${a}/g" -e "s/@BHP/${b}/g" -e "s/@Kv/${d}/g" -e "s/${W[w]}/'WELLP'/g"
CO2_GRIDZ${z}.DATA > ./DATAGEN/CO2_Z${z}A${a}B${b}C${c}${d}D${d}E${E}F${F}W${X[w]}.DATA
done
done
done
done
done
done

#Start the same process for new depth

#echo "DEPTH 2000"

# E is current depth
E=2000

# F is current Reservoir pressure
F=210


#Runs through all CO2_GRIDZ.DATA files for different grid block numbers
for z in ${Z[@]}
do

# Run through all horizontal perforations
for ((w=0;w<${#W[@]};++w))
do

#Run through all pVT tables for specific depth
for a in ${A3[@]}
do

# Run through all BHP
for b in ${B3[@]}
do

# Run though  all COREY include files
for c in ${C[@]}
do

# Run through all vertical permeabilities
for d in ${D[@]}
do
echo "${X[w]} $z $a $b $c $d $E $F"
```

```
# Exhanges all keywords builds new .DATAfile in DATAGEN folder
sed -e "s/@DEPTH/${E}/g" -e "s/@POREP/${F}/g" -e "s/@COREY/${c}${d}/g" -e
"s/@pVT/${a}/g" -e "s/@BHP/${b}/g" -e "s/@Kv/${d}/g" -e "s/${W[w]}/'WELLP'/g"
CO2_GRIDZ${z}.DATA > ./DATAGEN/CO2_Z${z}A${a}B${b}C${c}${d}D${d}E${E}F${F}W${X[w]}.DATA
done
done
done
done
done
done

#Start the same process for new depth

#echo "DEPTH 2500"

# E is current depth
E=2500

# F is current Reservoir pressure
F=260


#Runs through all CO2_GRIDZ.DATA files for different grid block numbers
for z in ${Z[@]}
do

# Run through all horizontal perforations
for ((w=0;w<${#W[@]};++w))
do

#Run through all pVT tables for specific depth
for a in ${A4[@]}
do

# Run through all BHP
for b in ${B4[@]}
do

# Run though  all COREY include files
for c in ${C[@]}
do

# Run through all vertical permeabilities
for d in ${D[@]}
do
echo "${X[w]} $z $a $b $c $d $E $F"

# Exhanges all keywords builds new .DATAfile in DATAGEN folder
sed -e "s/@DEPTH/${E}/g" -e "s/@POREP/${F}/g" -e "s/@COREY/${c}${d}/g" -e
"s/@pVT/${a}/g" -e "s/@BHP/${b}/g" -e "s/@Kv/${d}/g" -e "s/${W[w]}/'WELLP'/g"
CO2_GRIDZ${z}.DATA > ./DATAGEN/CO2_Z${z}A${a}B${b}C${c}${d}D${d}E${E}F${F}W${X[w]}.DATA
done
done
done
done
done
done
```

APPENDIX C: UNIX SCRIPT TO RUN .DATA FILES IN ECLIPSE 100

This script runs all .DATA generated in the script from Appendix B files in eclipse 100. The script also moves the .RSM files into a folder named RSM and removes the rest of the output files from simulation.

To speed up the simulation time it is advised to split the script into several smaller scripts, to run several eclipse simulations in parallel.

```
#This file runs .DATA files in eclipse 100

#!/bin/bash
# Put in dummy zerot'h element

# Parameters used are:
# Z  =  Number of grid blocks length in z-direction
# Ai = pVT table include file for a certain depth, with different #salinities (PVDO,
PVDG)
# Bi = BHP for certain depths
# C  = COREY, Relative permeability include file (SOF2, SGFN)
# D  = Kv, Permeability in Z-direction
# E  = DEPTH, fixed for each loop (1000, 1500, 2000, 2500m)
# F  = POREP, reservoir pressure, fixed for each loop,(110, 160, 210, 260 bar)
# W  = WELL PERFORATIONS, defines how many perforations in x and y #direction the
production well will have
# X  = names the different perforations

# Z decides which .DATA file to be used for generating new files
Z=( " " "10" "30" "75" )

# W define number of perforation in horizontal direction for producer
W=( "'WELLP'" "--@3" "--@5" )

# X write number of perforation in file name
X=( "1" "3" "5" )

# A defines pVT table include file. Integer define depth used
A1=( " " "D1000S4" "D1000S8" "D1000S12" "D1000S20" )
A2=( " " "D1500S4" "D1500S8" "D1500S12" "D1500S20" )
A3=( " " "D2000S4" "D2000S8" "D2000S12" "D2000S20" )
A4=( " " "D2500S4" "D2500S8" "D2500S12" "D2500S20" )

# B is BHP values; 1st integer is 60% lithostatic pressure.
# 2nd integer is middle value between lithostatic and reservoir #pressure. Integer
define depth used
B1=( " " "156" "133" )
B2=( " " "231" "195.5" )
B3=( " " "306" "258" )
B4=( " " "381" "320.5" )

# C represents inputfile for relative permeability curves dependent on #COREY exponents
Nw and Ng
C=( " " "NW2NG2" "NW4NG2" "NW4NG4" "NW6NG4" )
```

```
# D defines Vertical permeability, Kv
D=( " " "5" "50" "250" "500" )

#echo "DEPTH 1000"

# E is current depth
E=1000

# F is current Reservoir pressure
F=110

#Runs through all CO2_GRIDZ.DATA files for different grid block numbers
for z in ${Z[@]}
do

# Run through all horizontal perforations
for ((w=0;w<${#W[@]};++w))
do

#Run through all pVT tables for specific depth
for a in ${A1[@]}
do

# Run through all BHP
for b in ${B1[@]}
do

# Run though  all COREY include files
for c in ${C[@]}
do

# Run through all vertical permeabilities
for d in ${D[@]}
do
echo "${X[w]} $z $a $b $c $d $E $F"

#Run all files in eclipse
@eclipse -ver 2013.2 -file CO2_Z${z}A${a}B${b}C${c}D${d}E${E}F${F}W${X[w]} -local
>/dev/null 2>&1
# Copies all .RSM files into folder named .RSM
mv CO2_Z${z}A${a}B${b}C${c}D${d}E${E}F${F}W${X[w]}.RSM ./RSM/.
# Deletes all files
rm -f
CO2_Z${z}A${a}B${b}C${c}D${d}E${E}F${F}W${X[w]}.{DBG,ECLEND,EGRID,INSPEC,MSG,PRT,RSM,RS
SPEC,SMSPEC,UNRST,UNSMRY,INIT}
done
done
done
done
done
done

# Start eclipse runs again at depth = 1500m

#echo "DEPTH 1500"

# E is current depth
E=1500

# F is current Reservoir pressure
F=160
```

```
#Runs through all CO2_GRIDZ.DATA files for different grid block numbers
for z in ${Z[@]}
do

# Run through all horizontal perforations
for ((w=0;w<${#W[@]};++w))
do

#Run through all pVT tables for specific depth
for a in ${A2[@]}
do

# Run through all BHP
for b in ${B2[@]}
do

# Run though  all COREY include files
for c in ${C[@]}
do

# Run through all vertical permeabilities
for d in ${D[@]}
do
echo "${X[w]} $z $a $b $c $d $E $F"

#Run all files in eclipse
@eclipse -ver 2013.2 -file CO2_Z${z}A${a}B${b}C${c}D${d}E${E}F${F}W${X[w]} -local
>/dev/null 2>&1
# Copies all .RSM files into folder named .RSM
mv CO2_Z${z}A${a}B${b}C${c}D${d}E${E}F${F}W${X[w]}.RSM ./RSM/.
# Deletes all files
rm -f
CO2_Z${z}A${a}B${b}C${c}D${d}E${E}F${F}W${X[w]}.{DBG,ECLEND,EGRID,INSPEC,MSG,PRT,RSM,RS
SPEC,SMSPEC,UNRST,UNSMRY,INIT}
done
done
done
done
done
done

# Start eclipse runs again at depth = 2000m

#echo "DEPTH 2000"

# E is current depth
E=2000

# F is current Reservoir pressure
F=210


#Runs through all CO2_GRIDZ.DATA files for different grid block numbers
for z in ${Z[@]}
do

# Run through all horizontal perforations
for ((w=0;w<${#W[@]};++w))
do

#Run through all pVT tables for specific depth
for a in ${A3[@]}
do
```

```
# Run through all BHP
for b in ${B3[@]}
do

# Run though  all COREY include files
for c in ${C[@]}
do

# Run through all vertical permeabilities
for d in ${D[@]}
do
echo "${X[w]} $z $a $b $c $d $E $F"

#Run all files in eclipse
@eclipse -ver 2013.2 -file CO2_Z${z}A${a}B${b}C${c}D${d}E${E}F${F}W${X[w]} -local
>/dev/null 2>&1
# Copies all .RSM files into folder named .RSM
mv CO2_Z${z}A${a}B${b}C${c}D${d}E${E}F${F}W${X[w]}.RSM ./RSM/.
# Deletes all files
rm -f
CO2_Z${z}A${a}B${b}C${c}D${d}E${E}F${F}W${X[w]}.{DBG,ECLEND,EGRID,INSPEC,MSG,PRT,RSM,RS
SPEC,SMSPEC,UNRST,UNSMRY,INIT}
done
done
done
done
done
done

# Start eclipse runs again at depth = 2500m

#echo "DEPTH 2500"

# E is current depth
E=2500

# F is current Reservoir pressure
F=260


#Runs through all CO2_GRIDZ.DATA files for different grid block numbers
for z in ${Z[@]}
do

# Run through all horizontal perforations
for ((w=0;w<${#W[@]};++w))
do

#Run through all pVT tables for specific depth
for a in ${A4[@]}
do

# Run through all BHP
for b in ${B4[@]}
do

# Run though  all COREY include files
for c in ${C[@]}
do

# Run through all vertical permeabilities
for d in ${D[@]}
```

```
do
echo "${X[w]} $z $a $b $c $d $E $F"

#Run all files in eclipse
@eclipse -ver 2013.2 -file CO2_Z${z}A${a}B${b}C${c}D${d}E${E}F${F}W${X[w]} -local
>/dev/null 2>&1
# Copies all .RSM files into folder named .RSM
mv CO2_Z${z}A${a}B${b}C${c}D${d}E${E}F${F}W${X[w]}.RSM ./RSM/.
# Deletes all files
rm -f
CO2_Z${z}A${a}B${b}C${c}D${d}E${E}F${F}W${X[w]}.{DBG,ECLEND,EGRID,INSPEC,MSG,PRT,RSM,RS
SPEC,SMSPEC,UNRST,UNSMRY,INIT}
done
done
done
done
done
done
```

APPENDIX D: C-SCRIPT TO SAVE DATA FROM .RSM FILES

This C-script is used for extracting relevant data from the .RSM files from simulations

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

// This script runs through a .RSM file and reads time,
// injected CO2 and produced brine at breakthrough (GOR=10 Sm3/Sm3)
// and last time step (GOR = 50)
// This script has to be compiled before used to read a .RSM file


int main(int argc,char* argv[])
{
//specify GOR = 10
  const double posprod = 10.0;
  FILE* fp;




// Specify characters and integers used in script
  char word1[20],word2[20],w3[20];
  char ch ;
  int not = 0, ax = 0 ;
  int i,ret,count,btcount;
  int order1,order2;
  double line[10],bttime,btinj,injtime,injtot,prod,a,b,btwat,wattot;
  int flag;

//Open file specified in script or in command window
  fp=fopen(argv[1],"r");


// Run through characters until tabulator count is 30, to skip unnecessary words
while ( 1 )
{
ch = fgetc ( fp ) ;
if ( not == 29 )
break ;

if ( ch == '\t' )
not++ ;
}


  // Find power used in columns  FGIPG and FGPT by counting asterisk
      fscanf(fp,"%s",word1);
    if(word1[0]=='*')
    {
    order1=atoi(&word1[5]);
    fscanf(fp,"%s",word2);
      if(word2[0]=='*')
      {
      order2=atoi(&word2[5]);
```

```
        for(i=1;i<3;i++)fscanf(fp,"%*s");
      }
      else
      {
        order2=0;
        fscanf(fp,"%*s");
      }

    }


  else
  {
    order1=0;
    order2=0;
    fscanf(fp,"%*s");
  }


// Run through all rows and find Breakthrough and GOR=50
  flag=0;
  count=0;
  do
  {
    count++;
    fscanf(fp,"%s",w3);

 // break loop if SUMMRAY is read
       if(strcmp(w3,"SUMMARY")==0)break;
    else line[0]=atof(w3);

       //Read all colums
    for(i=1;i<10;i++)ret=fscanf(fp,"%lf",&line[i]);

       //Save breakthrough values
    if(line[4]>posprod && flag==0)
    {
      bttime=line[0];
      btinj=line[8];
      btcount=count;

      if(order1!=0)
      // Determine power of FOPT by using asterisk
         // Multiply with powers found previously
      {
        if(order1==3)btinj=btinj*1e3;
              else btinj=btinj*1e6;
      }
      flag=1;
    }
  }

  while(count<10000 && ret!=EOF);
  if(count==10000 || ret==EOF)
  {
    printf("**ERROR** Wrong RSM format for file %s\n",argv[1]);
    exit;
  }
  injtime=line[0];
  injtot=line[8];
  prod=line[9];

// Multiplies with power, if power is present
```

```
  if(order1!=0)
  {
    if(order1==3)injtot=injtot*1e3;
    else injtot=injtot*1e6;
  }
  if(order2!=0)    fscanf(fp,"%f",&a);
  {
    if(order2==3)prod=prod*1e3;
    else prod=prod*1e6;
  }
   // If gas production rate do not reach 10 before last time step breakthrough and
total are equal
           if(bttime<posprod)
         {
           bttime=injtime;
           btinj=injtot;
         }


  //Reads 8 next words
  for(i=1;i<8;i++)fscanf(fp,"%*s");

// Search for power multiplier in FOPT
  fscanf(fp,"%s",w3);
  if(w3[0]=='*')order1=atoi(&w3[5]);
//If there are no power read next word
    else
   {
     fscanf(fp,"%*s");
     order1=0;
    }

// To not count the last line twice
  if(order1==0)btcount--;

// Save breakthrough water in a and total water in b
  for(i=1;i<=btcount;i++)
  {
    fscanf(fp,"%lf",&a);
    fscanf(fp,"%lf",&b);

  }
  btwat=b;
  if(order1!=0)
  {
    if(order1==3)btwat=btwat*1e3;
    else btwat=btwat*1e6;
  }
  do
  {
    ret=fscanf(fp,"%lf",&a);
  }while(ret!=EOF);

  wattot=a;

// Multiplying with power, if power is present
    if(order1!=0)
  {
    if(order1==3)wattot=wattot*1e3;
    else wattot=wattot*1e6;
  }
// If wattot is not counted in second column
    if(wattot<btwat)wattot=btwat;
```

```
//If breakthough happens on last line
  if(btwat<1)
       {
        btwat=wattot;
       }
// Print all variables separated by tab
  printf(" DEPTH\t PRES\t HEIGHT\t PERF\t SAL\t BHP\t COREY\t PERM\t
%lf\t%lf\t%lf\t%lf\t%lf\t%lf\n",bttime,btinj,btwat,injtime,injtot,wattot);

  return 0;
}
```

APPENDIX E: UNIX-SCRIPT TO SAVE RELEVANT DATA IN SINGLE FILE

Unix-script to use the compiled C-script and write data extracted from the C-script into a single file.

```
#!/bin/bash
# Put in dummy zerot'h element

#This program extracts desired numbers from .RSM files and prints them in a single text
file
#By reading file names and converting into output


# File name parameters
Z=( "10" "30" "75" )
# Equivalent output: Aquifer height m
HEIGHT=( "20" "60" "150" )


# File name paramters
W=( "1" "3" "7" )
# Equivalent output: # of perforations in X and Y direction
X=( "1" "3" "7" )

# A is file name parameter
# SAL is equivalent output: Salinity TDS %wt
SAL=( "4" "8" "12" "20")
A1=( "D1000S4" "D1000S8" "D1000S12" "D1000S20" )
A2=( "D1500S4" "D1500S8" "D1500S12" "D1500S20" )
A3=( "D2000S4" "D2000S8" "D2000S12" "D2000S20" )
A4=( "D2500S4" "D2500S8" "D2500S12" "D2500S20" )

# B is name convention and BHP
B1=( " " "156" "133" )
B2=( " " "231" "195.5" )
B3=( " " "306" "258" )
B4=( " " "381" "320.5" )

# File name parameters
C=( "NW2NG2K" "NW4NG2K" "NW4NG4K" "NW6NG4K" )
# Equivalent output: first integer is NW second integer is NG
COREY=( "2\t 2" "4\t 2" "4\t 4" "6\t 4" )

# Name convention and Vertical permeability
K=( " " "5" "50" "250" "500" )

# E is name convention and aquifer depth
E=1000

# F is name convention and reservoir pressure
F=110

# Loops to run through all file names
for ((z=0;z<${#Z[@]};++z))

 do
```

```
for ((w=0;w<${#W[@]};++w))

 do

for ((a=0;a<${#A1[@]};++a))
 do


 for b in ${B1[@]}
do

for ((c=0;c<${#C[@]};++c))

 do

for k in ${K[@]}
do
echo "${X[x]} ${Z[z]} ${A1[a]} $b ${C[c]} $k $E $F"

#Create separate text file with input data
./a.out ../../CO2_RSM/CO2_Z${Z[z]}A${A1[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.RSM
>
./TEST/RESULT/DATAFILECO2_Z${Z[z]}A${A1[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.txt

# Exhanges all keywords and builds new .DATAfile
sed  -e "s/DEPTH/${E}/g" -e "s/PRES/${F}/g" -e "s/COREY/${COREY[c]}/g" -e
"s/SAL/${SAL[a]}/g" -e "s/BHP/${b}/g" -e "s/PERM/${k}/g" -e "s/PERF/${X[w]}/g" -
e"s/HEIGHT/${HEIGHT[z]}/g"
./TEST/RESULT/DATAFILECO2_Z${Z[z]}A${A1[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.txt
>
./TEST/RESULT/FINAL/NEWFILE${Z[z]}A${A1[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.txt

#Merges all input data files into one
cat ./TEST/RESULT/FINAL/NEWFILE*.txt > ./DATA.txt

 done
 done
 done
 done
 done
 done

#Run through loops at new depth

# E is name convention and aquifer depth
E=1500

# F is name convention and reservoir pressure
F=160

# Loops to run through all file names
for ((z=0;z<${#Z[@]};++z))

 do

for ((w=0;w<${#W[@]};++w))

 do

for ((a=0;a<${#A2[@]};++a))
 do
 for b in ${B2[@]}
do

for ((c=0;c<${#C[@]};++c))

 do

for k in ${K[@]}
do
echo "${X[x]} ${Z[z]} ${A2[a]} $b ${C[c]} $k $E $F"

#Create separate text file with input data
./a.out ../../CO2_RSM/CO2_Z${Z[z]}A${A2[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.RSM
>
./TEST/RESULT/DATAFILECO2_Z${Z[z]}A${A2[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.txt

# Exhanges all keywords and builds new .DATAfile
sed  -e "s/DEPTH/${E}/g" -e "s/PRES/${F}/g" -e "s/COREY/${COREY[c]}/g" -e
"s/SAL/${SAL[a]}/g" -e "s/BHP/${b}/g" -e "s/PERM/${k}/g" -e "s/PERF/${X[w]}/g" -
e"s/HEIGHT/${HEIGHT[z]}/g"
./TEST/RESULT/DATAFILECO2_Z${Z[z]}A${A2[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.txt
>
./TEST/RESULT/FINAL/NEWFILE${Z[z]}A${A2[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.txt

#Merges all input data files into one
cat ./TEST/RESULT/FINAL/NEWFILE*.txt > ./DATA.txt

 done
 done
 done
 done
 done
 done

#Run through loops at new depth

# E is name convention and aquifer depth
E=2000

# F is name convention and reservoir pressure
F=210


# Loops to run through all file names
for ((z=0;z<${#Z[@]};++z))

 do


for ((w=0;w<${#W[@]};++w))

 do

for ((a=0;a<${#A3[@]};++a))
 do


 for b in ${B3[@]}
do

for ((c=0;c<${#C[@]};++c))
```

```
 do


for k in ${K[@]}
do
echo "${X[x]} ${Z[z]} ${A3[a]} $b ${C[c]} $k $E $F"

#Create separate text file with input data
./a.out ../../CO2_RSM/CO2_Z${Z[z]}A${A3[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.RSM
>
./TEST/RESULT/DATAFILECO2_Z${Z[z]}A${A3[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.txt

# Exhanges all keywords and builds new .DATAfile
sed  -e "s/DEPTH/${E}/g" -e "s/PRES/${F}/g" -e "s/COREY/${COREY[c]}/g" -e
"s/SAL/${SAL[a]}/g" -e "s/BHP/${b}/g" -e "s/PERM/${k}/g" -e "s/PERF/${X[w]}/g" -
e"s/HEIGHT/${HEIGHT[z]}/g"
./TEST/RESULT/DATAFILECO2_Z${Z[z]}A${A3[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.txt
>
./TEST/RESULT/FINAL/NEWFILE${Z[z]}A${A3[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.txt

#Merges all input data files into one
cat ./TEST/RESULT/FINAL/NEWFILE*.txt > ./DATA.txt

 done
 done
 done
 done
 done
 done
 #Run through loops at new depth

# E is name convention and aquifer depth
E=2500

# F is name convention and reservoir pressure
F=260


# Loops to run through all file names
for ((z=0;z<${#Z[@]};++z))

 do


for ((w=0;w<${#W[@]};++w))

 do

for ((a=0;a<${#A4[@]};++a))
 do


 for b in ${B4[@]}
do


for ((c=0;c<${#C[@]};++c))

 do


for k in ${K[@]}
do
```

```
echo "${X[x]} ${Z[z]} ${A4[a]} $b ${C[c]} $k $E $F"

#Create separate text file with input data
./a.out ../../CO2_RSM/CO2_Z${Z[z]}A${A4[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.RSM
>
./TEST/RESULT/DATAFILECO2_Z${Z[z]}A${A4[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.txt

# Exhanges all keywords and builds new .DATAfile
sed  -e "s/DEPTH/${E}/g" -e "s/PRES/${F}/g" -e "s/COREY/${COREY[c]}/g" -e
"s/SAL/${SAL[a]}/g" -e "s/BHP/${b}/g" -e "s/PERM/${k}/g" -e "s/PERF/${X[w]}/g" -
e"s/HEIGHT/${HEIGHT[z]}/g"
./TEST/RESULT/DATAFILECO2_Z${Z[z]}A${A4[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.txt
>
./TEST/RESULT/FINAL/NEWFILE${Z[z]}A${A4[a]}B${b}C${C[c]}${k}D${k}E${E}F${F}W${X[w]}.txt

#Merges all input data files into one
cat ./TEST/RESULT/FINAL/NEWFILE*.txt > ./DATA.txt

 done
 done
 done
 done
 done
 done
```

APPENDIX F: ECLIPSE CODE

Eclipse code for different aquifer heights

Aquifer height 20m

```
RUNSPEC
TITLE

-- This is a template to generate .DATA files for aquifer heigth 20 meters

DIMENS
30 30  25/
OIL
GAS
UNIFIN
UNIFOUT
METRIC
WELLDIMS
-- max. wells   max. conn.    max. gr.     max. wells/group
      2           150            1              2               /
TABDIMS
-- Table Of Dimensions
-- NTSFUN  NTPVT  NSSFUN  NPPVT NTFIP NRPVT
-- ------  -----  ------ -----  ----- -----
     1       1      100     100     1   100    /
-- NTSFUN: No. of saturation tables entered.
-- NTPVT : No. of PVT tables entered (in the PROPS section).
-- NSSFUN: Max. no. of saturation node in each saturation table, ie.,
--         Max. no. of data points in each table.
-- NPPVT : Max. no. of pressure nodes in any PVT table
-- NTFIP : Max. no. of FIP regions def using FIPNUM in REGIONS section
-- NRPVT : Max. no. of Rs nodes in any live oilpvt table

START
 1     DEC    2013  12:00:00/

NSTACK
 200 /

--===============================================================
GRID
--===============================================================

INIT

-- Top 3m of aquifer
BOX
1 30 1 30 1 12/

DX
 10800*50 /

DY
 10800*50 /

DZ
 10800*0.25  /
```

```
PORO
     10800*0.30/
PERMX
     10800*500  /
PERMY
     10800*500  /
PERMZ
     10800*@Kv    /

-- Next 3m of aquifer
BOX
1 30 1 30 13 18/

DX
 5400*50 /

DY
 5400*50 /

DZ
 5400*0.5  /


PORO
     5400*0.30/
PERMX
     5400*500  /
PERMY
     5400*500 /
PERMZ
     5400*@Kv    /

-- Bottom of aquifer
BOX
1 30 1 30 19 25/

DX
 6300*50 /

DY
 6300*50 /

DZ
 6300*2  /


PORO
     6300*0.30/
PERMX
     6300*500  /
PERMY
     6300*500  /
PERMZ
     6300*@Kv    /

BOX
  1 30  1 30   1  1 /
TOPS
  900*@DEPTH/
ENDBOX
```

```
--===============================================================
PROPS
--===============================================================

DENSITY
--Densities at surface conditions
--Oil           Brine           Gas
-- kg/m3
  1020.824421   1020.824421   1.87191 /

-- Surface temp: 4 °C
-- Temp gradient: 3 °C/100m
-- Surface pres: 10bar
-- Pres gradient: 1 bar/10m

--Include files pvT table
INCLUDE
'../BUILD/PVTTABLE/@pVT.inc'/

--Include files rel.perm curves and capillary pressure
INCLUDE
 '../BUILD/COREY/@COREY.inc'/

ROCK
-- Pref     Compressibility
  @POREP       1E-5  /

-- Saturation Dependent Data


-- ===============================================================================
REGIONS
-- ===============================================================================

--===============================================================
SOLUTION
--===============================================================

EQUIL
--Datum depth   Pinit  WOC     pcwoc    GOC   pcgoc   Rs   Rv  Accuracy    Init. comp.
--   m           bar    m       bar      m     bar
   @DEPTH     @POREP   2700      0.0     0.0    0.0    1    0     0/


RPTRST
BASIC=2 DENO /

RPTSOL
DENO /

SUMMARY ================================================================
-------- THIS SECTION SPECIFIES DATA TO BE WRITTEN TO THE SUMMARY FILES
-------- AND WHICH MAY LATER BE USED WITH THE ECLIPSE GRAPHICS PACKAGE
------------------------------------------------------------------------
WBHP
'WELLI'  'WELLP'
/
FGPR
FGIR
FOPR
FGIPL
FGIPG
```

```
FGPT
FGIPR
--FGIT
FOPT
SEPARATE
RPTONLY
RUNSUM
EXCEL

SCHEDULE ================================================================
---------THIS SECTION SPECIFIES THE OPERATIONS TO BE SIMULATED
-------------------------------------------------------------------------

RPTSCHED
'RESTART' 'FIP=2' 'CPU=2' /

MESSAGES
-- Print limits                                         Stop limits
-- Messages Comments Warning Problems Error Bug  Messages Comments Warnings Problems
Error Bug
    6000     6000    10000   100000     2    100   60000    60000    100000   1000000
2   100 /

WELSPECS
-- General Spesification Data For Wells
--    WELL    WELL   L O C A T I O N      BHP    PREF.   DRAINAGE   *      *  Cross flow
--    NAME    GROUP    I        J       DATUM   PHASE    RADIUS
--   -------  -----  -----      ----     -----   -----   -------
     'WELLI'  'G1'      1        1        1*    'GAS'      1*      1*    1*      NO/
     'WELLP'  'G1'     10       10        1*    'OIL'      1* 1* 'SHUT'/
/

--Connection for 1, 3 and 7 perforations in x and y direction
COMPDAT
-- Connection Between Wells and Blocks
--    WELL         L O C A T I O N                    Saturation  Transmis.  Well Bore
--    NAME       I     J    K(upper) K(lower) STATUS  Table No.   Factor   Diameter, m
Eff. Kh  Skin  D-fact   Direction
--   ------   -----  ---   -------- --------  ------ ----------  ---------  ---------
-------  ----  ------   ---------
     'WELLI'    1     1        1      25       'OPEN'     0          1*        0.178
1*        1*     1*        1*    /
     'WELLP'   30    30        25     25       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   29    30        25     25        'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   28    30        25     25        'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   27    30        25     25        'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   26    30        25     25        'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   25    30        25     25        'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   24    30        25     25        'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   30    29        25     25        'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   30    28        25     25        'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   30    27        25     25        'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
```

```
     --@7   30    26        25     25       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   30    25        25     25       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   30    24        25     25       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@3   29    30        25     25       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@3   28    30        25     25       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@3   30    29        25     25       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@3   30    28        25     25       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
/



WCONINJE
-- Control Data For Injection Well
--    WELL    INJ              CONTROL   FLOW-RATE-TARGET      BHP      THP     VFP
DISSOLVED GAS IN
--    NAME    TYPE    STATUS    MODE    SURFACE  RESERVOIR    TARGET   TARGET  TABLE#
INJECTION LIQUID
--                                       Sm3/day  Rm3/day      bars    bars
Sm3 gas/ Sm3 liq
--   ------  ------   ------   -------  -------  ---------    ------   ------  ------
-------------
     'WELLI'  'GAS'   'OPEN'   'BHP'     1*      1*          @BHP       1*      1*
1*  /
/

WCONPROD
-- Control Data For Prodution Well
--    WELL            CONTROL  Oil rate  Water rate  Gas rate  Liquid rate  Res.fluid
rate    BHP      THP       VFP
--    NAME    STATUS    MODE     TARGET     TARGET     TARGET      TARGET       TARGET
TARGET   TABLE#    TABLE#
--   ------   ------   -------   ------   ------     ------     ------          ------
------     ------    -----
      WELLP    OPEN      BHP      1*         1*         1*          1*             1*
@POREP       1*        1*   /
/

--Stop injection at GOR=50Sm3/Sm3
WECON
--  WELL     MIN  MIN   MAX    MAX  MAX   WORK   END
--  NAME     OIL  GAS  WATER   GOR  WGR   OVER   FLAG
    'WELLP'   1*   1*    1*     50    1*   WELL     YES /
/

TUNING
--TSINIT  TSMAXZ  TSMINZ  TSMCHP  TSFMAX  TSFMIN  TSFCNV  TSDIFF  THRUPT TMAXWC
   1       100       0.1    .15     3.0     0.3    0.1     1.25    1E20   1* /
--TRGTTE  TRGCNV  TRGMBE  TRGLCV  XXXTTE  XXXCNV  XXXMBE  XXXLCV  XXXWFL  TRGFIP
TRGSFT
   0.1     0.001  1.0E-7  1.0E-4    10     0.01   1.0E-6  0.001   0.001   0.025     1*
/
-- NEWTMX   NEWTMN  LITMAX  LITMIN  MXWSIT  MXWPIT DDPLIM  DDSLIM  TRGDPR  XXXDPR
    12        1      200      1       8       8    1.0E6   1.0E6   1.0E6   1.0E6 /
```

```
TSTEP
    1000*100/
TSTEP
    1000*100/
TSTEP
    1000*100/
END
```

Aquifer height 60m

```
RUNSPEC
TITLE
CCS concept water injection - thermal effects

-- This is a template to generate .DATA files for aquifer heigth 30 meters

DIMENS
30 30  45/
OIL
GAS
UNIFIN
UNIFOUT
METRIC
WELLDIMS
-- max. wells   max. conn.    max. gr.     max. wells/group
      2           150            1              2               /
TABDIMS
-- Table Of Dimensions
-- NTSFUN  NTPVT  NSSFUN  NPPVT NTFIP NRPVT
-- ------  -----  ------ -----  ----- -----
     1       1      100     100     1   100    /
-- NTSFUN: No. of saturation tables entered.
-- NTPVT : No. of PVT tables entered (in the PROPS section).
-- NSSFUN: Max. no. of saturation node in each saturation table, ie.,
--         Max. no. of data points in each table.
-- NPPVT : Max. no. of pressure nodes in any PVT table
-- NTFIP : Max. no. of FIP regions def using FIPNUM in REGIONS section
-- NRPVT : Max. no. of Rs nodes in any live oilpvt table
--PARALLEL
--4  1*/

--SATOPTS
--HYSTER/

START
 1     DEC    2013  12:00:00/

NSTACK
 200 /


--===============================================================
GRID
--===============================================================

INIT

-- Top 3m of aquifer
BOX
1 30 1 30 1 12/

DX
 10800*50 /

DY
 10800*50 /

DZ
```

```
 10800*0.25  /

PORO
     10800*0.30/
PERMX
     10800*500  /
PERMY
     10800*500  /
PERMZ
     10800*@Kv    /

-- Next 3m of aquifer
BOX
1 30 1 30 13 18/

DX
 5400*50 /

DY
 5400*50 /

DZ
 5400*0.5  /

PORO
     5400*0.30/
PERMX
     5400*500  /
PERMY
     5400*500 /
PERMZ
     5400*@Kv    /

-- Bottom of aquifer
BOX
1 30 1 30 19 45/

DX
 24300*50 /

DY
 24300*50 /

DZ
 24300*2  /

--North Sea Shallow properties
PORO
     24300*0.30/
PERMX
     24300*500  /
PERMY
     24300*500  /
PERMZ
     24300*@Kv    /

BOX
  1 30  1 30   1  1 /
TOPS
  900*@DEPTH/
ENDBOX
```

```
--===============================================================
PROPS
--===============================================================

DENSITY
--Densities at surface conditions
--Oil           Brine           Gas
-- kg/m3
  1020.824421   1020.824421   1.87191 /

-- Surface temp: 4 °C
-- Temp gradient: 3 °C/100m
-- Surface pres: 10bar
-- Pres gradient: 1 bar/10m

--Include files pvT table
INCLUDE
'../BUILD/PVTTABLE/@pVT.inc'/

--Include files rel.perm curves and capillary pressure
INCLUDE
 '../BUILD/COREY/@COREY.inc'/

ROCK
-- Pref     Compressibility
  @POREP       1E-5  /

-- Saturation Dependent Data

-- ==============================================================================
REGIONS
-- ==============================================================================

--===============================================================
SOLUTION
--===============================================================

EQUIL
--Datum depth   Pinit  WOC     pcwoc    GOC   pcgoc   Rs   Rv  Accuracy    Init. comp.
--   m           bar    m       bar      m     bar
   @DEPTH     @POREP   2700      0.0     0.0    0.0    1    0     0/

RPTRST
BASIC=2 DENO /

RPTSOL
DENO /

SUMMARY ================================================================
-------- THIS SECTION SPECIFIES DATA TO BE WRITTEN TO THE SUMMARY FILES
-------- AND WHICH MAY LATER BE USED WITH THE ECLIPSE GRAPHICS PACKAGE
------------------------------------------------------------------------
WBHP
'WELLI'  'WELLP'
/
FGPR
FGIR
FOPR
FGIPL
FGIPG
FGPT
--FGIT
FOPT
```

```
FGIPR
SEPARATE
RPTONLY
RUNSUM
EXCEL

SCHEDULE =================================================================
---------THIS SECTION SPECIFIES THE OPERATIONS TO BE SIMULATED
-------------------------------------------------------------------------

RPTSCHED
'RESTART' 'FIP=2' 'CPU=2' /

MESSAGES
-- Print limits                                         Stop limits
-- Messages Comments Warning Problems Error Bug  Messages Comments Warnings Problems
Error Bug
    6000      6000   10000   100000    2    100   60000    60000    100000   1000000
2   100 /

WELSPECS
-- General Spesification Data For Wells
--    WELL    WELL   L O C A T I O N     BHP    PREF.   DRAINAGE   *     *  Cross flow
--    NAME    GROUP    I        J       DATUM   PHASE    RADIUS
--   -------  -----  -----     ----     -----   -----   -------
     'WELLI'  'G1'      1        1        1*    'GAS'      1*      1*    1*      NO/
     'WELLP'  'G1'     30       30        1*    'OIL'      1* 1* 'SHUT'/
/

--Connection for 1, 3 and 7 perforations in x and y direction
COMPDAT
-- Connection Between Wells and Blocks
--    WELL         L O C A T I O N                   Saturation  Transmis.  Well Bore
--    NAME      I     J     K(upper) K(lower) STATUS  Table No.   Factor   Diameter, m
Eff. Kh  Skin  D-fact   Direction
--   ------   -----  ---   -------- --------  ------ ----------  ---------  ---------
-------  ----  ------   ---------
     'WELLI'    1     1        1      45       'OPEN'     0          1*        0.178
1*       1*     1*        1*    /
     'WELLP'   30    30        45     45       'OPEN'     0          1*        0.178
1*       1*     1*        'Z'    /
     --@7   29    30         45     45       'OPEN'     0          1*        0.178
1*       1*     1*        'Z'    /
     --@7   28    30         45     45       'OPEN'     0          1*        0.178
1*       1*     1*        'Z'    /
     --@7   27    30         45     45       'OPEN'     0          1*        0.178
1*       1*     1*        'Z'    /
     --@7   26    30         45     45       'OPEN'     0          1*        0.178
1*       1*     1*        'Z'    /
     --@7   25    30         45     45       'OPEN'     0          1*        0.178
1*       1*     1*        'Z'    /
     --@7   24    30         45     45       'OPEN'     0          1*        0.178
1*       1*     1*        'Z'    /
     --@7   30    29         45     45       'OPEN'     0          1*        0.178
1*       1*     1*        'Z'    /
     --@7   30    28         45     45       'OPEN'     0          1*        0.178
1*       1*     1*        'Z'    /
     --@7   30    27         45     45       'OPEN'     0          1*        0.178
1*       1*     1*        'Z'    /
     --@7   30    26         45     45       'OPEN'     0          1*        0.178
1*       1*     1*        'Z'    /
     --@7   30    25         45     45       'OPEN'     0          1*        0.178
1*       1*     1*        'Z'    /
```

```
     --@7   30    24        45     45       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@3   29    30        45     45       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@3   28    30        45     45       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@3   30    29        45     45       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@3   30    28        45     45       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
/


WCONINJE
-- Control Data For Injection Well
--    WELL    INJ              CONTROL   FLOW-RATE-TARGET      BHP      THP     VFP
DISSOLVED GAS IN
--    NAME    TYPE    STATUS    MODE    SURFACE  RESERVOIR    TARGET   TARGET  TABLE#
INJECTION LIQUID
--                                       Sm3/day  Rm3/day      bars    bars
Sm3 gas/ Sm3 liq
--   ------  ------   ------   -------  -------  ---------    ------   ------  ------
-------------
     'WELLI'  'GAS'   'OPEN'   'BHP'     1*      1*          @BHP       1*      1*
1*  /
/

WCONPROD
-- Control Data For Prodution Well
--    WELL             CONTROL  Oil rate  Water rate  Gas rate  Liquid rate  Res.fluid
rate    BHP      THP       VFP
--    NAME    STATUS    MODE     TARGET     TARGET     TARGET      TARGET       TARGET
TARGET   TABLE#    TABLE#
--   ------   ------   -------   ------   ------      ------     ------          ------
------     ------    -----
      WELLP    OPEN      BHP      1*         1*          1*          1*             1*
@POREP       1*        1*   /
/

--Stop injection at GOR=50Sm3/Sm3
WECON
--  WELL     MIN  MIN   MAX    MAX  MAX   WORK   END
--  NAME     OIL  GAS  WATER   GOR  WGR   OVER   FLAG
    'WELLP'   1*   1*    1*     50    1*   WELL     YES /
/

TUNING
--TSINIT  TSMAXZ  TSMINZ  TSMCHP  TSFMAX  TSFMIN  TSFCNV  TSDIFF  THRUPT TMAXWC
   1       100      0.1    .15     3.0     0.3    0.1     1.25    1E20   1* /
--TRGTTE  TRGCNV  TRGMBE  TRGLCV  XXXTTE  XXXCNV  XXXMBE  XXXLCV  XXXWFL  TRGFIP
TRGSFT
   0.1     0.001  1.0E-7  1.0E-4    10     0.01   1.0E-6  0.001   0.001   0.025     1*
/
-- NEWTMX   NEWTMN  LITMAX  LITMIN  MXWSIT  MXWPIT DDPLIM  DDSLIM  TRGDPR  XXXDPR
    12        1      200      1       8       8    1.0E6   1.0E6   1.0E6   1.0E6 /

TSTEP
    1000*100/
TSTEP
    1000*100/
TSTEP
    1000*100/
END
```

Aquifer height 150m

```
RUNSPEC
TITLE

-- This is a template to generate .DATA files for aquifer heigth 150 meters

DIMENS
30 30  90/
OIL
GAS
UNIFIN
UNIFOUT
METRIC
WELLDIMS
-- max. wells   max. conn.    max. gr.     max. wells/group
      2           150            1              2               /
TABDIMS
-- Table Of Dimensions
-- NTSFUN  NTPVT  NSSFUN  NPPVT NTFIP NRPVT
-- ------  -----  ------ -----  ----- -----
     1       1      100     100     1   100    /
-- NTSFUN: No. of saturation tables entered.
-- NTPVT : No. of PVT tables entered (in the PROPS section).
-- NSSFUN: Max. no. of saturation node in each saturation table, ie.,
--         Max. no. of data points in each table.
-- NPPVT : Max. no. of pressure nodes in any PVT table
-- NTFIP : Max. no. of FIP regions def using FIPNUM in REGIONS section
-- NRPVT : Max. no. of Rs nodes in any live oilpvt table

START
 1     DEC    2013  12:00:00/

NSTACK
 200 /

--===============================================================
GRID
--===============================================================

INIT

-- Top 3m of aquifer
BOX
1 30 1 30 1 12/

DX
 10800*50 /

DY
 10800*50 /

DZ
 10800*0.25  /

PORO
     10800*0.30/
PERMX
     10800*500  /
```

```
PERMY
     10800*500  /
PERMZ
     10800*@Kv    /

-- Next 3m of aquifer
BOX
1 30 1 30 13 18/

DX
 5400*50 /

DY
 5400*50 /

DZ
 5400*0.5  /

PORO
     5400*0.30/
PERMX
     5400*500  /
PERMY
     5400*500 /
PERMZ
     5400*@Kv    /

-- Bottom of aquifer
BOX
1 30 1 30 19 90/

DX
 64800*50 /

DY
 64800*50 /

DZ
 64800*2  /

PORO
     64800*0.30/
PERMX
     64800*500  /
PERMY
     64800*500  /
PERMZ
     64800*@Kv    /

BOX
  1 30  1 30   1  1 /
TOPS
  900*@DEPTH/
ENDBOX

--=================================================================
PROPS
--=================================================================

DENSITY
--Densities at surface conditions
--Oil           Brine           Gas
-- kg/m3
```

```
  1020.824421   1020.824421   1.87191 /

-- Surface temp: 4 °C
-- Temp gradient: 3 °C/100m
-- Surface pres: 10bar
-- Pres gradient: 1 bar/10m

--Include files pvT table
INCLUDE
'../BUILD/PVTTABLE/@pVT.inc'/

--Include files rel.perm curves and capillary pressure
INCLUDE
 '../BUILD/COREY/@COREY.inc'/

ROCK
-- Pref     Compressibility
  @POREP       1E-5  /

-- Saturation Dependent Data

-- ============================================================================
REGIONS
-- ============================================================================

--=================================================================
SOLUTION
--=================================================================

EQUIL
--Datum depth   Pinit  WOC     pcwoc    GOC   pcgoc   Rs   Rv  Accuracy    Init. comp.
--   m           bar    m       bar      m     bar
   @DEPTH     @POREP   2700      0.0     0.0    0.0    1    0     0/

RPTRST
BASIC=2 DENO /

RPTSOL
DENO /

SUMMARY ================================================================
-------- THIS SECTION SPECIFIES DATA TO BE WRITTEN TO THE SUMMARY FILES
-------- AND WHICH MAY LATER BE USED WITH THE ECLIPSE GRAPHICS PACKAGE
------------------------------------------------------------------------
WBHP
'WELLI'  'WELLP'
/
FGPR
FGIR
FOPR
FGIPL
FGIPG
FGPT
--FGIT
FOPT
FGIPR
SEPARATE
RPTONLY
RUNSUM
EXCEL

SCHEDULE ===============================================================
---------THIS SECTION SPECIFIES THE OPERATIONS TO BE SIMULATED
```

```
------------------------------------------------------------------------

RPTSCHED
'RESTART' 'FIP=2' 'CPU=2' /

MESSAGES
-- Print limits                                        Stop limits
-- Messages Comments Warning Problems Error Bug  Messages Comments Warnings Problems
Error Bug
    6000     6000    10000   100000    2    100   60000    60000    100000   1000000
2   100 /

WELSPECS
-- General Spesification Data For Wells
--    WELL    WELL   L O C A T I O N     BHP    PREF.   DRAINAGE   *     *  Cross flow
--    NAME    GROUP    I         J       DATUM    PHASE    RADIUS
--   -------  -----  -----     ----     -----   -----   -------
     'WELLI'  'G1'      1         1        1*    'GAS'      1*      1*    1*      NO/
     'WELLP'  'G1'     30        30        1*    'OIL'      1* 1* 'SHUT'/
/

--Connection for 1, 3 and 7 perforations in x and y direction
COMPDAT
-- Connection Between Wells and Blocks
--    WELL        L O C A T I O N                  Saturation  Transmis.  Well Bore
--    NAME      I     J     K(upper) K(lower) STATUS  Table No.   Factor   Diameter, m
Eff. Kh  Skin  D-fact   Direction
--   ------   -----  ---   -------- --------  ------ ----------  ---------  ---------
-------  ----  ------   ---------
     'WELLI'    1     1        1      90       'OPEN'     0          1*        0.178
1*        1*     1*        1*    /
     'WELLP'   30    30     90     90          'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   29    30        90     90       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   28    30        90     90       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   27    30        90     90       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   26    30        90     90       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   25    30        90     90       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   24    30        90     90       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   30    29        90     90       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   30    28        90     90       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   30    27        90     90       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   30    26        90     90       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   30    25        90     90       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@7   30    24        90     90       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@3   29    30        90     90       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@3   28    30        90     90       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
     --@3   30    29        90     90       'OPEN'     0          1*        0.178
1*        1*     1*        'Z'    /
```

```
     --@3   30    28         90     90       'OPEN'     0          1*         0.178
1*        1*     1*        'Z'    /
/

WCONINJE
-- Control Data For Injection Well
--    WELL    INJ              CONTROL   FLOW-RATE-TARGET      BHP      THP     VFP
DISSOLVED GAS IN
--    NAME    TYPE    STATUS    MODE    SURFACE  RESERVOIR    TARGET   TARGET  TABLE#
INJECTION LIQUID
--                                       Sm3/day  Rm3/day      bars    bars
Sm3 gas/ Sm3 liq
--   ------  ------   ------   -------  -------  ---------    ------   ------  ------
-------------
     'WELLI'  'GAS'   'OPEN'   'BHP'     1*      1*          @BHP       1*      1*
1*  /
/

WCONPROD
-- Control Data For Prodution Well
--    WELL             CONTROL  Oil rate  Water rate  Gas rate  Liquid rate  Res.fluid
rate    BHP      THP       VFP
--    NAME    STATUS    MODE     TARGET     TARGET     TARGET      TARGET       TARGET
TARGET   TABLE#    TABLE#
--   ------   ------   -------   ------   ------      ------     ------          ------
------     ------    -----
      WELLP    OPEN      BHP      1*         1*          1*          1*              1*
@POREP       1*        1*   /
/

--Stop injection at GOR=50Sm3/Sm3
WECON
--  WELL     MIN  MIN   MAX    MAX  MAX   WORK   END
--  NAME     OIL  GAS  WATER   GOR  WGR   OVER   FLAG
    'WELLP'   1*   1*    1*     50    1*   WELL     YES /
/

TUNING
--TSINIT  TSMAXZ  TSMINZ  TSMCHP  TSFMAX  TSFMIN  TSFCNV  TSDIFF  THRUPT TMAXWC
   1       100      0.1    .15     3.0     0.3    0.1     1.25    1E20   1* /
--TRGTTE  TRGCNV  TRGMBE  TRGLCV  XXXTTE  XXXCNV  XXXMBE  XXXLCV  XXXWFL  TRGFIP
TRGSFT
   0.1     0.001  1.0E-7  1.0E-4    10     0.01   1.0E-6  0.001   0.001   0.025     1*
/
-- NEWTMX   NEWTMN  LITMAX  LITMIN  MXWSIT  MXWPIT DDPLIM  DDSLIM  TRGDPR  XXXDPR
    12        1      200      1       8       8    1.0E6   1.0E6   1.0E6   1.0E6 /

TSTEP
    1000*100/
TSTEP
    1000*100/
TSTEP
    1000*100/
END
```

APPENDIX G: MATLAB CODE

Matlab code using least squares optimization to find regression coefficients for different perforation lengths

Perforation length 1

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%This script takes in the regressors and observed values  %
%from simulations and uses least squares optimization to  %
%find regression coefficients for 1 perforated grid block %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

format long;

%Import Input data, 45 columns
filename = 'INPUTPerf1.txt';
delimiterIn = '\t';
A = importdata(filename,delimiterIn);
A_trans = A';

%Load measured dimensionless time to B vector
TIME = 'TIMEPerf1.txt';
varIn = '\t';
B = importdata(TIME,varIn);

%Load measured dimensionless injected CO2 to C vector
INJECT = 'INJECTPerf1.txt';
delimiterIn = '\t';
C = importdata(INJECT,delimiterIn);

%Load measured dimensionless produced water to D vector
WATER = 'WATERPerf1.txt';
delimiterIn = '\t';
D = importdata(WATER,delimiterIn);

%Use least squares method to find coefficients
x=A\B;
y=A\C;
z=A\D;

%Transpose coefficients to print row-wise, instead of columnwise
x=x';
y=y';
z=z';
R=[x;y;z];

%Write result in Excel file
result='MATLABperf1.xlsx';
xlswrite(result,R,'PROXY');
```

Perforation length 2

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%This script takes in the regressors and observed values   %
%from simulations and uses least squares optimization to   %
%find regression coefficients for 2 perforated grid blocks %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

format long;

%Import Input data, 45 columns
filename = 'INPUTPerf2.txt';
delimiterIn = '\t';
A = importdata(filename,delimiterIn);
A_trans = A';

%Load measured dimensionless time to B vector
TIME = 'TIMEPerf2.txt';
varIn = '\t';
B = importdata(TIME,varIn);


%Load measured dimensionless injected CO2 to C vector
INJECT = 'INJECTPerf2.txt';
delimiterIn = '\t';
C = importdata(INJECT,delimiterIn);

%Load measured dimensionless produced water to D vector
WATER = 'WATERPerf2.txt';
delimiterIn = '\t';
D = importdata(WATER,delimiterIn);

%Use least squares method to find coefficients
x=A\B;
y=A\C;
z=A\D;

%Transpose coefficients to print row-wise, instead of columnwise
x=x';
y=y';
z=z';
R=[x;y;z];

%Write result in Excel file
result='MATLABperf2.xlsx';
xlswrite(result,R,'PROXY');
```

Perforation length 3 and 7

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%This script takes in the regressors and observed values   %
%from simulations and uses least squares optimization to   %
%find regression coefficients for 7 perforated grid blocks %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

format long;
```

```matlab
%Import Input data, 45 columns
filename = 'INPUTPerf3and7.txt';
delimiterIn = '\t';
A = importdata(filename,delimiterIn);
A_trans = A';

%Load measured dimensionless time to B vector
TIME = 'TIMEPerf3and7.txt';
varIn = '\t';
B = importdata(TIME,varIn);

%Load measured dimensionless injected CO2 to C vector
INJECT = 'INJECTPerf3and7.txt';
delimiterIn = '\t';
C = importdata(INJECT,delimiterIn);

%Load measured dimensionless produced water to D vector
WATER = 'WATERPerf3and7.txt';
delimiterIn = '\t';
D = importdata(WATER,delimiterIn);

%Use least squares method to find coefficients
x=A\B;
y=A\C;
z=A\D;

%Transpose coefficients to print row-wise, instead of columnwise
x=x';
y=y';
z=z';
R=[x;y;z];

%Write result in Excel file
result='MATLABperf3and7.xlsx';
xlswrite(result,R,'PROXY');
```