Relative permeability estimation

for two phase flow in porous media

by

Rebecca Jacobs

TU Delft

Relative permeability estimation

for two phase flow in porous media

by

Rebecca Jacobs

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Thursday November 17, 2016 at 04:00 PM.

Student number:	4169190	
Project duration:	January 1, 2016 – November 17, 2016	
Thesis committee:	Prof. dr. ir. A. W. Heemink,	TU Delft
	Dr. ir. J. E. Romate,	TU Delft, supervisor
	Dr. ir. F. H. van der Meulen,	TU Delft

This thesis is confidential and cannot be made public until November 17, 2016.

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

TU Delft

Preface

Preface...

Rebecca Jacobs
Delft, November 2016

When looking for a thesis project, I had decided that I would like a project that touches numerical mathematics and statistics. These two fields of mathematics meet often in the mathematical physics. This duality is what attracted me to this project and both fields were clearly visible throughout. I enjoyed working on this project and the rest of my master applied mathematics. I would like to thank Johan Romate and Arnold Heemink for their guidance during this project.

Contents

1 Introduction

Even in modern times, where green energy initiatives are quickly developing and the fossil fuels are deemed a commodity that is sure to become superfluous, still The Netherlands consumes 960,600 bbl/day (2014 est.) of refined petroleum products [10]. This means that, at least for now, improving the effectiveness of oil extraction from the earth can make a big difference.
Also, when current methods of extracting oil have extracted all the oil that can be extracted, still massive amounts of oil remain in the ground[25]. Enhancing oil recovery techniques can help make these oil reserves accessible.

1.1. Introduction oil extraction

In the ideal case, oil recovery would go along the following lines. First some oil is found somewhere in the ground. The oil is typically found in sandstone or limestone layers,trapped beneath an impermeable layer. This impermeable layer is referred to as cap rock. For this toy example the oil is trapped in a sandstone layer that is sandwiched between two rock layers. It is impossible or very hard for the oil to move through the cap rock layer. Now the person that wants to extract the oil from the sandstone layer, drills a hole in the ground, which is commonly referred to as the production well, until he reaches the sandstone layer that contains oil. When it is reached, the enormous pressure difference between the sandstone layer and the atmosphere, forces the oil to flow up to the surface. Now the person trying the extract the oil only needs to somehow collect the oil at the surface.
After a while, the pressure in the subsurface drops because some of the oil left the sandstone layer. The oil stops flowing upwards to the surface, but there is still a lot of oil in the ground. Now the person extracting the oil would like to access the oil that does not automatically flow to the surface. What he can do is that he drills another hole, the pumping well, somewhere a bit further away. He drills until he finds the same sandstone layer. Then he starts pumping water into the sandstone layer to increase the pressure difference again. This flow of oil and water that moves through the rock is called flow in porous media and similar processes are found in a wide variety of applications such as modelling polluted water. For oil production this is a very simplified case of flow in porous media. In reality the subsurface contains water oil and gas. These three substances do not mix, which is why modelling them is referred to as three phase flow through porous media.
This is the basic general idea of oil extraction, but in reality it is far more complicated. For instance when the oil still flows from the reservoir naturally the pressure difference is often so large that the oil flows much harder then can be processed or collected. To prevent this from happening it is possible to put pressure on the production well to reduce the pressure difference. This is only a minimal example of how complicated oil extraction may get. The rock layers may lay very deep or the oil reservoir lies in hard to reach places, such as under the ocean. It is easily imaginable that oil fields under the oceans give rise to a whole set of complications in the oil recovery process. To pump water into the right layers it is key to know what the layers look like deep under the ground. This again is a complicated engineering problem. All these engineering problems need to be solved.

1.2. Special core analysis

To be able to solve some of the problems involved with extracting oil from the ground, a special core analysis experiment is performed. To determine certain characteristics of the rock a hole is drilled and a column of rock is removed. from this meters long column a horizontal core is removed. On this core, experiments can be done to estimate characteristics of the rock. Of interest is for instance the porosity of the rock,which is the percentage of air in the rock. Another interesting characteristic is the permeability of the rock, which measures how easy it is for a fluid to propagate through the rock. Other interesting characteristics are the residual oil and the connate water. The residual oil is the oil that never leaves the rock. This is oil that is stuck in pores and cannot be removed. The connate water is similarly to the residual oil, the water that never leaves the rock. It is unlikely to have oil that contains no water.
There are different types of experiments that can be done on such a core. This project only focusses on steady state experiments. In these experiments an oil filled core is taken and then water is pumped through to simulate the process of oil extraction in the ground. The experiment ends when it reaches a steady state. For the experiment used in this thesis, the inflow is kept at a constant rate. This means that the pressure is regulated at one end of the core and is kept constant at the other end. The measured data are the pressure signal that is necessary to keep the inflow constant, the water saturation of the core after the experiment and the produced oil. So the pressure and the produced oil are time signals and the water saturation is a spatial profile. Alternative experiments are for instance a centrifuge experiment where fluid is forced through the core by centrifuging the core. The main difference between these experiments is that in the steady state experiment the maximal force is limited by the pump used to force the mixture through the core. For the centrifuge experiment it is limited by how quickly the centrifuge can spin the core around. Typically much higher forces can be achieved by a centrifuge experiment. Because of this difference, certain experiments are better to estimate specific properties of the rock. To get a full picture of the rock, typically both experiments are done on a core or on cores from the same reservoir.
Since flow in porous media is a highly non linear processes, it is even with sufficient data not easy to find certain parameters. Matching parameters to the data is often referred to as history matching. Doing so using an automatised algorithm is called automatic history matching. In this thesis some automatic history matching techniques shall be reviewed.

1.3. Researchs goals

On a large scale the permeability and the porosity together define how quickly a fluid flows through the rock. But when there are multiple phases present in the rock they interact on a small scale. This interaction also influences how a fluid or mixture can move through a rock. It can even make a difference which fluid was there first. The scale at which these interactions take place is too small to model for an entire oil reservoir. That is why the relative permeability is introduced. The relative permeability is a function of the saturation. The saturation in this thesis is the fraction of oil or water present in the rock. The oil saturation is the fraction of oil in the fluid mixture and the water saturation is the fraction in water in the mixture. Since for this project only Special Core Analysis (SCAL) experiments are used only two phase flows need to be assesed. And since there are only two phases, their saturations must add up to one. In this thesis the relative permeability is a function of the water saturation. The relative permeability must represent all the small scale effects and must be easily computable so it can be used on large scale models. Since it is synthetic very little is known about it. The facts that are known are[22]:

- The relative permeabilities are functions of the saturation and are non linear
- For two phase flow the curves are defined for every water saturation larger than the connate water value and smaller then the residual oil value
- For two phase flow the sum of the two relative permeability curves is always smaller than one
- For two phase flow the oil relative permeability curve is decreasing and the water relative permeability curve is increasing function of the water saturation

The first objective of this thesis is to estimate the relative permeability curves. This entails investigating suitable parametrisations, and effective methods for solving the parameter estimation problem. This is done to be able to answer the main questions that motivated this thesis: What does the uncertainty look like for these estimates of the relative permeability curve and what factors influence this uncertainty.

Also, a similar analysis will be done for the capillary pressure curve. The capillary pressure is the pressure difference between the two phases. The small scale effect of surface tensions allows for a difference in pressures between the two phases. Even though these capillary pressures are very small in comparison to the pressure in an oil field, neglecting to model them can give significant errors in oil production predictions. The parameters of the capillary pressure can again be estimated using history matching techniques and for this thesis there is an emphasis on how the uncertainty is modelled and what is of great influence of this uncertainty.
A last research question in this thesis is what estimating a larger set of parameters does for the uncertainty of the estimate. How do the uncertainties in the relative permeability curve and the relative permeability curves relate, and how estimating both at the same time influences the quality of the estimate.
To answer these questions, a small model is implemented that simulates a core flooding experiment. Then the parameters of the relative permeability curve and the capillary curve are estimated using data assimilation methods.

1.4. Outline of this thesis

In chapter 2 the model that simulates a core flooding experiment is described. Also standard parametrisations of the relative permeability curve and capillary pressure curve will be described. In chapter 3 general data assimilation techniques are described. Four different parameter estimation techniques are discussed explicitly. This gives an overview of some common techniques that could be used for such an application. In chapter 4 different parametrisations of the relative permeability curves are described. The parametrisations have varying degrees of freedom. Also, some enforce the known characteristics of the relative permeability curve more than others. The experiments that aim to define what influences uncertainties and the results of these experiments are described in chapter 5. The concept of a Twin experiment is also explained here. Chapter 6 contains the conclusion of this research project. chapter 6 also contains recommendations for further research.

2
Model

This model describes the flow of oil and water in a steady state SCAL experiment. During the steady state experiments an oil filled core is flushed with an oil-water mixture. At first the oil-water mixture is for instance 60% oil. This mixture is pumped into the core with a constant inflow velocity. When a steady state is achieved the saturation profile is measured. During the experiments the pressure drop and the produced oil is measured as a time signal. Then the experiment is repeated using a different oil percentage mixture.

2.1. General Model

Newtonian flow of incompressible fluids in porous media is generally modelled using two physical relations. The first is conservation of mass and the second is Darcy's equation. When modelling multiple flows in porous media each phase gives rise to a partial differential equation. For this project we have two phases, so the pde's for two phase flow in a porous medium in one dimension will be given.
The conservation law gives:

$$\frac{\partial \phi S_o}{\partial t} + \frac{\partial}{\partial x}\left(q_o\right) = 0, \tag{2.1}$$

$$\frac{\partial \phi S_w}{\partial t} + \frac{\partial}{\partial x}\left(q_w\right) = 0. \tag{2.2}$$

with S_o and S_w the oil and water saturation of the rock. ϕ is the porosity of the rock. These equations give a relation between the Darcy velocity and the saturation.
The second set of equations comes from Darcy's equation:

$$q_o = -\frac{KK_{ro}}{\mu_o}\left(\frac{\partial P_o}{\partial x}\right) = -\lambda_o\left(\frac{\partial P_o}{\partial x}\right), \tag{2.3}$$

$$q_w = -\frac{KK_{rw}}{\mu_w}\left(\frac{\partial P_w}{\partial x}\right) = -\lambda_w\left(\frac{\partial P_w}{\partial x}\right). \tag{2.4}$$

Here K is the permeability of the rock. This is a property of the rock. K_{rw} and K_{ro} are the relative permeabilities. These are functions of the saturations and capture the interactions of the fluids and the rock. The relative permeability will be further explained later. The μ_o and μ_w are the viscosities of the oil and the water. The λ_w and λ_o are called the transmissibilities of the phases.
Darcy's equation gives a relation between the pressure gradient and the Darcy velocity. P_o and P_w represent the pressure of the oil and water. q_o and q_w represent the Darcy velocities of the fluids. The Darcy velocity is simply the velocity of the fluid scaled to the porosity of the rock. So for a rock with non constant porosity but constant pressure gradient and other parameters, the Darcy velocity is constant. This model has the assumption that the flow is horizontal and gravity can be ignored.
Combining these equations gives a set of two equations that captures the flow of two phases in a porous

medium. The result is:

$$\frac{\partial S_o}{\partial t}\phi - \frac{\partial}{\partial x}\left(\frac{KK_{ro}}{\mu_o}\left(\frac{\partial P_o}{\partial x}\right)\right) = 0, \tag{2.5}$$

$$\frac{\partial S_w}{\partial t}\phi - \frac{\partial}{\partial x}\left(\frac{KK_{rw}}{\mu_w}\left(\frac{\partial P_w}{\partial x}\right)\right) = 0. \tag{2.6}$$

The saturation is the percentage of space that is filled with that phase. This gives rise to the conditions that $S_o + S_w = 1$. This is the first connection between the two equations. This gives rise to the following equation as part of the model:

$$S_w = 1 - S_o. \tag{2.7}$$

This means that there is only one saturation left to compute. The second connection between the two equations is the capillary pressure. In practice the pressure of the water and the oil display a small difference. This pressure difference is called the capillary pressure. This gives: $P_o - P_w = P_c$. To make this system solvable this function is set as a function of the saturations. More on this function will be explained later.

2.2. Numerics

Solving the system of equations described in section 2.1 can be done using many different methods. In this case finite volumes is chosen for the discretisation. In order to solve the system it is chosen to use IMPES. This scheme uses Implicit Pressure and Explicit Saturation. That means that the model is rewritten to two equations, one of which called the pressure equation and is non-linearly dependent on the saturation. The other equation is called the saturation equation and does linearly depend on the pressure. That is why one equation can be solved implicitly and the other is solved explicitly. IMPES is in contrast to it's fully implicit counterpart not unconditionally stable. It is however a computationally less expensive method. To get to the IMPES formulation the two original model equations (2.1) and (2.2) are subtracted and the capillary pressure relation is used. This results in:

$$-\frac{\partial}{\partial x}\left(\lambda_w\left(\frac{\partial P_o}{\partial x} + \frac{\partial P_c}{\partial x}\right)\right) - \frac{\partial}{\partial x}\left(\lambda_o \frac{\partial P_o}{\partial x}\right) = 0, \tag{2.8}$$

$$\lambda_o = \frac{KK_{ro}}{\mu_o}, \tag{2.9}$$

$$\lambda_w = \frac{KK_{rw}}{\mu_w}. \tag{2.10}$$

Since after these manipulations, there is only oil pressure and capillary pressure, the oil pressure will be denoted as the P for pressure without a subscript. This equation can be solved implicitly when the water saturation is known. This is because the relative permeability is a function that depends on the saturation alone. Now the discretised equation becomes:

$$\begin{aligned}
&2\lambda_{w,i+\frac{1}{2}}\left(\frac{P_{i+1}-P_i}{(\Delta x_{i+1}+\Delta x_i)\Delta x_i}\right) - 2\lambda_{w,i-\frac{1}{2}}\left(\frac{P_i - P_{i-1}}{(\Delta x_i + \Delta x_{i-1})\Delta x_i}\right) \\
&-2\lambda_{w,i+\frac{1}{2}}\left(\frac{P_{c,i+1}-P_{c,i}}{(\Delta x_{i+1}+\Delta x_i)\Delta x_i}\right) - 2\lambda_{w,i-\frac{1}{2}}\left(\frac{P_{c,i} - P_{c,i-1}}{(\Delta x_i + \Delta x_{i-1})\Delta x_i}\right) \\
&+2\lambda_{o,i+\frac{1}{2}}\left(\frac{P_{i+1}-P_i}{(\Delta x_{i+1}+\Delta x_i)\Delta x_i}\right) - 2\lambda_{o,i-\frac{1}{2}}\left(\frac{P_i - P_{i-1}}{(\Delta x_i + \Delta x_{i-1})\Delta x_i}\right) = 0
\end{aligned} \tag{2.11}$$

Note that the λ_o for oil and λ_w for water should be known on the cell interfaces. Since the relative permeability is a function of saturation, that is only known on the cell centres these are not known on the interfaces. To indeed get the so called transmissibility on the interfaces an upwind relation is used. This means that $\lambda_{o,i+\frac{1}{2}} = \lambda_{o,i}$. This is done because of the direction of the flow is from right to left. Which means that all information is moving in that direction, thus the upwind relation is logical.
After the pressure is solved, the saturation can be solved for a new time using equation (2.2). This equation gives rise to the following discretiziation:

$$\begin{aligned}
&2\lambda_{o,i+\frac{1}{2}}\left(\frac{P_{i+1}-P_i}{(\Delta x_{i+1}+\Delta x_i)\Delta x_i}\right) - 2\lambda_{o,i-\frac{1}{2}}\left(\frac{P_i - P_{i-1}}{(\Delta x_i + \Delta x_{i-1})\Delta x_i}\right) \\
&\qquad = \frac{\phi}{\Delta t}\left(S_{w,i}^{t+\Delta t} - S_{w,i}^{t}\right).
\end{aligned} \tag{2.12}$$

Since the pressure was just calculated this can be solved explicitly. Note that the transmissibility term contains a relative permeability, which is a function of the saturation. The saturation at the previous time step is used here.
The stability of the IMPES method is determined by the largest eigenvalue of the pressure equation. In a simplification of this model called Buckley Leverett the speed of the shock is the same as the largest eigenvalue. The result of a system with eigenvalues that are too large is oscillations in the solution. The CFL limit is a limit that depends on the ratio between δx and δt and is required for stability of the system. Since the value of the largest eigenvalue is determined by a complex interaction of different parameters[11] the exact CFL limit is not easy to compute for this problem. Determining what values should be chosen for δt is hard and to do this automatically more complicated numerical methods should be used. Because the numerical aspects of this model are not the focus for this project, simply a sufficiently small time step has been used. This has been tested by trying some different values for δt.
To work with this model in data assimilation some formal notation is necessary. In data assimilation it is common to speak of the state of the model, defined as:

$$\psi_t = \begin{bmatrix} P_t \\ S_t \\ O_t \end{bmatrix}. \tag{2.13}$$

Note that P_t and S_t are vectors, the size of the spatial grid. O_t is a one dimensional vector and represents the amount of produced oil up to that time.
Solving equation (2.11) and then (2.12) propagates the state of the model one time step. This dynamical model is denoted as

$$\psi_{t+\delta t} = f(\psi_t). \tag{2.14}$$

2.2.1. Boundary conditions

There are different ways to treat the boundary conditions for the system described here. For the pressure equation there is a necessity of a pressure boundary condition or inflow boundary condition on each side of the system. Also, at the inflow side of the core the saturation must be known. For each boundary a choice has to be made. The pressure can be prescribed outside the domain on each side directly, which gives a Dirichlet boundary condition.The other option is that the inflow speed is be prescribed. This results in a Neumann boundary condition on the pressure.
Prescribing the inflow speed in the model is the same as setting the total Darcy velocity to a constant. This results in

$$q = q_o + q_w = c, \tag{2.15}$$

$$\frac{-KK_{ro}}{\mu_o}\frac{\partial P_o}{\partial x} + \frac{-KK_{rw}}{\mu_w}\frac{\partial P_w}{\partial x} = c, \tag{2.16}$$

$$\frac{\partial P_o}{\partial x}\left(\frac{-KK_{ro}}{\mu_w} + \frac{-KK_{rw}}{\mu_w}\right) + \frac{\partial P_c}{\partial x}\frac{-KK_{rw}}{\mu_w} = c. \tag{2.17}$$

For prescribing a constant inflow at the left boundary the discretisation for the first cell becomes:

$$\frac{P_{o,1} - P_{o,0}}{\Delta x} = \left(-\lambda_{w,\frac{1}{2}}\frac{P_{c,1} - P_{c,0}}{\Delta x} + c\right)\frac{1}{\lambda_{o,\frac{1}{2}} + \lambda_{w,\frac{1}{2}}}. \tag{2.18}$$

Prescribing the pressure at the left side of the domain results in the simple relation:

$$P_o = P_r \tag{2.19}$$

For some chosen pressure P_r. The SCAL experiment that is modelled in this thesis prescribes the inflow at one side and keeps the pressure constant at the other side of the core.
The chosen upwind method of retrieving a transmissibility on the interfaces implies a boundary condition on the inflow boundary of the domain. For the SCAL experiment modelled here the inflow boundary will be on the left side of the core. The saturation on the left boundary of the domain must therefore be prescribed. Physically this implies that the saturation of the pumping well is prescribed.

the

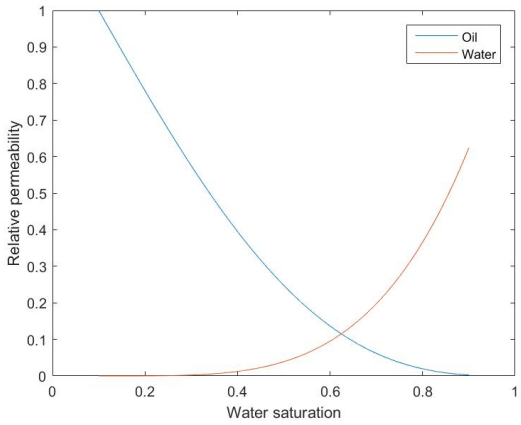


Figure 2.1: Relative permeability for connate water 0.1 residual oil 0.1 and so called Corey parameter $\lambda = 2$

2.3. Relative permeability and capillary pressure

As described before, if the relative permeability curves are a function of the saturation this model is solvable. A physical characteristic of the reservoir porous rock is the pressure of connate water. The connate water is the percentage of water that is always present in the rock. Before the extraction of oil begins, there is already some water present. On the other side, a rock will typically have residual oil after the extraction of oil. The residual oil is the oil that is trapped in the pores such that it can never leave the rock and denoted as S_{or} in this model. These two characteristics are defining factors for the relative permeability curves. A very common way of defining the relative permeability curves is the Brooks-Corey [6] model. This model is given as:

$$k_{ro}(S_o) = \left(\frac{S_o - S_{or}}{1 - S_{or}}\right)^2 \tag{2.20}$$

$$k_{rw}(S_o) = \left(\frac{1 - S_o}{1 - S_{or}}\right)^2 \left(1 - \left(\frac{S_o - S_{or}}{1 - S_{or}}\right)^4\right) \tag{2.21}$$

Here the capillary pressure is modeled as a function of the saturation only. This assumption enables us to solve the system with IMPES. The parametrisation used here is:

$$P_c(S_o) = \left(\frac{1 - S_{or}}{S_o - S_{or}}\right)^{\frac{1}{3}} \tag{2.22}$$

2.4. Buckley-Leverett

A very common simplification of the model is presented in the 1942 paper by Buckley and Leverett[7]. Here the equations (2.2) and (2.1) are rewritten such that they no longer depend on the pressure. The derivation starts with proposing general velocity $q = q_w + q_o$. Note that this velocity q is independent of place. Next the fractional flow function is defined as $f = \frac{q_w}{q}$. Now under the assumption that there is no capillary pressure, the system can be rewritten into

$$\frac{\partial S_w}{\partial t}\frac{\phi}{q} + \frac{\partial f}{\partial x} = 0, \tag{2.23}$$

$$f = \frac{k_{rw}}{k_{rw} + \frac{\mu_o}{\mu_w} k_{ro}}. \tag{2.24}$$

The fractional flow form is used in most literature because the capillary pressure is often very small in comparison to the oil and water pressure. The Buckley-Leverett form is also cheaper to compute since no pressures are needed. Another upside of the Buckley-Leverett form is that this equation has an analytical solution, which is useful to verify the numerical solution with.

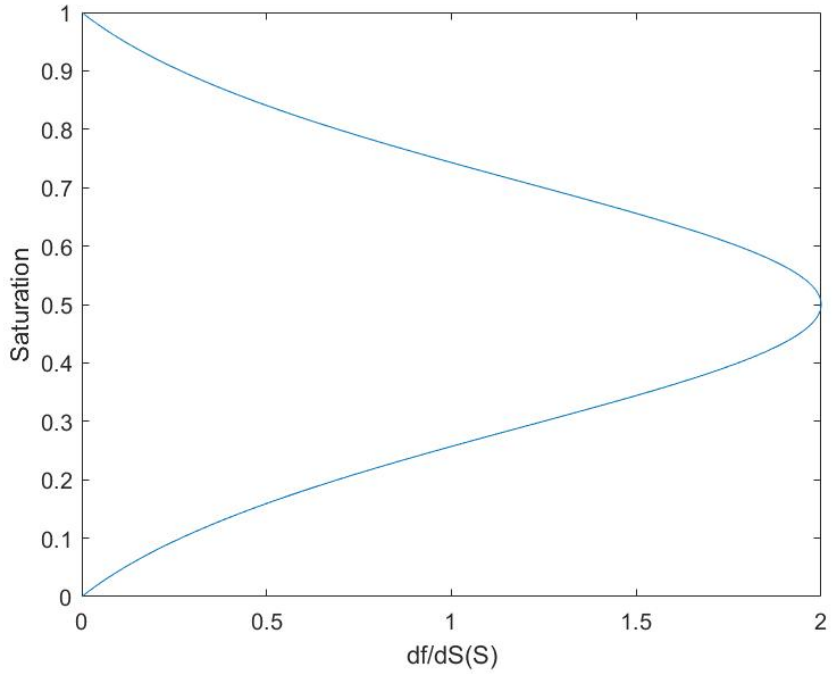


Figure 2.2: The saturation plotted against $\frac{\partial f}{\partial S}(S)$

2.4.1. Analytical solution Buckley-Leverett

Now a derivation of the analytical solution will be given. As explained before, S_w depends on x and t, which means that in this system it is true that:

$$dS_w = \frac{\partial S_w}{\partial t} dt + \frac{\partial S_w}{\partial x} dx \tag{2.25}$$

Since the aim is to find the characteristic of the saturation front the equation becomes

$$0 = \frac{\partial S_w}{\partial t} dt + \frac{\partial S_w}{\partial x} dx \tag{2.26}$$

this gives rise to the following equation

$$\frac{dx}{dt} = \frac{q}{\phi} \frac{df}{dS_w}. \tag{2.27}$$

This equation can be solved analytically by integrating over time which gives

$$x = \frac{qt}{\phi} \frac{df}{dS_w}, \tag{2.28}$$

for the water front. Because of this, the solution is called the frontal advance equation. There is no reference to the left and right pressure on the domain any more. This is replaced now by q which is the same on the whole domain at any time.
Plotting the water saturation against the place gives figure 2.2. Note that this does not give a good reflection of the physical situation. There are two different saturations for each place. This is due to the discontinuous nature of the solution.

In order to come up with a unique solution that complies with everything deduced before, the fractional flow function gives additional information. This will be done under the assumption that q is constant over time too. This is not a necessary demand for the formulation mentioned in the paper by Buckley-Leverett [7] but makes the analytical solution a bit simpler.
Up to the shock, the solution follows the derivative of the fractional flow function set out to x up to the shock.

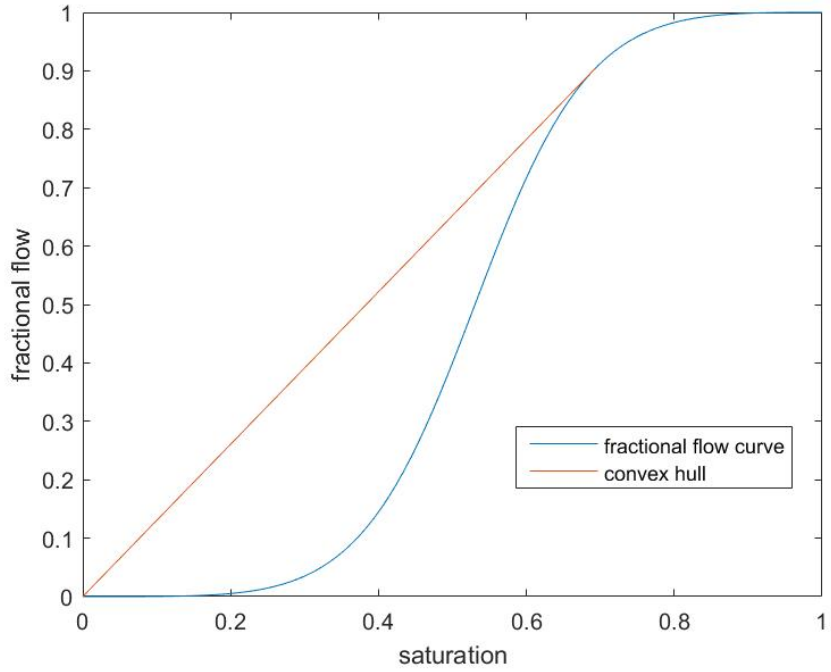


Figure 2.3: Fractional flow curve

Passed the shock, the solution equals zero. Now the only thing necessary is the speed of the shock. The speed of the shock can be derived using the fractional flow function and its convex hull as seen in figure (2.3). Where the convex hull of the fractional flow function hits the fractional flow function, is where the fractional flow function is equal to its derivative. The fractional flow function of the saturation for which this is true is the shock speed. So for shock speed $f(s^*)$ it is true that $f(s^*) = \frac{f(s^0)-f(s^*)}{s^0-s^*}$. The value of s^* must now be numerically determined. When the value of S^* is known, the solution is known on the entire domain, see figure (2.4). In this figure a simpler relative permeability curve is used. This curve is given as:

$$K_{rw}(S_w) = S_w^2 \tag{2.29}$$

$$K_{ro}(S_w) = (1 - S_w)^2 \tag{2.30}$$

Also, to simplify there is no connate water or residual oil in the system. Note many aspects of this solution depend on choices made in the parameters, relative permeability curves and the capillary pressure curve.

2.5. Comparison model and analytical solution

A method of assessing the quality of the model is to compare it to the analytical Buckley-Leverett model. In order to do this the boundary condition of the model must have a given flow rate as the inflow boundary condition. This is because the analytical solution allows q to be prescribed and the inflow boundary condition allows for this too. In order to asses the quality of the numeric solution the distance between the analytical saturation profile and the numerically found saturation profile at a certain time T.
The error is calculated as follows for N grid points:

$$E(N) = \Sigma_i \frac{|S'(x_i) - S(x_i)|}{N} \tag{2.31}$$

Where S' is the numerically calculated saturation, S is the exact saturation and E is the error.
Note that the exact saturations came from equation (2.28), which was not lineair in S. In order to compute values for S, $0 = \frac{qt}{\phi}\frac{df}{dS_w} - x$ was computed numerically for each grid on which the error is calculated. This numerical approximation may have a small influence on the errors depicted in figure 2.5.

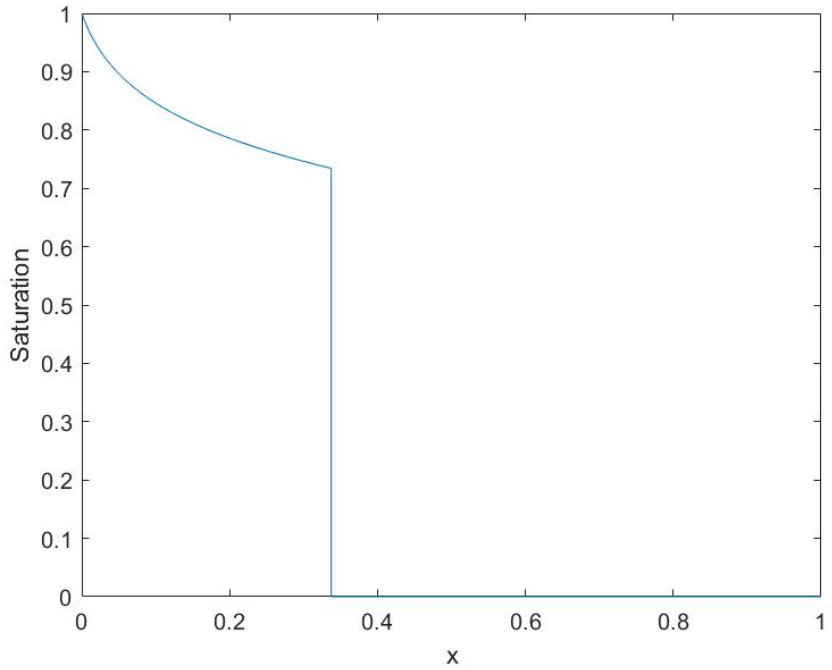


Figure 2.4: Typical Buckley Leverett saturation profile

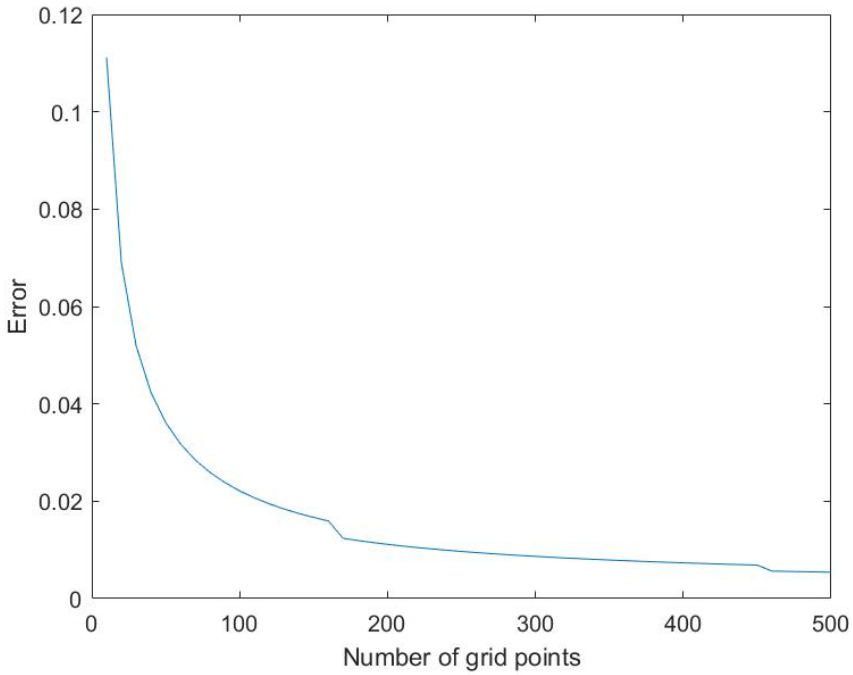


Figure 2.5: Model error

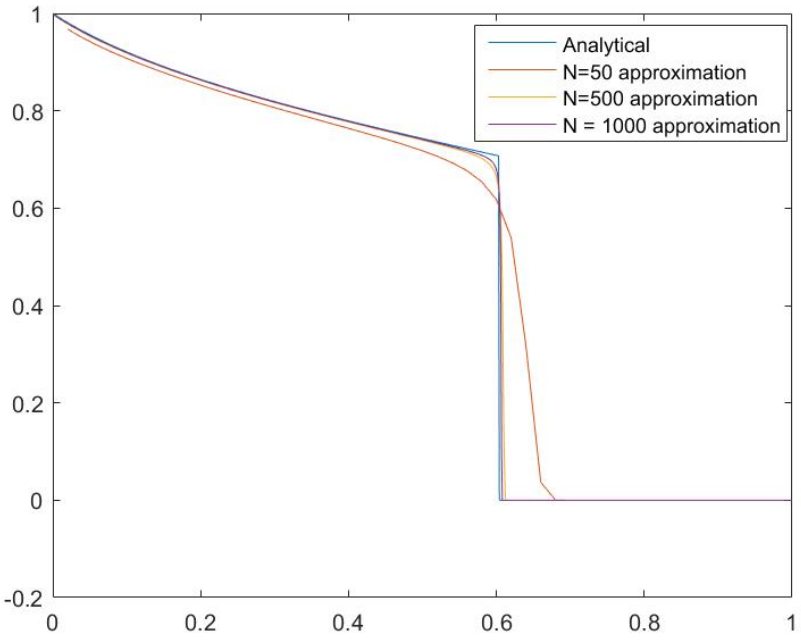


Figure 2.6: Saturation for different grids

3

Data Assimilation

Data assimilation is the study of combining actual measured data with model results. In every model there are errors. Everything from model errors to incorrect initial values can cause the model to stray from the reality it is trying to mimic. To improve these model results or to retrieve better parameter estimates, measurements of reality can be used. The general idea is that a model on its own can easily miss certain physical realities due to errors in parameters or simplifications of physical forces. Measured data without the underlying forces or processes on the other hand are poor means to predict the future. In general data assimilation in oil reservoir engineering is called automatic history matching. This is because the goal is to find the model and parameters that will match historical production data of reservoirs. When this is done with a computer, it is referred to as automatic.
There are various approaches to data assimilation and all methods have their strengths and weaknesses. Factors that often determine which method is suitable are the number of unknowns that are need to be estimated and whether or not the errors are near normally distributed. Some methods require far greater assumptions than others, which is of course something to take into account when selecting a particular method. Some methods allow for a continuous data assimilation cycle so data can be used to update the solution in real time. Others just allow all data to be assimilated at once. First some general notes on data assimilation will be made in section 3.0.1. The remaining part of the chapter is devoted to some different methods that are often used for data assimilation will be explained.

3.0.1. General notations in data assimilation

Data assimilation has some standard formulations and ideas that are good to know before looking at specific data assimilation methods. In general the goal of data assimilation is to improve the estimate of the state and or of a set of parameters. As mentioned in equation (2.13) and (2.14), the true state ψ_t, is a vector that contains all the model variables at time t. For the model used in this thesis that means it contains the pressure on all points of the domain, the saturations in all points of the domain and the amount of produced oil at time t. So as mentioned before,

$$\psi_t = \begin{bmatrix} P_t \\ S_t \\ O_t \end{bmatrix}, \tag{3.1}$$

$$\psi_{t+\delta t} = f_\theta(\psi_t) + \epsilon_\theta. \tag{3.2}$$

Where f propagates the model one time step under assumption of a set of parameters θ. When the parameters are considered uncertain, the dynamical model will make an error, which equals ϵ_θ. Propagating the model one time step means solving the pressure equation once and solving the saturation equation once. Note that indeed not all the entire state is measurable in a SCAL experiment. The saturation profile for instance is typically only measured at the end of the experiment.
Parameter estimation is hard for non-linear problems. For any set of parameters θ, or in the case of this model, a set of relative permeability curves, the corresponding states $\psi_t(\theta)$ can be computed for all times t. So this is the state for a certain relative permeability curve. Usually this curve is defined by a hand full of parameters, which are stored in vector θ. This state can then be compared to the data. However, since it is

impossible to reverse the process of going from parameter to state , finding a good fit for the parameters is an inverse modelling problem. This is why complicated data assimilation techniques are necessary, which require more accurate error modelling.
To improve the state estimate, data is used. The data is generally notated as d_t for time t. The dimensionality of the data is generally not the same as the state. This is because in real life situations, usually, not every point can be measured. The matrix H_t is then usually used to denote the mapping that maps the data grid to the state grid.
Then different types of errors can be distinguished for these types of problems. The first type is the measurement errors, denoted as a vector ϵ. So the aim is to retrieve ψ_t, the true state. But due to measurement errors it is only possible to observe

$$H_t\psi_t + \epsilon_t = d_t. \tag{3.3}$$

Note that if the available data is not constant over time, the matrix H_t depends on the time. In this project, pressure and produced oil are available at all times, but the saturation profile is only available at the end of the experiment. In many applications these measurement errors are assumed independent from each other. This a necessary assumption for some data assimilation methods.
The second type of error is the model error. This is the error that is made because the model is an imperfect representation of the SCAL experiment. Because the goal of this thesis is not verifying this model, but to investigate methods that quantify uncertainty, the model error is presumed zero. As mentioned before the vector of model parameters is denoted as θ. The current estimate of this parameter is, because it is an estimate, imperfect. Thus it has a certain error. In some data assimilation methods the errors are assumed Gaussian. When this is the case the errors are defined in a covariance matrix. The covariance matrix of the state is denoted as C_ψ. The covariance matrices of the data and the parameter are C_d and C_θ.
As mentioned before, when a give sate ψ_θ is given it is not possible to retrieve by inverting the model. What is possible in many cases is to determine that one state ψ_θ is a better fit than $\psi_{\theta'}$. Being able to see this difference allows search the state for the optimal set of parameters. The methods given in this section search for this set of parameters θ cleverly.

3.1. Deterministic calibration

One group of methods that is often used for parameter estimation is deterministic calibration. These methods in general have some kind of error criterion that gives a measure for the difference between the measured data and the model outcome. When having some kind of method to measure the data, an algorithm can iteratively reduce that distance until an optimal parameter value is found.

3.1.1. Object functions

The goal of the object function is that it measures the error of the proposed solution. When the current proposed model comes as close as possible to the true model the function should have a minimum. Not actually reaching zero has two causes. First the model error,when the chosen parametrisation is unable to take the exact form of the true relative permeability curve. This prohibits the state ψ_θ from being exactly the same as the true model state.
Secondly the data itself prohibits the object function from reaching zero. The data is assumed to have some measurement errors. So even when the true parameters are found exactly, the state of the model will still not be the same as the measured state. It is the difference between, observing $\psi + \epsilon$ and wanting to retrieve θ that corresponds with the true state ψ.
The data that is measured in this project is the amount of produced oil, the saturation profile at the end of the steady state experiment and the pressure drop. These data types all have a different order of magnitude and differently sized errors. This makes the construction of an object function more complicated. The general form of the object function for this project with three data types is:

$$J(\theta) = \sum_{i=1}^{N_T}\left(\frac{(P_{\theta,i} - P_i)^2}{\alpha_P}\right) + \sum_{i=1}^{N_x}\left(\frac{(S_{\theta,i} - S_i)^2}{\alpha_S}\right) + \sum_{i=1}^{N_T}\left(\frac{(O_{\theta,i} - O_i)^2}{\alpha_O}\right)$$

The P represents the measured pressure at the inflow boundary of the domain. The S represents the saturation measured at the end of the experiment and the O represents the produced oil. N_T is the total amount of time grid points and N_x is the amount of spacial grid points. The α terms are used to assign different weights to the different data types. These weights should correspond to the known uncertainties of the different data

types. If the measurements of a certain type are very precise, that data should be more trusted. The θ is the set of parameters. The variables with the sub script θ give the values of the simulation with that θ. The ones without a θ sub script are the measured values.
Further known information on the parameters can be added to the object function in the form of a regularisation term. The regularisation term can penalize behaviour which is known to be unexpected and enforce information that is known but not easily extractable from the measurements. Different options of regularisation for the oil related permeability are:

$$\sum_i \left(\frac{max(0, \theta_i - \theta_{i+1})}{\alpha_r} \right) \tag{3.4}$$

and

$$\sum_i \left(\frac{|\theta_i - \theta_{i+1}|}{\alpha_r} \right) \tag{3.5}$$

Here equation (3.4) penalizes a non decreasing relative permeability for the oil. Equation (3.5) penalizes great fluctuations in the relative permeability. Note that it is important to give the correct weights to the different data types. Putting too much weight on decreasing the fluctuations will drown out the measured data and will result in a constant relative permeability.
How the parametrisation of the relative permeability reacts to the regularisation terms can give information on the quality of the measured data. Because there are multiple types of measured data that have different added values, the balance between the regularisation and the measurements may not be constant in a single simulation. The saturation measurements, for instance, only give information on a very specific part of the domain of the relative permeability curve. So there there will be less information from measurements available outside that part of the domain. This may cause the regularisation term to overtake the solution on a part of the domain. Now that there is a sound criterion to measure the error of the of the model for a given parameter, an algorithm that will find the minimal error is introduced.

3.1.2. Newton algorithm

Newtonian optimisation is an effective steepest descent method for problems such as the one presented here. The steepest descent method generally looks like:

$$x_{n+1} = x_n - \nabla J(x_n). \tag{3.6}$$

for ∇J the derivative of the object function and x_n the current estimate of the parameter. So the goal of the optimisation is generally to find the minimum of the object function. The object function here is the distance between the true data of the twin experiment and the current data that is based of the current model parameters.
This method is very easy to implement but it may take a long time to converge. When the number or parameters increases, calculating the Jacobian becomes computationally more expensive. However, relative good convergence makes the method a relatively cheap parameter estimation method. A problem with this method is that it can converge to local optima of the object function is non-convex. So even when a solution is found, in many cases there is no way to know for sure that it is optimal. The process starts with an initial guess for the parameter. Furthermore it is hard to know what a reasonable guess is for the initial parameter.
An improvement to a simple steepest descent is Newton iteration. Where a general steepest descent method simply steps in the direction of the gradient for some stepsize, Newton iteration adapts the step size to the curvature of the object function. The Newton iteration is defined as [27]:

$$H(x_n)(x_{n+1} - x_n) = -\nabla J(x_n).$$

Where $H(x_n)$ is the Hessian matrix and contains all the second derivatives of the object function. Newton iteration converges at a quadratic rate and since steepest descent is only known to converge linearly Newton iterations make for a faster method. The down side of Newton iteration is that it is more sensitive to a bad starting point and one should pay extra attention to a good starting point.
One of the downsides of the Newton method is that it requires the Hessian matrix to be calculated. This is numerically expensive.
The Quasi-Newton method is an adaptation of the Newton method and calculates the Hessian indirectly using the object function and the gradient to retrieve information about the curvature of the object function.

`Matlab` has a standard implementation of a Quasi-Newton method [2]. The quasi-Newton method builds up the problem to:

$$\min_x \frac{1}{2} x^T H x + c^T x + b \tag{3.7}$$

which is then optimal when $\nabla J(x^*) = Hx^* + c = 0$. So the optimal solution must be $-H^{-1}c$.
Quasi-Newton methods update the Hessian sequentiality. There are many different methods for updating the Hessian but BFGS is a very common choice and works well as a general purpose method. BFGS, named after Broyden,Fletcher,Goldfarb and Shanno [23],[17], updates the Hessian in the following manner:

$$H_{k+1} = H_k + \frac{q_k q_k^T}{q_k^T s_k} - \frac{H_k s_k s_k^T H_k^T}{s_k^t H_k S_k}, \tag{3.8}$$

where

$$s_k = x_{k+1} - x_k, \tag{3.9}$$

$$q_k = \nabla J(x_{k+1}) - \nabla J(x_k). \tag{3.10}$$

After the determination of the search direction $d = -H^{-1}\nabla J(x_k)$, a line search is done.
The general idea of a line search is that given the direction of the search, an optimal step size is given by simply searching along the give direction. The line search in the `Matlab` Quasi-Newton algorithm works by dividing a suitable interval in sub intervals and then estimating the minimum of the object function using polynomial interpolation. The eventually chosen step size must fullfill the Wolfe conditions[26]. The Wolfe conditions are give as follows:

$$J(x_k + \alpha d_k) \leq J(x_k) + c_1 \alpha \nabla J(x_k) d_k \tag{3.11}$$

$$\nabla J(x_k + \alpha d_k) d_k \geq c_2 \nabla J(x_k) d_k \tag{3.12}$$

for some constant $c_1 \in (0,1)$and $c_2 \in (c_1, 1)$. The first Wolfe condition given in equation (3.11) demands that the stepsize gives sufficient decrease. The second Wolfe condition given in equation (3.12) rules out very small steps that do not make use of the high curvature. The Wolfe conditions together ensure that the chosen step size is a high quality estimate of the true optimal step size.
A Newton or quasi-Newton method requires a stopping criterion. In the `Matlab` optimisation toolbox several different criterion can be set. The maximal amount of iterations can be set for instance. Also a minimum value for the step size and for instance a minimum value for the gain of the objective function can be set.

Algorithm 1 Deterministic calibration

```
θ ← θ_0
while stop = 0 do
    Run model to evaluate state ψ_θ
    Evaluate J
    Evaluate ∇J and H to obtain d
    Execute line search to evaluate α
    θ = θ + αd
    if Stopping criteria are met then
        stop = 1
From Hessian H create confidence intervals
```

Confidence intervals
The deterministic calibration optimisation methods do not give immediate confidence bounds. The method does give an estimate of the Hessian or the actual Hessian. Under assumption of a Gaussian error distribution the Hessian of the objective function and the covariance matrix of the errors are related[28]. This relation will now be derived.
For a random normal vector θ with mean θ^* and covariance Σ_θ the probability density function is:

$$p(\theta) = (2\pi)^{-\frac{N_\theta}{2}} |\Sigma_\theta|^{-\frac{1}{2}} \exp(-\frac{1}{2}(\theta - \theta^*)' |\Sigma|_\theta^{-1} (\theta - \theta^*)). \tag{3.13}$$

Now because the parameter errors are assumed Gaussian the maximum of the probability density equals the maximum likelihood estimate. Because taking the natural logarithm of an object function keeps intact the desired properties,but allows for computational convenience it is common practice to do so. The object function can now be defined as:

$$J(\theta) \equiv -\ln(p(\theta)) = \frac{N_\theta}{2}\ln(2\pi) + \frac{1}{2}\ln|\Sigma_\theta| + \frac{1}{2}(\theta-\theta^*)'\Sigma_\theta^{-1}(\theta-\theta^*). \tag{3.14}$$

The (i,i')th entry of the Hessian matrix is defined as $\frac{\partial J(\theta)^2}{\partial\theta_i\partial\theta_i'}$. These relations then result in:

$$\frac{\partial J(\theta)^2}{\partial\theta_i\partial\theta_i'} = \Sigma^{-1}_{\theta,i,i'} \tag{3.15}$$

So the Hessian matrix is equivalent to the inverse of the covariance matrix under the assumption that the underlying random vector is normal.
In the experiments described here this result means that it is possible to retrieve a confidence interval as long as the optimisation gives a Hessian matrix such as the Newton method, but this confidence interval is only accurate under the assumption that the error is a Gaussian vector. Another issue is that the Hessian returned by the optimisation routine is not always the true Hessian. For Quasi-Newton it is a local numerical approximation of the real Hessian matrix.

3.2. Bayesian theory in data assimilation

When combining model states and real time measurements Bayesian theory turns out to be usefull. Many different methods for data assimilation such as the Ensemble Kalman Filter, the particle filter and the MCMC are based on this theory. In general Bayes formula is given as follows [8]:

Bayes rule. *Consider two random variables X and Y with conditional probability density function $p(x|y)$. It is true that:*

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}, \text{ for } p(y) \neq 0. \tag{3.16}$$

Generally the y is in data assimilation represented with d for the data and x is represented with ψ for the state. In many applications the state is the direct result of some chosen set of parameters called θ. This means that the conditional relation between the data and the parameters equals the conditional relation between the data and the state.
So Bayes rule gives information on the relation of two dependent variables. In many data assimilation methods the aim is to combine information measurements and model states to form one better solution. The general relation to this formula is that due to measurement errors, the measurements are actually realisations of a random variable. Better yet, since the measurements depend on the true state its corresponding density can be modelled with a conditional density $p(d|\theta)$ where d is the data of the measurements and θ represents the parameters. When the aim of the data assimilation is to estimate parameter values, the goal is to find $p(\theta|d)$. This is usually done by sampling cleverly from some easy to compute distribution, such that the eventual sample approximates a real sample from $p(\theta|d)$. In the context of Bayes formula three different methods that are commonly used in data assimilation will be assessed. The Kalman filter (KF) and the Particle filter (PF) are based on sequentially updating. The Markov Chain Monte Carlo (MCMC) method does this by only accepting certain realisations of a prior to a sample.

3.2.1. Ensemble Kalman Filter

One of the methods that uses the Bayes rule is the Kalman Filter (KF) and all its variations[15],[16]. A general motivation of the method is described in this section. The KF is a method that propagates the model for a certain time step,

$$\psi_t = \begin{bmatrix} P_t \\ S_t \\ O_t \end{bmatrix}, \tag{3.17}$$

$$\psi_{t+\delta t} = f_\theta(\psi_t) + \epsilon_\theta. \tag{3.18}$$

and is then able to integrate measurements when they become available. It uses estimated uncertainty of the model state and the measurements to determine the optimal linear estimate of the model state $\psi_t(\theta)$ under

condition of the measurements and the assumption that the model errors are Gaussian and the covariance matrix is known. It is a method that is very commonly used in large data assimilation problems. That is why a motivation of the KF will now be shown.
Consider θ a set of parameters of a dynamical system

$$d_t = H_t \psi_t + \epsilon_d \tag{3.19}$$

. Here H_t is some linear transformation that converts the full state to the measurable part of the model. So the measurements are certain parameters of the model plus a Gaussian random variable ϵ_d with mean zero and covariance matrix C_d. A standard object function under assumption of Gaussian errors can be defined as:

$$J(\theta,\psi) = \int_D (\theta-\theta^f)^T C_\theta^{-1}(\theta-\theta^f)\mathrm{d}\Theta + (d-HG(\theta))^T C_d^{-1}(d-HG(\theta)). \tag{3.20}$$

Note that this object function is build to minimise the distance between the true data and the model generated data, the distance between the initial guess parameter and the current model parameter and the model error. The Bayesian formulations allow for finding the variance minimising solution. After all, in real life the answer matters only as much as the available information on how good the answer is.
The strong relation between a classical object function formulation and the Bayesian formulation will now be shown. Assume Gaussian parameter errors with known covariance matrix C_θ. Defining a pdf $f(\theta)$ for the parameters, a pdf $f(\psi|\theta)$ for the model and assuming the $f(d|\theta,\psi) = f(d|\psi)$, which means that the measurements are to be assumed independent of the parameters. This gives:

$$p(d|\psi) \propto e^{-\frac{1}{2}(d-HG(\theta))^T C_d^{-1}(d-HG(\theta))}, \tag{3.21}$$

$$p(\theta) \propto e^{-\frac{1}{2}\int_D (\theta-\theta^f)^T C_\theta^{-1}(\theta-\theta^f)\mathrm{d}x} \tag{3.22}$$

Now Bayes rule gives:

$$f(\theta|d) \propto f(d|\theta) f(\theta). \tag{3.23}$$

All the assumptions on Gaussian errors with mean zero give in combination with relation (3.23):

$$f(\theta,\psi|d) \propto e^{J(\theta,\psi)}. \tag{3.24}$$

With this derivation it is shown that under all those assumptions on Gaussianity the traditional optimisations and the Bayesian perspective are heavily related. This interesting fact on its own however does not yet give better methods of finding $f(\theta,\psi)$ or $f(\theta)$). Especially not because of the assumed model errors. In the classical optimisation perspective the model errors are left out of consideration. In this new situation the method will try to reduce model errors as well as parameter errors. This makes the optimisation much harder. Assuming that the model is perfect and has no errors already simplifies the problem somewhat. Since all the errors are zero under that assumption the covariance matrix of the model errors C_q is zero and the objective function becomes:

$$J(\theta,\psi) = \int_D (\theta-\theta^f)^T C_\theta^{-1}(\theta-\theta^f)\mathrm{d}x + (d-H\psi)^T C_d^{-1}(d-H\psi). \tag{3.25}$$

With this assumption the optimal iterative relation has been proben to be [4],[5]:

$$\begin{bmatrix}\psi_i^a \\ \theta_i^a\end{bmatrix} = \begin{bmatrix}\psi_i^b \\ \theta_i^b\end{bmatrix} + \begin{bmatrix}C_\psi & C_{\theta,\psi} \\ C_{\theta,\psi} & C_\theta\end{bmatrix} H^T \left(H \begin{bmatrix}C_\psi & C_{\theta,\psi} \\ C_{\theta,\psi} & C_\theta\end{bmatrix} H^T + C_\theta \right)^{-1} \left(d - H \begin{bmatrix}\psi_j^b \\ \theta_i^b\end{bmatrix} \right). \tag{3.26}$$

To be more precise, the relation give in equation(3.26) minimizes the uncertainty by combining the data and the model state optimally. Note that in order to do this, the uncertainty of the model and of the measurements need to be known. When the data is self simulated such as the case for the twin experiment, the covariance matrix can be calculated as long as the model is linear. For non linear systems such the one in this thesis, the method needs an extension called the Ensemble Kalman Filter(EnKF).
All Ensemble methods make use of an ensemble of realisations and use these so called ensemble members to store the information on uncertainty of the system. The EnKF set up would start with creating a collection of N parameters θ_i that are realisations of the prior distribution of the parameters. For each of the N θ's the model is run, which gives N different realisations ψ_i. Now the general recursive relation that gives the variance minimising solution for the parameter has been shown to be [4],[5]:

Where the covariance matrices can be numerically constructed from the ensembles themselves [14]. The recurrence relation (3.26) gives a new set of values for θ_i and the whole process of propagating the model to get values for ψ_i on the next time and estimating the new covariance matrices can start again.
The linearity of the recurrence relation given in equation (3.26) makes the EnKF cheap to compute. This is why Ensemble Kalman methods are often used in problems with very large amounts of parameters. Since the method ends up with an Ensemble of parameters, the confidence intervals require very little extra work. The Ensemble should already have the correct distribution so all the knowledge necessary to quantify the uncertainty is directly available. The downside of the method is of course that all errors must be presumed Gaussian.

Algorithm 2 Ensemble Kalman Filter

1: Initiate Ensemble members $\theta_i \leftarrow$ relisation of chosen random variable
2: Construct C_d from known data errors
3: Construct C_θ form prior distribution of θ
4: Initialize model for each ensemble member
5: **for** #data **do**
6: Construct$C_{\theta,d}$ from ensemble memebers
7: **for** # ensemble member **do**
8: $$\begin{bmatrix} \psi_i^a \\ \theta_i^a \end{bmatrix} = \begin{bmatrix} \psi_i^b \\ \theta_i^b \end{bmatrix} + \begin{bmatrix} C_\psi & C_{\theta,\psi} \\ C_{\theta,\psi} & C_\theta \end{bmatrix} H^T \left(H \begin{bmatrix} C_\psi & C_{\theta,\psi} \\ C_{\theta,\psi} & C_\theta \end{bmatrix} H^T + C_\theta \right)^{-1} \left(d - H \begin{bmatrix} \psi_i^b \\ \theta_i^b \end{bmatrix} \right)$$
9: Forward model ψ_i^a to next time where data is available using

$$\psi_{t+\delta t}^b = f_\theta(\psi_t^a). \tag{3.27}$$

3.2.2. Particle filter

The variance minimising characteristic of the Kalman Filter makes it a very attractive method. However, when the density of $p(\theta|d)$ has a significant third moment, it is impossible to represent using a Gaussian distribution. Also, shapes like a bimodal probability density function would be impossible to recognise from a Gaussian model. A solution for this problem can be found in the Particle filter method. The particle method is a generalisation of the EnKF method and does not assume Gaussian errors. What is preserved is the sequential nature of the EnKF.
The basis of the method lies in the fact that the state can be modelled as a Markov chain. The data is then modelled conditionally independent. The key of the Particle filter is the principle of importance sampling[4], [19]. This principle will be explained here.
Assume a set of states $\psi_{1:k}$ at the times t_1 up to t_k, a set of measurements $d_{1:k}$ of all measurements up to time k and a set $\psi_{1:k}^i$ of support points with associated weights w_k^i. The weights will sum to one. Then it is true that:

$$p(\psi_{0:k}|d_{1:k}) \approx \sum_i w_k^i \delta(\psi_{1:k} - \psi_{1:k}^i), \tag{3.28}$$

for $\delta(\cdot)$ the Dirac delta measure. The principle of importance sampling lies in finding the right values for the weights. Assume a probability density π that is proportional to $p(\psi_{0:k}|d_{1:k})$, where $\pi(x)$ is easily evaluated but hard to draw from. Now for a set of realisations $\psi^i \sim q(\cdot)$, which is called the importance density , the approximation to the density is given by weights:

$$w_k^i \propto \frac{p(\psi_{0:k}^i|d_{1:k})}{q(\psi_{0:k}^i|d_{0:k}^i)} \tag{3.29}$$

Looking at the sequential situation, relation (3.29) becomes:

$$w_k^i \propto \frac{p(d_k|\psi_k^i)\,p(\psi_k^i|\psi_{k-1}^i)\,p(\psi_{0:k-1}^i|d_{1:k-1})}{q(\psi_k^i|\psi_{0:k-1}^i, d_{1:k})\,q(\psi_{0:k-1}^i|d_{1:k-1})} = w_{k-1}^i \frac{p(d_k|\psi_k^i)\,p(\psi_k^i|\psi_{k-1}^i)}{q(\psi_k^i|\psi_{0:k-1}^i, d_{1:k})}. \tag{3.30}$$

Now the density function can be approximated by:

$$p(\psi_{0:k}|d_{1:k}) \approx \sum_i w_k^i \delta(\psi_{1:k} - \psi_{1:k}^i). \tag{3.31}$$

The biggest known problem with the particle filter is the concept of degeneracy. This entails that after a few iterations all but one particle will have negligible weight. It is proven that indeed the variance of the particles increase over time [13]. This means that the degeneracy problem cannot be avoided. The speed at which the variance increases can however be reduced. The concept of re-sampling can reduce the degeneration effect by terminating particles with a very small weight and focussing on particles with a larger weight. This method however has some drawbacks too. By resampling some particles the particles are no longer independent and some convergence results on this method no longer hold. From a more practical point of view resampling also introduces some limitations. No longer can the particles be treated parallel, since they are all needed together for the resampling.
This was a general motivation of the Particle filter for state estimation. For the experiments in this thesis the goal is to estimate parameters. So instead of $p(\psi|d)$ the aim is to be able to sample from $p(\psi,\theta|d)$ or better yet $p(\theta|d)$. In a Markov setting, methods usually combine the Markov Chain Monte Carlo method (MCMC), which will be explained in more detail in section 3.2.3, with a Sequential Monte Carlo Method (SMC) such as the Particle filter described here. One such method will be discussed now to point out some general ideas on how to iteratively estimate $p(\theta|d)$ and some shortcomings of these methods. The method is called the Particle Marginal Metropolis-Hastings (PMMH) which samples form a distribution that approximates $p(\theta,\psi|d)$. Just like the MCMC the algorithm proposes a new draw from parameter θ. Then determines an acceptance probability for that proposed parameter value and does a random draw form a uniform distribution to determine whether to actually accept or not. This concept is called the Metropolis-Hastings algorithm. Back in the context of the Particle filter. Ideally one would like use proposal density:

$$q(\theta',\psi'|\theta,\psi) = q(\theta'|\theta)\,p_{\theta'}(\psi'|d), \tag{3.32}$$

for some proposal $q(\theta',\theta)$ density that quantifies how to propose a new draw for the parameter. The next step would be finding the probability for which the new proposed parameter should be accepted. This probability would then be:

$$\min\left(1, \frac{p_{\theta'}(d)\,p(\theta')\,q(\theta|\theta')}{p_{\theta}(d)\,p(\theta)\,q(\theta'|\theta)}\right). \tag{3.33}$$

It is however not possible to compute from $p_{\theta'}(d)$ and $p_{\theta}(d)$ or to sample from $p_{\theta'}(\psi'|d)$. So the ideal situation will not work. The PMMH gives an approximation of this situation. The key of the solution lies in the fact that the PMMH approximates $p_{\theta'}(\psi'|d)$ using for instance a Particle filter.
The particle filter with its extensions is a method to sample from the distribution of $p(\theta,\psi|d)$. However, the particle filter and the different variations such as the PMMH suffer from degeneration. So effectively sampling requires lots of particles and a lot of resampling. This makes the method computationally demanding.

Algorithm 3 Particle filter

1: Forward model

$$\psi_{t+\delta t} = f_\theta(\psi_t). \tag{3.34}$$

to calculate ψ
2: Propose $\psi^i \sim q(\cdot)$
3: Compute weights w_1^i
4: Normalise weights $W_1^i = \frac{w_1^i}{\Sigma_i w_1^i}$
5: **for** k =# data **do**
6: **if** There are not enough particles with significant weights **then**
7: Resample particles
8: Propose $\psi_k^i \sim q(\cdot|\psi_{1:k}^i, d_{1:k})$
9: Update weights w_k^i
10: Normalize weights $W_k^i = \frac{w_k^i}{\Sigma_i w_k^i}$

3.2.3. Markov Chain Monte Carlo

The Markov Chain Monte Carlo method (MCMC) is a method to evaluate some hard to determine probability distribution. So in this report the goal of the MCMC is to sample from $p(\theta|d)$. The Metropolis-Hastings (MH)

Algorithm 4 Particle marginal Metropolis-Hastings

Choose θ_0 as initial guess for the parameter
Run Particle filter to estimate $p_{\theta_0}(\psi_{0:k}|d_{0:k})$ and sample $\psi \sim p_{\theta_0}(\cdot|d_{0:k})$
Sample $\theta' \sim p(\cdot|\theta)$
while stop = 0 **do**
    Run Particle filter to estimate $p_{\theta'}(\psi_{0:k}|d_{0:k})$ and sample $\psi' \sim p_{\theta_0}(\cdot|d_{0:k})$
    With probability

$$\min\left(1, \frac{p_{\theta'}(d)\,p(\theta')\,q(\theta|\theta')}{p_{\theta}(d)\,p(\theta)\,q(\theta'|\theta)}\right). \tag{3.35}$$

    accept θ'
    if θ' is accepted **then**
        Add θ' to the set of draws for θ
        Set $\theta = \theta'$ and set $\psi = \psi'$
    if Stopping criteria is met **then**
        Stop = 1

algorithm is a versatile MCMC method that is often used in physics [1]. The MH algorithm is, just as the PMMH, based on selecting the correct proposed parameter values to build a set of draws for the parameter that belong to the correct distribution. The basic mathematics behind the method will be explained.
Assume a Markov Chain $\Theta_t, \Theta_{t+1} \ldots$ with known probabilities for which Θ_t transitions to Θ_{t+1}. This Markov Chain is not from the distribution $p(\theta|d)$. However, if the correct set of realisations of this Markov Chain is chosen from the total Markov Chain, they could be realisations from the desired distribution. In order to select the correct realisations of the Markov Chain, the Hastings decision ratio is necessary [12]. This is given as:

$$\alpha = \min\left(1, \frac{p(\theta'|d)\,p(\theta|\theta')}{p(\theta|d)\,p(\theta'|\theta)}\right) \tag{3.36}$$

Now the suggested Θ_{t+1} is accepted with probability α. This is realized by drawing from a uniform distribution, and only accepted if the draw is smaller then α. There are some remarks to be made about this. First of all, if the random variable that propagates the Markov Chain is symmetric, $p(\theta|\theta') = p(\theta'|\theta)$, then $\alpha = \min\left(1, \frac{p(\theta'|d)}{p(\theta|d)}\right)$. Secondly, for the algorithm to work, it is necessary to be able to evaluate $p(\theta|d) \propto p(d|\theta)p(\theta)$ up to a constant factor. Now $p(\theta)$ can be evaluated when it is assumed a chosen prior. Choosing the prior is basically free, but a good prior will speed up the MCMC methods convergence. Since in fact very little is known about the prior, a uniform distribution is chosen. This is sometimes called an uninformative prior. It only cuts of the probability distribution of $p(\theta|d)$ at certain values that can be motivated by assessing the physical meaning. To evaluate the likelihood of the data,

$$l(\theta) = \frac{1}{(2\pi)^{\frac{N}{2}}\sqrt{|C_d|}} e^{-\frac{1}{2}(d-\psi_\theta)^T C_d^{-1}(d-\psi_\theta)} \tag{3.37}$$

is used. Just as before matrix C_d is the covariance matrix of the data. Since the modelled errors are measurement errors, it seems reasonable to assume independence and Gaussianity. This is why the likelihood is given as equation (3.37). Here ψ_θ is the model outcome for parameter θ used and d is again the measurements from the twin experiment. N is the number of measurements. Since for the proposed parameter values and for the old ones, the measurements are the same, the decision ratio can be written as:

$$\alpha = \min\left(1, e^{\frac{1}{2}(d-\psi_\theta)^T C_d^{-1}(d-\psi_\theta) - \frac{1}{2}(d-\psi_{\theta'})^T C_d^{-1}(d-\psi_{\theta'})} \frac{p(\theta')}{p(\theta)}\right), \tag{3.38}$$

for $p(\theta)$ the prior.
Any Markov Chain begins in some initial state and then starts moving between the different possible states. After a certain amount of time the chain is said to have converged to its equilibrium state. The same goes for the Markov Chain constructed by the MH algorithm. The draws from before the algorithm has converged can formally not be seen as realisations from the target distribution. This is sometimes called the burn in period[12] and these draws have to be removed from the chain. It is difficult however to know when the Markov Chain has converged. For a large set of accepted realisations the chain may seem to move around

a stable mean, but then it still may change and start to explore another area of the distribution. Especially for bimodal or multi modal distributions this is a known phenomenon. This means when the volume of the distribution is not highly connected but spread out over the space in clusters. There are no comprehensive diagnostic methods known now that can be run that will tell the user that the Markov Chain has not yet converged. The best one can do is to simulate a very long Markov Chain.
A diagnostic method that can be used is only feasible for low dimensional parameters. The goal is to know $p(\theta|d) = p(d|\theta)p(\theta)$, the problem is that $p(d|\theta)$ can only be evaluated up to a constant. So in reality it is only possible to compute:

$$p(\theta|d) \propto l(\theta, d)\, p(\theta). \tag{3.39}$$

The true value is given by:

$$p(\theta|d) = \frac{l(\theta, d)\, p(\theta)}{\int l(\theta, d)\, p(\theta) \mathrm{d}\theta}. \tag{3.40}$$

For an one dimensional parameter the integral can be numerically evaluated. This comes in handy when checking the performance of the method.

Algorithm 5 MCMC with Metropolis Hastings

1: Set $theta_0$ for some chosen initial value
2: Propogate model up to end time T for θ_0

$$\psi_{t+\delta t} = f_{\theta_0}(\psi_t). \tag{3.41}$$

3: Construct C_d^{-1}
4: **while** Stop = 0 **do**
5: Draw $\theta' = \theta \sim N(0, \sigma)$
6: Run model up to time T

$$\psi_{t+\delta t} = f_{\theta'}(\psi_t). \tag{3.42}$$

for θ' to obtain $\psi_{\theta'}$
7: Accept θ' with probability

$$\alpha = \min\left(1, e^{-\frac{1}{2}(d-\psi_\theta)^T C_d^{-1}(d-\psi_\theta)+\frac{1}{2}(d-\psi_{\theta'})^T C_d^{-1}(d-\psi_{\theta'})} \frac{p(\theta')}{p(\theta)}\right)$$

8: **if** θ' is accepted **then**
9: Add θ' to set of realisations of θ
10: Set $\theta = \theta'$ and set $\psi_\theta^a = \psi_{\theta'}$
11: **if** Stopping criteria is met **then**
12: Stop = 1

3.3. Summary

Four general types of methods to solve parameter estimation problems in data assimilation were presented in this chapter. Whereas the deterministic calibration does not work within the Bayesian framework, the MCMC algorithm relies solely on bayes theory. The Newton optimisation can only really give information about the certainty of the estimate when the errors are assumed Gaussian. The same goes for the Kalman Filters. The particle filter and the MCMC are able to find the error for any type of distribution. But two methods are much more costly in computation time. In this project the MCMC and the deterministic calibration have been implemented. These are the computationally easiest and a computationally harder method. These implementations should make it possible to see whether a Gaussian assumption on the errors is reasonable. It should also allow for a thorough investigation on the effects of differences in the quality of the data on the found parameters.

4

Parametrisations

There is a wide variety available of different parametrisations of the relative permeability curves. The true model makes use of the Corey parametrisations described in equation (4.1) and (4.2). These parametrisations motivate the first parametrisation suggested in this chapter. The different parametrisations here are just a small subset of all possible approaches. Many more can be thought of but for the purpose of investigating the effect of parametrisations on uncertainty this subset will suffice. They vary in degrees of freedom and in how rigorously they prescribe the known physical demands on the relative permeability curve.

4.1. Corey parameter based parameter estimation

The Corey curves used for this project:

$$k_{ro}(S_o) = \left(\frac{S_o - S_{or}}{1 - S_{or}}\right)^{\frac{2+3\theta_o}{\theta_o}}, \tag{4.1}$$

$$k_{rw}(S_o) = \left(\frac{1 - S_o}{1 - S_{or}}\right)^2 \left(1 - \left(\frac{S_o - S_{or}}{1 - S_{or}}\right)^{\frac{2+\theta_w}{\theta_w}}\right), \tag{4.2}$$

have parameters θ_o for the oil curve and θ_w for the water curve. Allowing this parameter λ to be free gives a set of curves with each having one degree of freedom. Because this parametrisation is a power law in its core it is very constrained. The parameters θ_w and θ_o are defined on $\mathbb{R}\backslash\{0\}$ but the curves converge to a certain limit curve when the parameters become very small or large.
Since the true parametrisation used in the model is included in the set of all parametrisations that can be made with these Corey curves, it should be fairly easy to retrieve the curve.

4.2. Spline based parametrisation

Instead of prescribing the entire shape it is also possible to demand less of the estimated curve. This gives a large amount of freedom for the relative permeability curve. This freedom allows the curve to exit the feasible solution space. The demands on continuity of the second derivative and on monotonicity, that are known to be true for the relative permeability, are not prescribed. A suitable parametrisation allows an algorithm to search within the feasible solution space. This results in quicker convergence then a parametrisation that also allows solutions that lie outside of this feasible solution space. In its basic form a spline will allow solutions that do not comply with the monotonicity demands. This is something to be mindful of when using such a parametrisation and might demand the use of a regularisation term in the object function. Two different splines will be suggested in this chapter.

4.2.1. Linear spline

The most simple spline that is appropriate here is a linear spline. This means that there are knots at a given set of points S_i between zero saturation and saturation one. And the value of the corresponding knot $Z_{i,o}, Z_{i,w}$ is

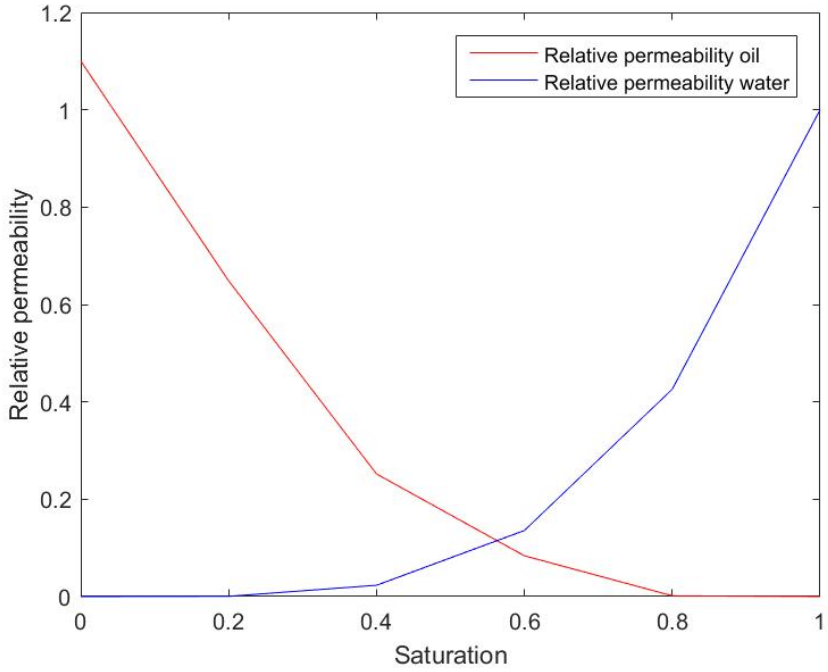


Figure 4.1: Set of example relative permeability curves for a linear spline parametrisation

the parameter that is to be estimated. All the knots combined in one vector also referred to as vector θ and are the parameters that the data assimilation method tries to estimate. The curve is then constructed by drawing a line from knot $Z_{i,\cdot}$ to $Z_{i+1,\cdot}$. A typical example of such a set of curves is given in figure 4.1.

In this parametrisation there is no demand for the relative permeability to be a monotonic function of the saturation. It is known that the relative oil permeability is zero at saturation one and that the relative water permeability is zero at zero. At these points the parameter can be set and does not have to be optimised. For this project the endpoint of the relative permeability curves are assumed to be known before starting the automatic history matching. The end points are for the oil curve the value of the relative permeability at saturation 0 and for the water relative permeability curve the value at saturation 1. This is done because due to the connate water and the residual oil, the ends of the curves shall never be used by the model. The saturation levels will never be that high or low. So it does not matter where the end points of the curves are exactly, but it makes the optimisation a lot easier.
For a random normal variable vector the inverse of the Hessian matrix equals the covariance matrix. This means that calculating the inverse of the Hessian matrix gives information on the quality of the solution. The confidence bounds for these spline solutions for the Newton method will be derived using this relationship. For the MCMC the confidence bounds can be derived from the sample of the parameter.

4.2.2. Smooth splines

Whereas the linear spline is easy to implement it is not continuously differentiable. Since the relative permeability is assumed to be continuously differentiable a linear spline might not be suitable. Whether it is a good parametrisation or not will be seen in chapter 5.
An alternative that is commonly chosen is cubic B-splines[9],[21]. The Basis spline is often referred to as a B-spline. The higher order B-spline gives smooth polynomial solutions, exactly as expected for the relative permeability.
The B-spline is formed by summing different basis functions, $B_{i,k}(S)$ that are defined on saturation intervals $[s_i, s_j]$. This gives $K_{r\cdot} = \sum_i Z_i B_{i,n}$. The Z_i are called the controlling knots. The first order basis function is defined as

$$B_{i,1}(S) = \begin{cases} \text{If } S \in [s_i, s_{i+1}] & 1 \\ \text{Else} & 0 \end{cases}$$

Then the higher order basis functions are defined as $B_{i,k}(S) = \frac{S-s_i}{s_{i+k-1}-s_i} B_{i,k-1}(S) + \frac{s_{i+k}-S}{s_{i+k}-s_{i+1}} B_{i+1,k-1}(S)$. This means that for a B-spline of order k, there is contribution from $B_{i-k-1,k}$ upto $B_{i,k}$.
One of the things that sets B-splines apart from other splines is that the spline does not necessarily pass through the controlling knots (s_i, Z_i). Instead the spline is more or less drawn in the direction of the controlling knots. It is possible however to ensure that the endpoints go through the end knots. This is ensured using equations (4.3) and (4.4).

$$Z_{-1} = 2Z_0 - Z_1 \tag{4.3}$$

$$Z_{n+1} = 2Z_n - Z_{n-1} \tag{4.4}$$

It is important that the splines go through the end knots because splines are very sensitive to overshoot. There will be no data available at the end points because of the residual oil and connate water. A significant overshoot will have an effect that reaches far further then an error made by setting the endpoint slightly wrong. The error made at the endpoint will be greater than the error made by setting the endpoint to a reasonable value.
Although there are smooth curves, there still is an issue with these splines. The problem is non-monotonic relative permeability curves. It is known that the relative permeability curve should be monotonic, so it is convenient if the paramterisation does not allow for non-monotonic curves. In the current situation, the controlling knots still move freely through the domain and nothing ensures that the basis points stay monotonic. What is proven however is that when the control points are monotonic, so is the spline [20]. So ensuring monotonic basis point is sufficient for allowing only monotonic curves.
An example of what the spline looks like is give in figure 4.3.

Modelling monotonic knots
Ensuring that the knots Z_i are monotonic can be done by transforming them from parameters x_i, which are the pseudo parameters to true paramters Z_i. For the oil relative permeabilities a suitable transformation is the inverse of[21]

$$x_i = \ln\left(\frac{Z_i - Z_{i-1}}{Z_{i+1} - Z_i}\right).$$

For the water relative permeability the transformation is the inverse of:

$$x_i = \ln\left(\frac{Z_i - Z_{i+1}}{Z_{i-1} - Z_i}\right).$$

This transformation is an alternative to adding the regularisation terms to the object function. In this case the reverse transformation means solving (4.5) for the water relative permeability.

$$\begin{bmatrix} -1 & \frac{1}{e^{x_1}+1} & & & \\ \frac{e^{x_2}}{e^{x_2}+1} & -1 & \frac{1}{e^{x_2}+1} & & \\ & \ddots & \ddots & \ddots & \\ & & \frac{e^{x_{n-1}}}{e^{x_{n-1}}+1} & -1 & \frac{1}{e^{x_{n-1}}+1} \\ & & & \frac{e^{x_n}}{e^{x_n}+1} & -1 \end{bmatrix} \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_{n-1} \\ Z_n \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \frac{Z_{n+1}}{(e^{x_n}+1)} \end{bmatrix} \tag{4.5}$$

And and (4.6) for the oil relative permeability.

$$\begin{bmatrix} 1 & -\frac{1}{e^{x_1}+1} & & & \\ -\frac{e^{x_2}}{e^{x_2}+1} & 1 & -\frac{1}{e^{x_2}+1} & & \\ & \ddots & \ddots & \ddots & \\ & & -\frac{e^{x_{n-1}}}{e^{x_{n-1}}+1} & 1 & -\frac{1}{e^{x_{n-1}}+1} \\ & & & -\frac{e^{x_n}}{e^{x_n}+1} & 1 \end{bmatrix} \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_{n-1} \\ Z_n \end{bmatrix} = \begin{bmatrix} \frac{e^{x_i} Z_0}{(e^{x_1}+1)} \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \tag{4.6}$$

So for each relative permeability curve a system needs to be solved.
To indicate the freedom this transformation for the oil relative permeability still allows a plot of a hundred transformed random uniform vectors is presented in figure 4.2. So to create one of the curves a vector of ten randomly uniform distributed variables is drawn. Then this random set is transformed using the transformation mentioned above. The resulting Z vector is then drawn in figure (4.6). Note that this transformation does not give only convex curves but still allows for quite some freedom. A curve that is presented here would

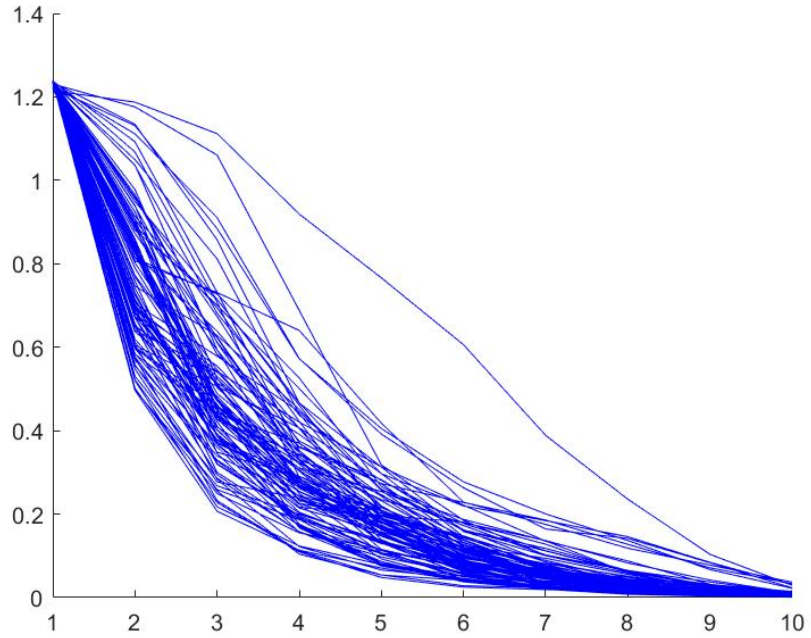


Figure 4.2: 100 Transformed uniform random vectors indicate what possible curves this transformation gives for the oil relative permeability knots.

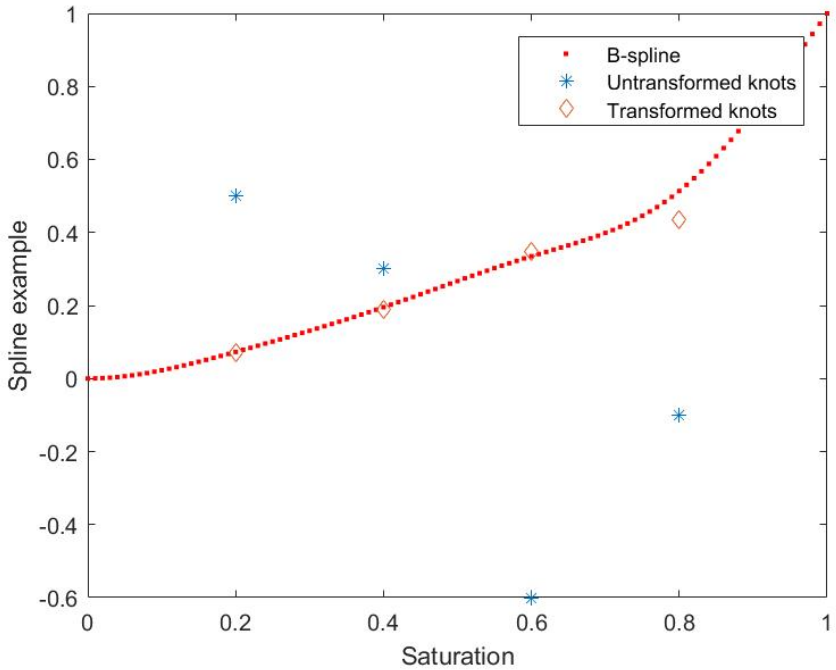


Figure 4.3: An example set of parameters for the relative water permeability, the transformation of that set of variables and the spline based of those transformed knots.

be the knots of a B-spline, for a oil relative permeability with ten degrees of freedom. Note that the amount of degrees of freedom is chosen arbitrarily for this example and equals 4 for the actual experiments. A concrete example of the transformation is given in figure 4.3.

Now the use of the transformation will be given. The optimisation has a set of pseudo parameters x_i. Then before the model is run the pseudo parameters are transformed to the real parameters Z_i. The model is run with the Z_i parameters and the model outcome determines the next step. The transformation became part of the parametrisation and at the end of the parameter estimation the algorithm gives an estimate of the pseudo parameters x_i. Which then have to be transformed to the corresponding Z_i parameters, to be able to plot the optimal relative permeability curves. For instance the callibration method would evaluate the object function as a function of the set parameters x_i, and calculate all the gradients relative to x_i. This makes it slightly harder to translate the given uncertainty on x_i to the uncertainty on the final relative permeability curves.

Confidence intervals for transformed knots

A standard implementation of the confidence bounds is not suitable for the monotonic splines described in section 4.2.2. This is because the transformation from the pseudo knots to the used knots is a nonlinear one. A common method to overcome this issue is to approximate the knots using a linearisation. A standard Taylor expansion for a nonlinear function f is:

$$f(\bar{z}) = f(\bar{p}) + \nabla f(\bar{p})|_{\bar{p}}(\bar{z} - \bar{p}).$$

This theory then gives that it is needed to solve:

$$\begin{bmatrix} -1 & -\frac{e^{x_1}}{(e^{x_1}+1)^2} & & & \\ \frac{e^{x_2}}{(e^{x_2}+1)^2} & -1 & \frac{-e^{x_2}}{(e^{x_2}+1)^2} & & \\ & \ddots & \ddots & \ddots & \\ & & \frac{e^{x_{n-1}}}{(e^{x_{n-1}}+1)}^2 & -1 & -\frac{e^{x_{n-1}}}{e^{x_{n-1}}+1} \\ & & & \frac{e^{x_n}}{(e^{x_n}+1)^2} & -1 \end{bmatrix} \nabla f(\bar{p}) = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \frac{Z_{n+1}e^{x_n}}{(e^{x_n}+1)} \end{bmatrix} \tag{4.7}$$

and,

$$\begin{bmatrix} 1 & \frac{e^{x_1}}{(e^{x_1}+1)^2} & & & \\ -\frac{e^{x_2}}{(e^{x_2}+1)^2} & 1 & \frac{e^{x_2}}{(e^{x_2}+1)^2} & & \\ & \ddots & \ddots & \ddots & \\ & & -\frac{e^{x_{n-1}}}{(e^{x_{n-1}}+1)^2} & 1 & \frac{e^{x_{n-1}}}{(e^{x_{n-1}}+1)^2} \\ & & & -\frac{e^{x_n}}{(e^{x_n}+1)^2} & 1 \end{bmatrix} \nabla f(\bar{p}) = \begin{bmatrix} \frac{e^{x_i}Z_0}{(e^{x_1}+1)^2} \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}. \tag{4.8}$$

After solving this system the knots that correspond to the confidence bounds are the knots given by $Z + \nabla f(\bar{p})\sigma$, all this is true under the assumption that the parameter error is a Gaussian random variable. Here σ is the local standard deviation deduced from the relation between the Hessian and the covariance matrix. So it is the square root of the diagonal of the inverse of the Hessian. The linear approximation of the transformation introduces new errors because everything is only evaluated locally. However, since the Hessian was already a local numerical approximation, not much precision is lost in comparison to when it was possible to use the exact parametrisation for the confidence bounds.

5

Experiments

The aim of this thesis project is to examine what factors influence the estimates of relative permeability curves and capillary pressure curves with the corresponding uncertainty from core experiments. There are two different areas that one can work on that influence the estimates of such curves. First is of course the core experiment itself. The amount and variety of the experiments determine with which precision the relative permeability can be estimated. Also the measurement errors of those experiments limit the precision up to which the curves can be estimated. Secondly the parametrisations of the curves and the actual estimation method can influence the result. In order to give a good broad representation of what influences the uncertainty and what the possibilities are, three different types of experiments will be presented. The first experiment will concern different parametrisations of the relative permeability curve en two methods to estimate the corresponding parameters. The second experiment will explore estimating the capillary pressure curve.

5.1. Twin experiment

As touched upon before, in this project parameters are retrieved using data. This is done using a twin experiment. The twin experiment is a experiment that uses self generated data to test the method used for retrieving parameters. A twin experiment generally follows the steps below:

- Generate true data using the model and the true parameters
- Add some artificial measurement noise to mimic actual field data
- Run model with some set of initial parameters
- Check difference model run and true data

Now checking the difference between the model run and the truth is done differently for the different data assimilation methods and will be explained for each type of method. The main idea however stays the same for all methods. The true data with added noise is always considered the truth, and using model runs with different parameter settings allows the method to try and find the truth. The fact that the truth is actually known because it was it was generated by the model itself gives the possibility to analyse the quality of the method.
Now the details of the data used for this project will be given. The errors in the data are assumed measurement errors and are assumed Gaussian i.i.d. This means identically independently distributed. So for each data type the data is modelled as $d = H\psi + \epsilon$ for $\epsilon \sim N(0,\sigma)$. The used standard deviations for the different data types are

$$\sigma_S = 0.04, \tag{5.1}$$

$$\sigma_P = 1379.95, \tag{5.2}$$

$$\sigma_o = 0.04. \tag{5.3}$$

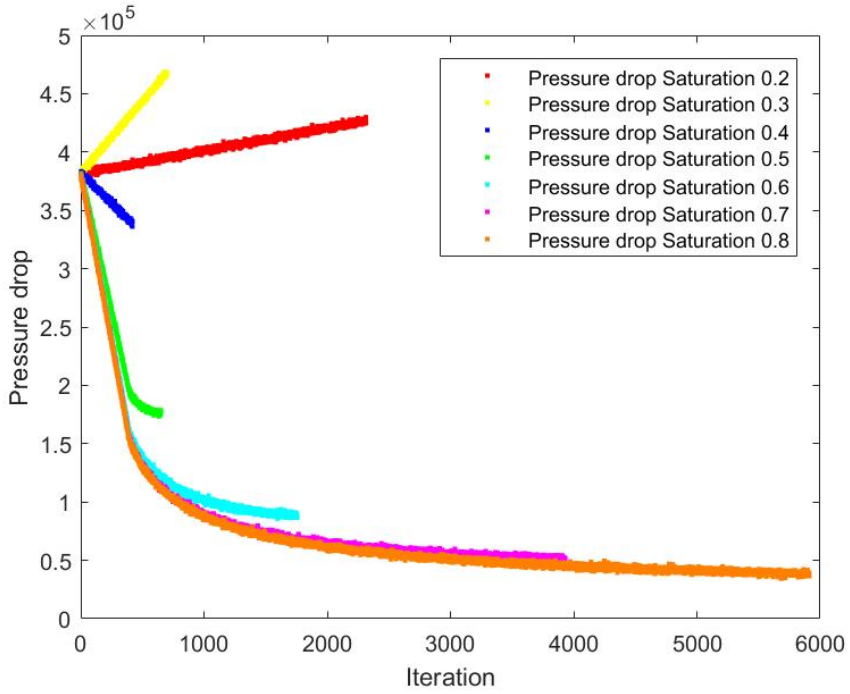


Figure 5.1: Example of pressure drop data

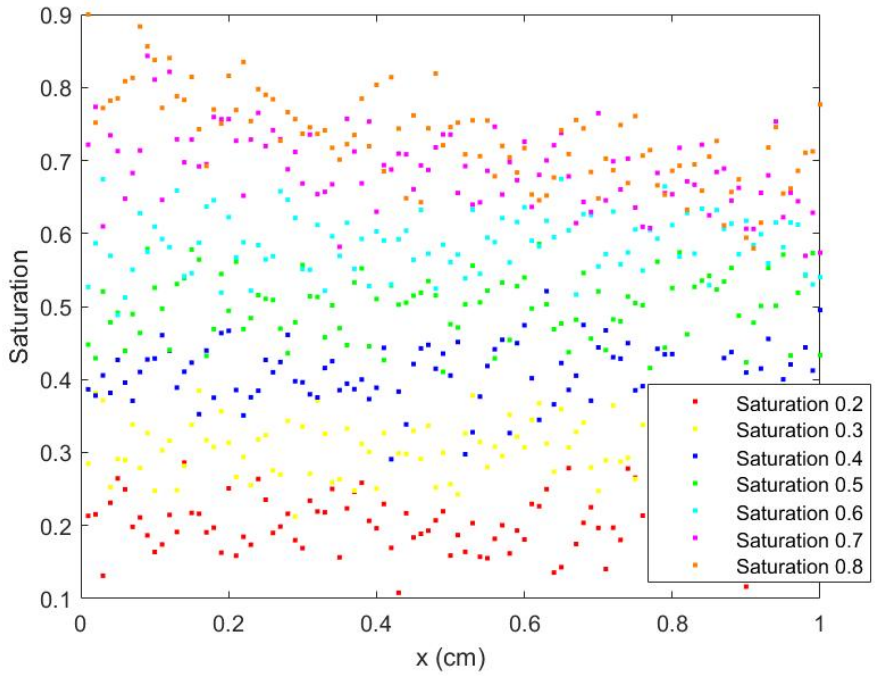


Figure 5.2: Example of Saturation profile data

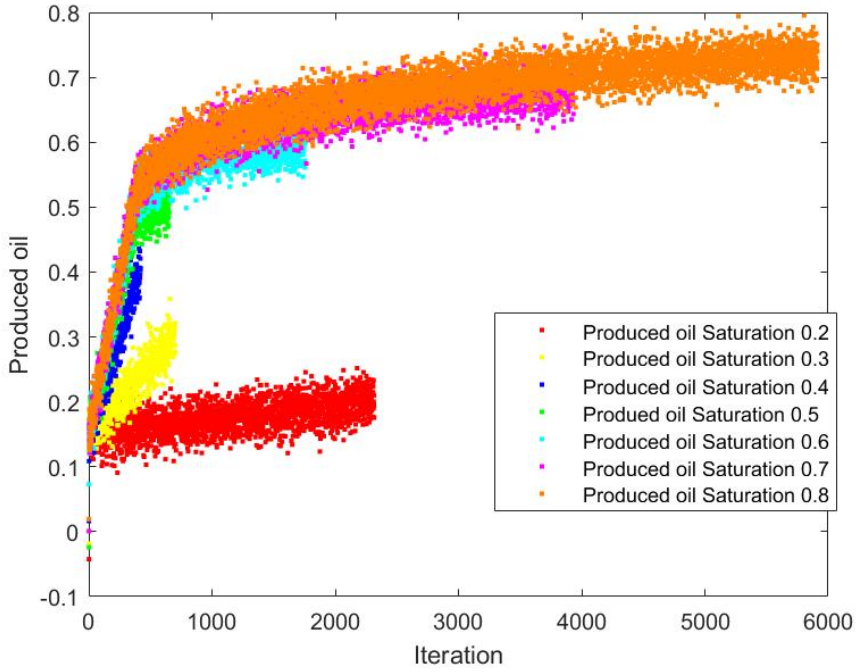


Figure 5.3: Example of produced oil data

Where again the S stands for saturation, the P stands for pressure drop and the o stands for produced oil signal.

It is interesting to see that the pressure does not drop for the low saturation values. This comes from the non-linearity of the relative permeability curves. Together the curves dictate fluctuations in transmissibility. This means how quickly the mixture can move trough the rock. Because the relative permeability curves are non linear, they do not sum to a constant. This explains why for the low saturation value the pressure is increasing, and for the higher saturation values the pressures drop over time. This can be seen in figure 5.1. Also examples of the other data types are given in figures 5.2 and 5.3

5.2. Parametrisations

This experiment will basically set a base line for different parametrisations of the relative permeability curves and what effect the different methods have on these parametrisations. This experiment will compare the following parametrisations:

- A power law estimate where only the Corey parameter is estimated.
- Linear splines with regularisations.
- Transformed B-Splines

The first two will be compared under a calibration scheme and the uncertainty will be determined from the Hessian matrix. The transformed B-splines are more suitable than the other parametrisations to also use in an MCMC scheme. So both of the methods will be applied to the B-splines and the results will be compared. The expectation is that the Power law estimate will be near perfect and will give a very low uncertainty. The downside is that the parametrisation allows for very little amount of freedom and would be a poor fit if the true relative permeability is not so close to the Corey curves. The linear splines do not give a continuously differentiable curve and are thus unable to capture the real relative permeability. The B-Splines should come closer to the real curve then the linear splines. The question is here what happens with the uncertainty and how much iterations does it take both methods when there are the same amount of degrees of freedom for the optimisation.
All the different parametrisations were tested on the same physical experiment. At time zero the core was

filled with a 0.85 oil saturation mixture. First a 0.8 oil saturation mixture was pumped through the core until the steady state was reached. This experiment was repeated with respectively 0.7, 0.6, 0.5, 0.4, 0.3 and 0.2 oil saturation. This gives seven saturation profiles, seven pressure drop signals and seven oil production signals. Because of the complex nature of the model, the required time to reach the steady state varies. When the model is run during the optimisation, the same number of time steps is used as was used in the data generation process.

5.2.1. Power law estimate

The initial value for this experiment for the parameter λ equals $[2,2]$. Using the Newton method the estimate for λ is $[2.0051, 2.0032]$. The result of this optimisation is shown in figure 5.4. The hessian matrix for this result equals:

$$H = 10^4 * \begin{vmatrix} 0.4872 & -1.3904 \\ -1.3904 & 1.5487 \end{vmatrix}.$$

Then σ equals $[-0.0209, -0.0165]$. This is of course not true since, standard deviations are defined positive. So what went wrong? The answer lies with a non positive definite hessian matrix. A continuous, twice differentiable function of several variables is convex on a convex set if and only if its Hessian matrix is positive semidefinite. The standard deviation is derived from the hessian under assumption of Gaussian errors, which would give a convex object function. Even when the errors are not exactly Gaussian it is not an odd assumption that the object function is convex near the minimum. After all, in any direction the object function should be increasing or at least non decreasing. However, when there is a strong dependence in the variables or a variable that has little effect on the model outcome, the object function is not necessarily so well behaved. The object function for the power law estimate is given in figure 5.5. Note that there seems to be a direction where the object function is near constant. So the derived sigma's are not a correct estimate of the uncertainty, because of to strong a dependence in the parameters. The eigenvalues and eigenvectors reflect this result. The eigenvalues of the hessian are $10^4 * -0.1252$ and $10^4 * 1.1476$ and their corresponding eigenvectors are:

$$\begin{bmatrix} -0.7702 \\ -0.6378 \end{bmatrix}, \begin{bmatrix} -0.6378 \\ 0.7702 \end{bmatrix}$$

The eigenvector belonging to the negative eigenvalue is in the direction where there is very little change of the object function [24] [3]. Even though this phenomena is able to distort the ability to estimate the uncertainty from the hessian, it is not expected to cause a highly negative eigenvalue. When the object function is flat it is expected that the eigenvalue is near zero in that direction. To find the cause for this it is necessary to dig a little deeper. For this simple case the true parameters were known. So it is possible to also compute the hessian at the exact optimum. This hessian equals:

$$H = \begin{bmatrix} 0.6118 & 6.5896 \\ -0.7738 & 20.5858 \end{bmatrix}, \tag{5.4}$$

and the diagonal of the inverse equals: $(1.1635, 0.0346)$. So at the exact minimum the hessian does look nice and qualifies as a means to measure the uncertainty. It is interesting to see is what the found optimal value gives for model results. In figure 5.6 the pressure measurements are given. Because of measurement errors the measured pressure values look noisy. The pressure signal that come from the found optimal parameters is also given. Notice that the line goes straight through the measurements without over fitting. Again this result is not surprising, the parametrisation does not allow for reaching each data point. Figure 5.7 gives the saturations of the steady state experiment. Again the result of the model with the estimated parameters is given. This line fits the data well, as expected. In figure 5.8 gives the produced oil for the experiment.

5.2.2. Linear splines

The linear spline is not automatically monotonic and thus it is interesting to see if the regularisation terms in the objective function are necessary. Since the linear splines cannot form a continuously differentiable function such as the Corey curve, the resulting relative permeability curves cannot be the exact Corey curves. So even if the measurement errors would be zero and if the method would be able to retrieve the perfect answer, there will still be a modelling error due to the parametrisation. For this experiment there where 8 degrees of freedom. 4 for the oil relative permeability and 4 for the water relative permeability. These are the knots of the linear spline and are situated at saturations $0.2, 0.4, 0.6, 0.8$. The resulting curves are given in

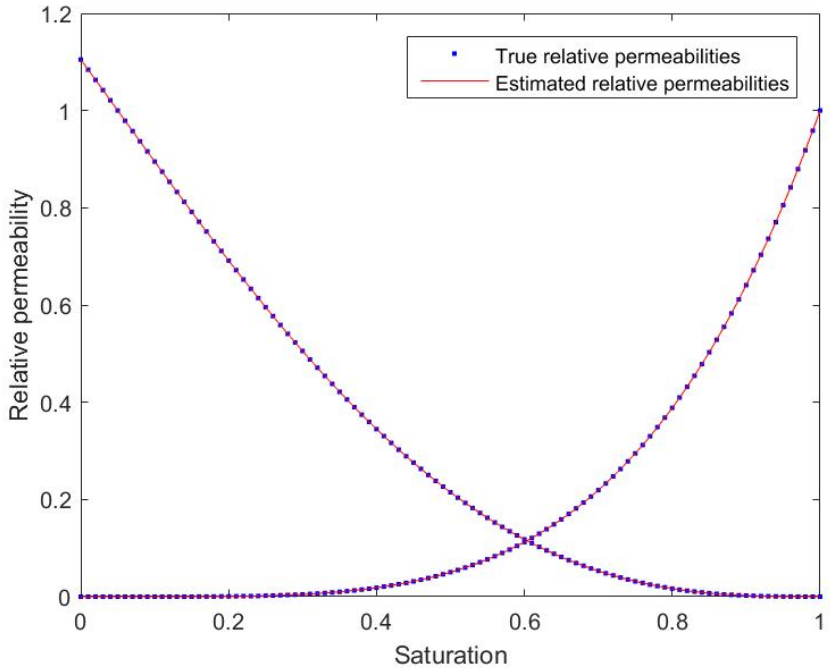


Figure 5.4: Result power law parametrisation

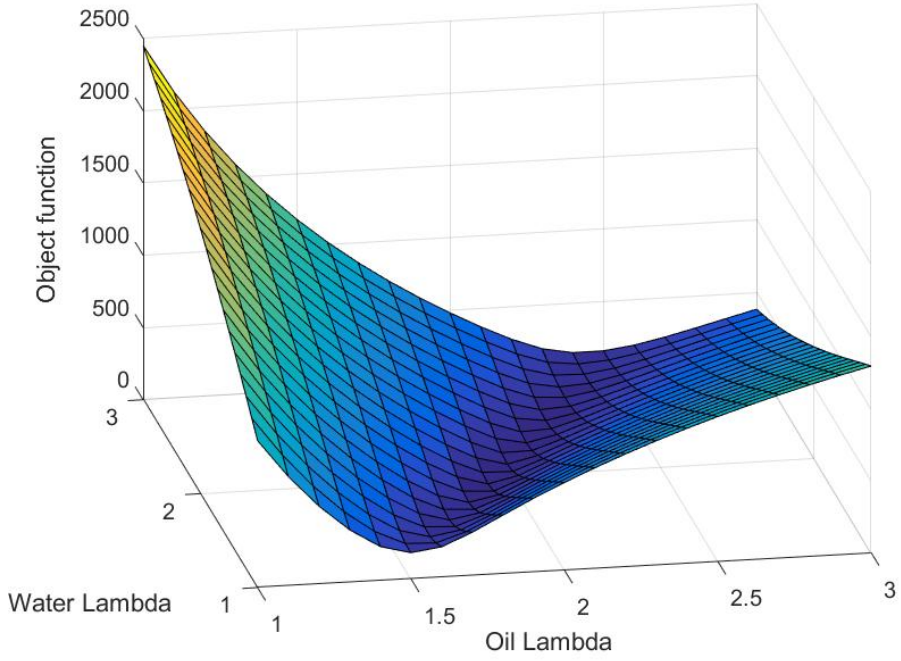


Figure 5.5: Object function for power law estimation

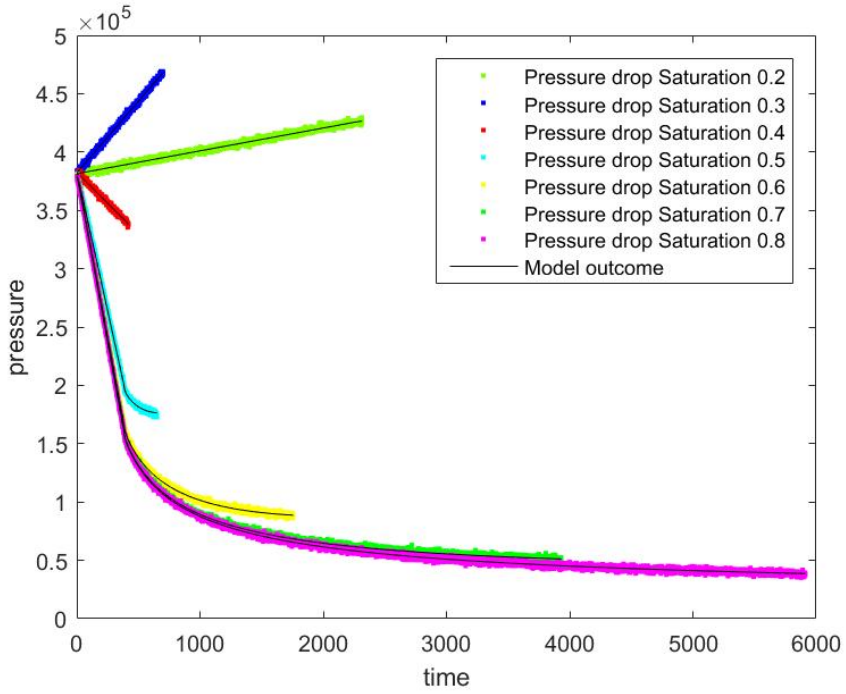


Figure 5.6: Result power law parametrisation for pressure drop

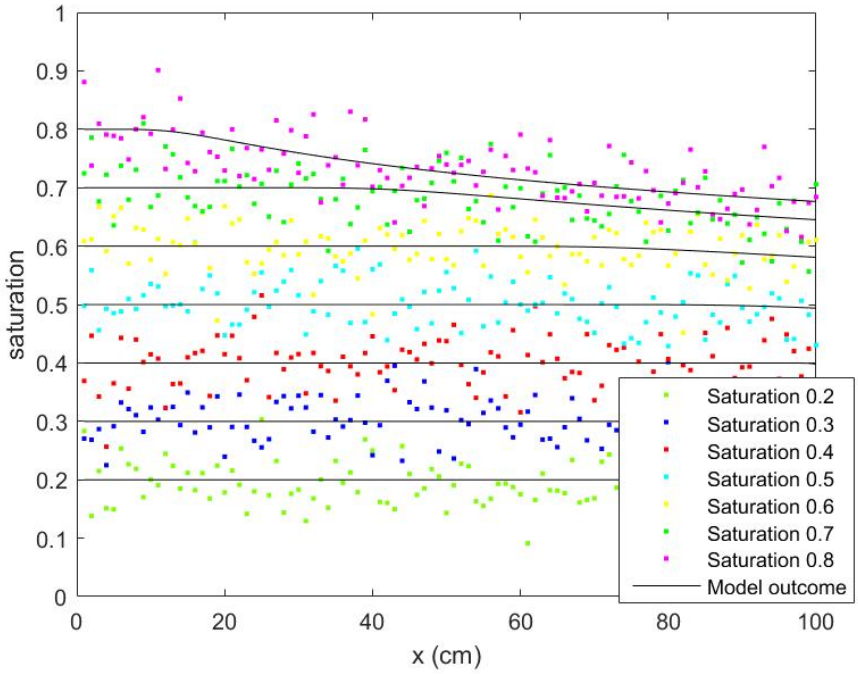


Figure 5.7: Result power law parametrisation for saturation profile

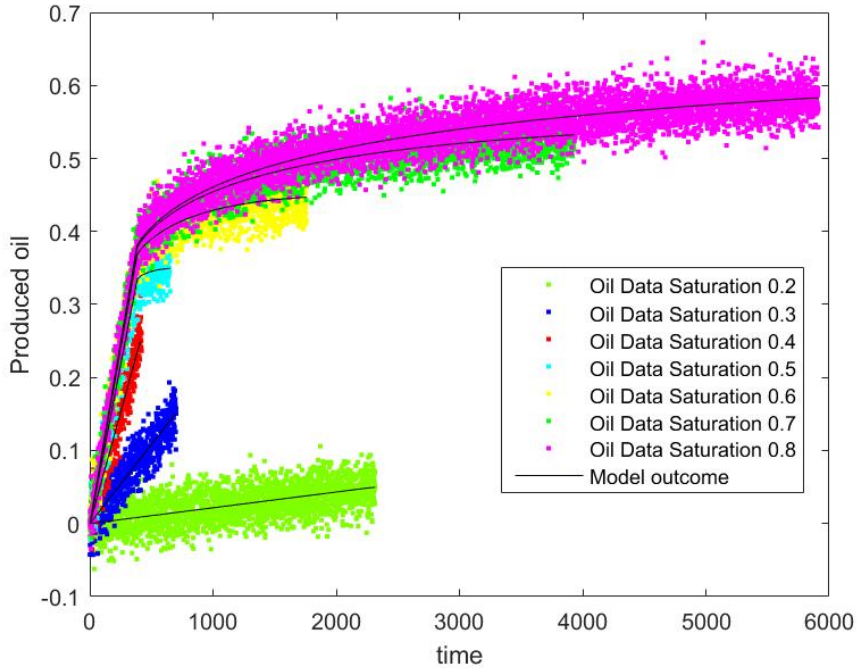


Figure 5.8: Result power law parametrisation for produced oil

figure 5.9. Note that even though the curve has the wrong shape, it is not far from the truth. However, just looking at the curves itself can be very deceiving. The actual place where the quality of the curve is visible would be the model outcome in combination with the measured data.

The pressure signals, the saturation profiles and the produced oil model outcome are given in figures 5.10a,5.10b and 5.10c for some of the saturations. Besides the model outcome, the used measured data is also given. Note that the relative permeability curve image suggested a somewhat reasonable fit but the model outcome and data have some large differences. Especially the pressure curve shows signals that the relative permeability is not a very good fit. The modelled saturation also lies right at the edge of the cloud of measured points. Note that for the power law estimate, the method was able to retrieve the parameter that allowed the model to go straight through the data cloud. This is all the result of an error made in the parametrisations.

For this experiment with no regularisation the resulting parameters where for the oil curve

$$(0.6485, 0.2518, 0.0846, -0.0010)$$

and for the water curve

$$(0.0005, 0.0232, 0.1355, 0.4254).$$

These are the values located at saturations $0.2, 0.4, 0.6, 0.8$. The resulting hessian equals:

$$H = 10^9 * \begin{bmatrix} 0.0003 & 0.0001 & 0.0000 & 0.0000 & 0.0226 & 0.0011 & 0.0000 & 0.0000 \\ 0.0001 & 0.0004 & 0.0001 & -0.0000 & 0.0005 & 0.0021 & 0.0002 & 0.0000 \\ 0.0000 & 0.0001 & 0.0004 & 0.0000 & 0.0000 & 0.0000 & 0.0002 & 0.0000 \\ 0.0000 & -0.000 & 0.0000 & 0.0001 & -0.0000 & -0.0000 & 0.0000 & 0.0000 \\ 0.0226 & 0.0005 & 0.0000 & -0.0000 & 1.3880 & 0.0212 & 0.0004 & 0.0000 \\ 0.0011 & 0.0021 & 0.0001 & -0.0000 & 0.0212 & 0.0532 & 0.0001 & -0.0000 \\ 0.0000 & 0.0002 & 0.0004 & 0.0000 & 0.0004 & 0.0001 & 0.0007 & 0.0000 \\ 0.0000 & 0.0000 & 0.0001 & 0.0000 & 0.0000 & -0.0000 & 0.0000 & 0.0000 \end{bmatrix}$$

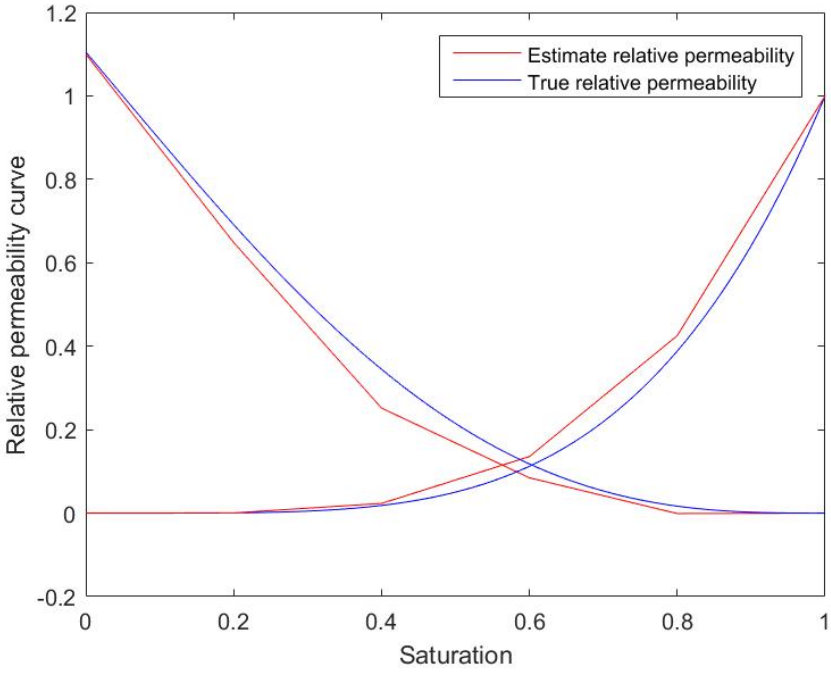
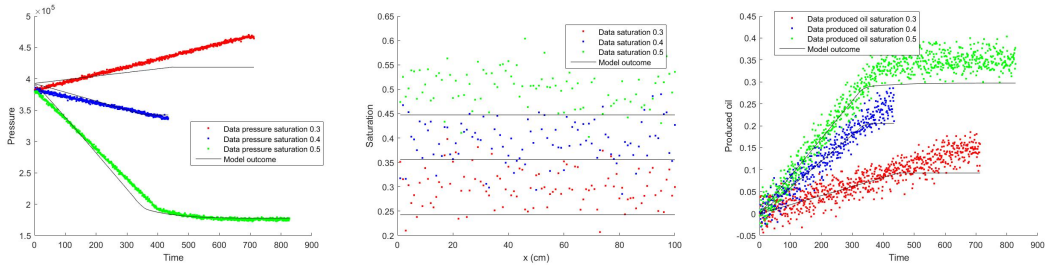
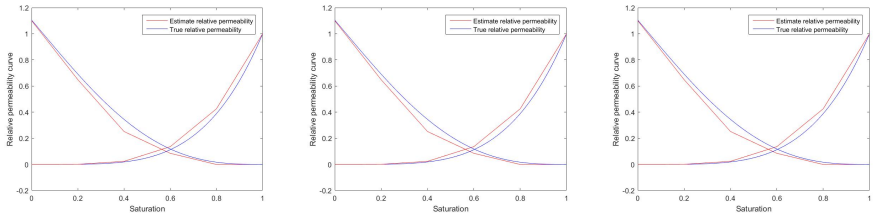


Figure 5.9: Result estimate with linear spline parametrisation with no regularisation term in the objective function



(a) Pressure (b) Saturation (c) Produced oil

Figure 5.10: Result model for optimal parameter using linear splines and no regularisation terms in the objective function and the measured data



(a) No regulation (b) Regulation term 1 (c) Regulation term 2

Figure 5.11: Resulting relative permeability curves for 8 degrees of freedom, linear splines and various regulation in the objective function

The diagonal of H^{-1} the equals:

$$10^{-4} * \begin{bmatrix} -0.0007 \\ 0.0003 \\ 0.0086 \\ 0.0175 \\ -0.0000 \\ 0.0000 \\ 0.0025 \\ 0.2159 \end{bmatrix}$$

. The most notable part of this series of variances, is that not all are positive. This is a problem since variances are defined positive. This is in fact again the same lesson on how hard parameter estimation can be for non-linear systems. The eigenvalues for this Hessian are

$$10^{11} * (-0.0001, 0.0000, 0.0001, 0.0001, 0.0003, 0.0010, 0.0529, 1.3887).$$

And the eigenvector belonging to the negative eigenvalue equals

$$(0.9724, -0.2298, 0.0137, -0.0027, -0.0157, -0.0047, 0.0352, -0.0046).$$

Probably the optimisation was unable to find the exact optimum, which gives an unsuitable Hessian. Since for this case, which is far more complicated than the Corey curves, the correct parameters are not known there is no way to check if a better set of parameters would indeed reduce the negative eigenvalues.
Besides the poor estimate of the Hessian there is another issue with this result, the calibration method is sensitive to finding a local optimum. In fact, doing the experiment again with a variety of initial guesses showed that the method is sensitive to getting stuck in local optima. This could explain the poor data fit and the clear structural error of underestimating the oil relative permeability and overestimating the water relative permeability.

In figures 5.11a, 5.11b and 5.11c. The relative permeability curves are given for a variety of regulation terms in the objective function. The first regulation term is given in equation (3.4) and the second regulation term is given in equation (3.5). Since the original result did already comply with the monotonicity demand, the results are very similar. The difference between the resulting sets of knots is in the order of 10^{-11} and is insignificant. So in this case no regulation is necessary.

5.2.3. B-splines

Deterministic calibration

For the B-splines, calibration becomes even harder. When the knots of the splines are not nicely arranged splines tend to overshoot. This can give negative relative permeabilities which will cause the model to crash. This is why the optimisation becomes much harder when the splines are of higher order. The overshoot can be prevented when limiting the free movement of the knots. This is done by transforming the knots using the transformation as described in section 4.2.2. For the comparison with the linear splines here, the optimisation is done again with 8 degrees of freedom. The order of the B-spline used here equals 4. The order of the B-splines that would give the optimal results are not known in reality. For other applications more research should be done to see whether higher order spline gives better result. Just as for the linear splines the end points of the B-spline are presumed known. Again, since the end points are never reached due to the

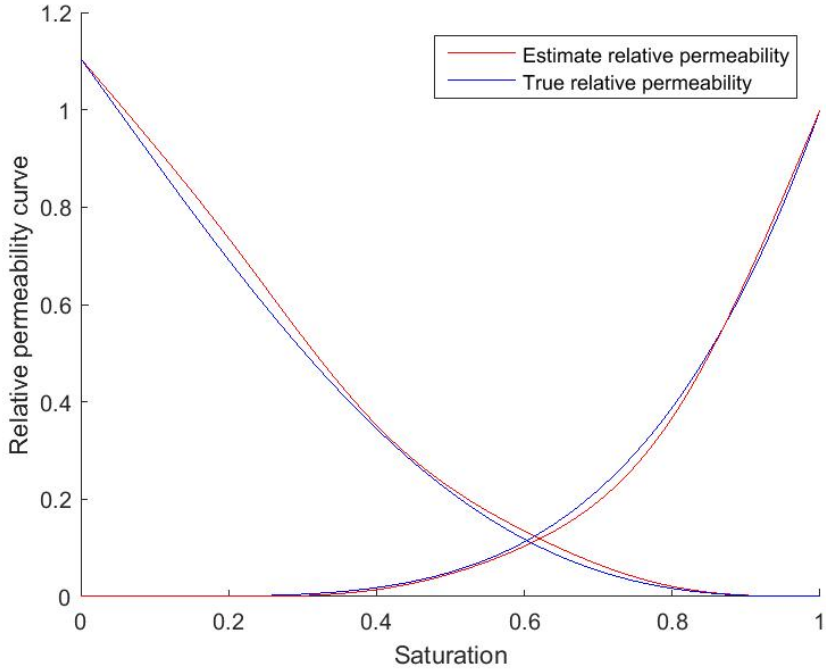
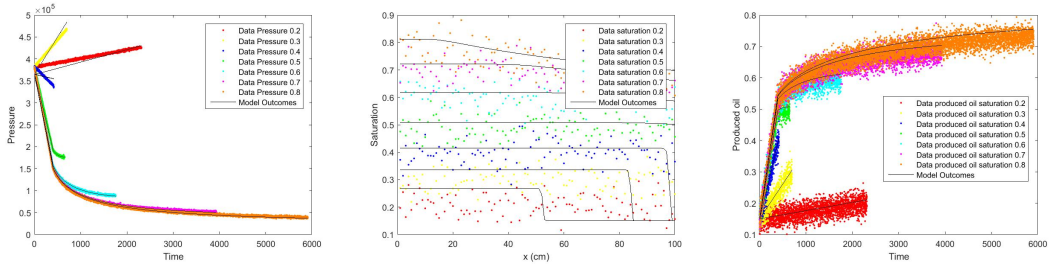


Figure 5.12: Result estimate with Bspline parametrisation with 8 degrees of freedom with Newton optimisation



(a) Pressure (b) Saturation (c) Produced oil

Figure 5.13: Result model for optimal parameter using Bsplines with 8 degrees of freedom and the measured data

residual oil and connate water so the effect of that assumption is reduced. The resulting set of parameters is

$$\theta = \begin{bmatrix} -41.2122 \\ 0.8693 \\ 0.3948 \\ 234.9123 \\ 9.8107 \\ 15.1149 \\ 0.8230 \\ -1.9496 \end{bmatrix}, \tag{5.5}$$

and the corresponding curves are given in figure 5.12. The fit seems not bad and, at first sight, looks better than the result for the linear splines as can be seen in figure 5.9. The true result again can only be seen in the fit of the model outcome in comparison to the data. These are given in figure 5.13a, 5.13c and 5.13b.

The problem with this estimate starts in a bad estimate of the relative permeability curve at the initial saturation value. The effect of the error made here can be seen through the rest of the experiment. Interesting to see is that in comparison to the result of the linear splines given in figure 5.10a , 5.10c and 5.10b the errors are of a different nature than the error made when using the B-spline. The saturation estimates are better for the linear splines but not for the pressure signals and the produced oil estimates. In figure 5.12 no uncertainty

is given. This is because like before the Hessian is not positive definite. The Hessian is given as:

$$H = \begin{bmatrix} 0.0001 & -0.0001 & -0.0000 & -0.0000 & 0.0054 & 0.0194 & 0.0544 & -0.0081 \\ -0.0001 & 8.9046 & 2.6614 & 0.0137 & -0.0934 & -46.4749 & -124.6700 & 1.0250 \\ -0.0000 & 2.6614 & 0.0640 & 0.0300 & -0.0828 & -12.2571 & -33.3467 & 0.9083 \\ -0.0000 & 0.0137 & 0.0300 & 0.0068 & -0.0414 & -1.1606 & -3.2568 & 0.4542 \\ 0.0054 & -0.0934 & -0.0828 & -0.0414 & 0.0023 & 0.0070 & 0.0190 & -0.0028 \\ 0.0194 & -46.4749 & -12.2571 & -1.1606 & 0.0070 & -2.2618 & -1.3014 & -0.0092 \\ 0.0544 & -124.6700 & -33.3467 & -3.2568 & 0.0190 & -1.3014 & 3.5744 & -0.0274 \\ -0.0081 & 1.0250 & 0.9083 & 0.4542 & -0.0028 & -0.0092 & -0.0274 & 0.0040 \end{bmatrix}$$

The diagonal of this matrix inverse than looks like

$$(-86.1412, -0.0726, -1.3445, -2.9501, -4.1895, -1.2882, -0.1818, -0.1368).$$

These values are obviously not variances and when taking a closer look at the eigenvalues. The eigenvalues are

$$(-132.0220, -1.1299, -0.6464, -0.1950, -0.0037, 0.0056, 0.6177, 143.6679),$$

so there are many linear combinations that give a direction for which the object function is non convex. Again this is probably due to the found optimal values not being at a minimum in the object function. But just as for the linear splines, the high dimensionality of the parameter makes it very difficult to find the true optimum. A method that would allow go give a better representation of the uncertainty is the MCMC. The MCMC method should also be able to find the global optimum because it maps out the full distribution of the parameters.

Markov chain Monte Carlo method

A estimate for the relative permeability curves using the MCMC will be given with the corresponding confidence bounds. The parametrisation used here will be the B-Splines. It will be interesting to see how many iterations are necessary to achieve a descent estimate with a reasonable amount of uncertainty. For this method, each relative permeability curve will be allowed 4 degrees of freedom. To check the MCMC there will also be a run with one degree of freedom which will be compared to the non stochastic solution that is computable for a one dimensional parameter. The MCMC implemented here has a uniform prior on a large domain. The prior is defined on the interval $(-1000, 1000)$. Such a prior is sometimes referred to as an uninformative prior because it reveals very little to nothing on the distribution of the parameter. This makes sense because of the synthetic nature of the curve and that so little is known about its nature.

As explained in section 3.2.3, the posteriori function can be calculated if the parameter is one dimensional. To confirm the convergence of the used MCMC algorithm such a posteriori has been calculated. In this case only the relative oil permeability has been chosen. This relative permeability curve has been given one degree of freedom in the form of one knot at saturation equals a half. The parametrisation used is the B-spline model with the transformation. This is done because it compares best to the optimisation with quadratic object function and no regularisation terms. For the water relative permeability curve the Corey curve is used, so this gives no modelling errors.

In figure 5.14 this single parameter is given for all accepted values. The typical burn in period of a MCMC where the algorithm is not yet converged can be seen clearly in this image. The Markov Chain seems stable from the hundredth iteration. For all further analysis of this parameter in this section those first hundred iterations are not used, to prevent the burn-in period from distorting the found results.

In figure 5.15 a scaled histogram is given of the realisations for θ. Also the numerically calculated probability density function is given. The two agree well, which confirms that the realisations from the MCMC method are indeed realisations from the posteriori distribution. Interesting is that the probability density function is slightly asymmetric with a slightly longer tail on the high values for θ than for the low values. This is mostly visible for the numeric calculation but also distinguishable for the histogram. This means that in fact the density function is indeed not Gaussian.

For comparison the object function of the experiment is given in figure 5.16. Note that the object function is not symmetric. This agrees with the result found for the MCMC equivalent. However, for the MCMC this result is quantifiable by calculating the skewness of the set of draws for the parameter, for the Newton method it is not. This is because the error calculated for the Newton method assumes a Gaussian error, and the Gaussian distribution is symmetric. So the error is already assumed symmetric in order determine any measure for uncertainty. The standard deviation from the sample created using the MCMC equals 0.0547. The mean

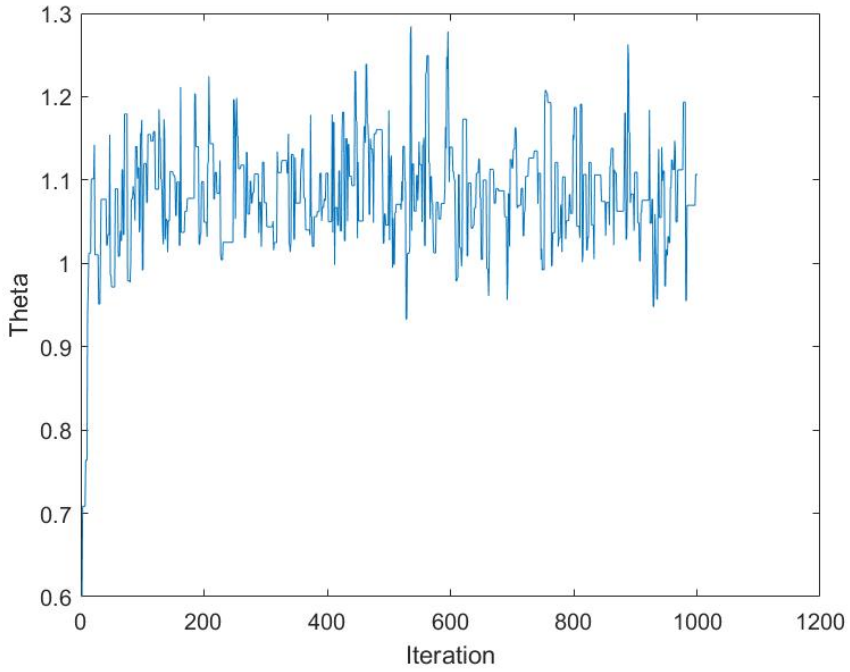


Figure 5.14: The Markov Chain of Theta for the one degree of freedom convergence test

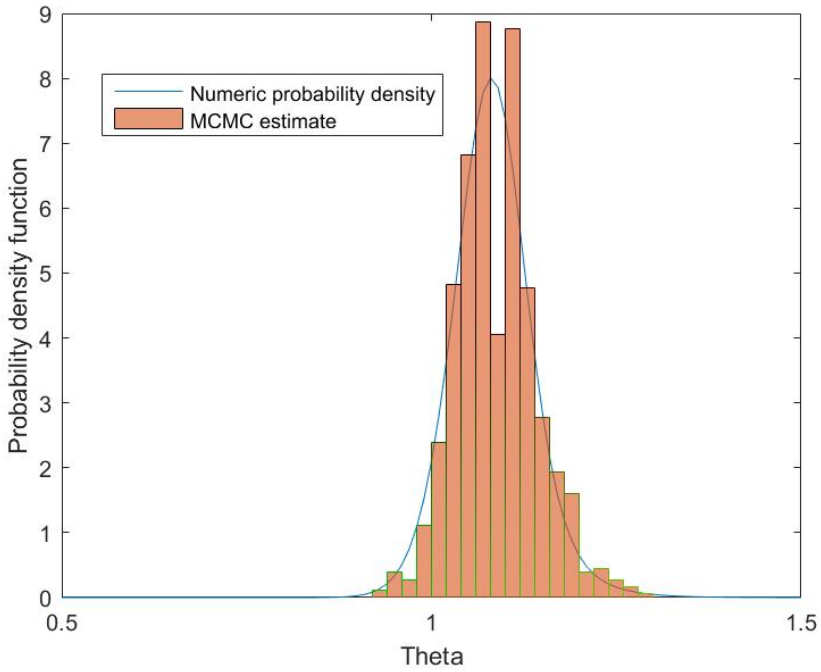


Figure 5.15: Scaled histogram of the realisations of Theta for the one degree of freedom convergence test and numerically found posteriori

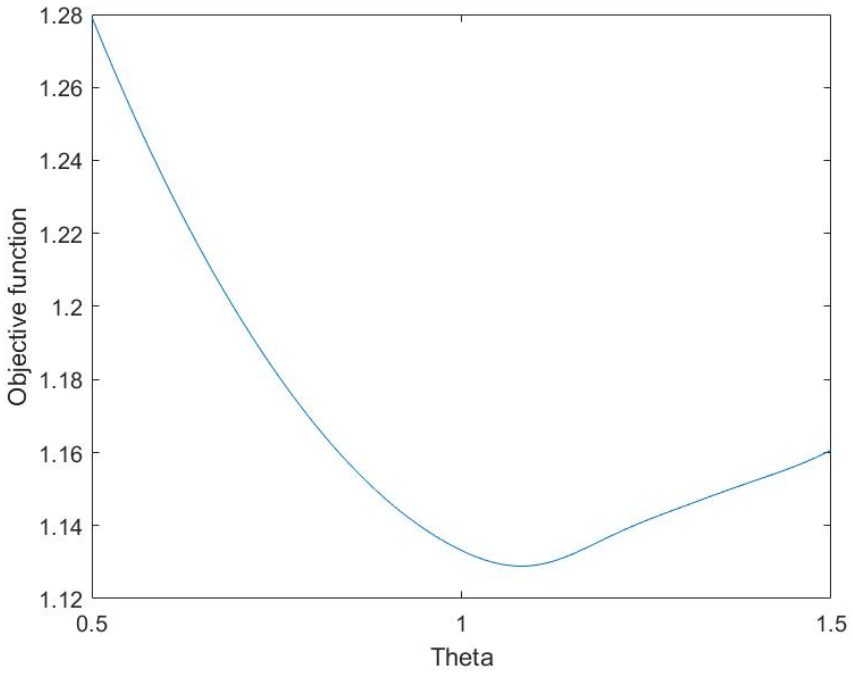
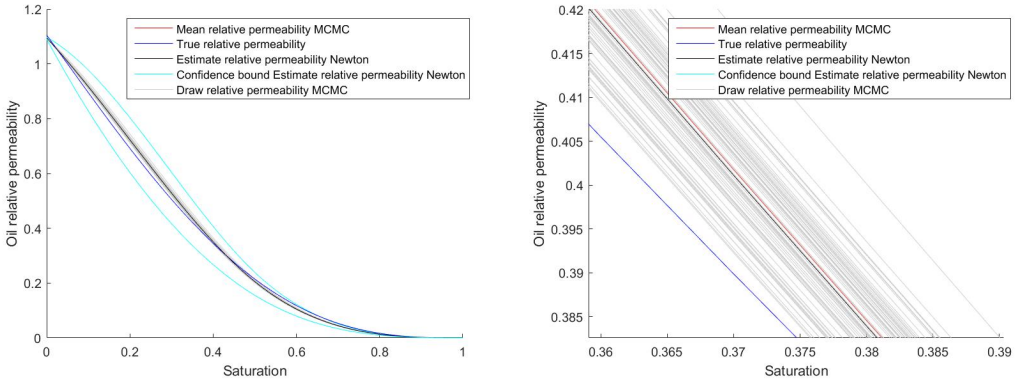


Figure 5.16: Object function of the Newton method for one degree of freedom



(a) Oil relative permeability curves (b) Zoomed oil relative permeability curve

Figure 5.17: The Markov Chain of Theta for the one degree of freedom, the Newton optimisation for one degree of freedom and a hundred realisations of the MCMC draws for θ

equals $\theta = 1.0878$. For comparison the Newton optimisation is also done for one degree of freedom. There the value for the parameter equals $\theta = 1.0811$ with a standard deviation of 0.8189. The result of these two methods is give in figure 5.17a and 5.17b. Interesting is that the found uncertainty for the MCMC is much smaller than that of the corresponding calibration.
The MCMC method has also been applied to the B-splines with 8 degrees of freedom. For this parametrisation retrieving the parameters is much harder than the 1 degree of freedom case described before. One of the problems that can arise in this case is if one of the parameters has little effect on the answer. That parameter will converge very slowly or not at all. Even though not all parameters have converged, the corresponding model outcomes may have converged. The resulting Markov Chains from the MCMC method is given in figure 5.18. Only when the Markov Chain has converged, can the realisations be used to estimate the uncertainty of the relative permeability curve. Some of the parameters have clearly not yet converged, such as θ_8. This parameter has clearly not converged and still moves in a certain direction. However, the final curve depends on all the parameters. Since the Markov Chain should spend the most amount of time at sets of parameters that correspond to high probability in the posteriori distribution, a clearly non-converged Markov Chain is bad news for estimating uncertainty. Another visual way of assessing if the Markov Chain has converged, is to asses the error or the objective function of the Markov Chain. If the Markov Chain has converged, the error should behave rather irregular, and should no longer move in a particular direction for large amounts of consecutive iterations. This is because the MCMC maps the full uncertainty of the parameters. A plot of the object function of the Markov Chain is given in figure 5.19. The error for this figure has been calculated as the value of the object function for the calibration method. The error is calculated every 50th realisation of the Markov Chain. Note that this chain does seem the have converged after a few hundred iterations of the MCMC method. The histograms of the realisations of the parameters are given in figure 5.20.
The means of the found parameter values are:

$$\theta = \begin{bmatrix} -41.1569 \\ 0.8709 \\ 0.4245 \\ 245.9597 \\ 9.7193 \\ 15.1248 \\ 0.8054 \\ -1.7222 \end{bmatrix} \tag{5.6}$$

The estimates from the different methods differ significantly. This can have many causes, such as the estimate of the B-spline was stuck in a local optimum or the Markov Chain has not yet converged.
It seems that there are now two contradicting signals to whether or not the Markov Chain has converged. But these two signals should not be taken as equally important. Not all parameters are of equal importance, or have similar effects on the final model results. It could very well be true that the actual error is only decided by a subset of those 8 parameters used for this experiment. If one parameter has not converged, it does not necessary mean that the object function has not converged. The Markov Chain of the object function gives information on the convergence of the entire full dimensional chain, the individual Markov Chains of the parameters can only say something on the convergence in the individual directions. The model outcomes and their uncertanty are given in figure 5.24b,5.24a,5.23b,5.23a,5.22b and 5.22a. These images show again that the estimate errors can be misleading. The actual error is greater than the one given by the MCMC method. This is because of the limitations of the chosen parametrisations an its inability of taking the shape of the actual curve. It can especially be seen for the pressure signal. The curve for saturation equals 0.2, lies mostly outside of the cloud of measurements. However, if the MCMC estimate of the uncertainty was to be trusted, it would be a very good estimated with small uncertainty.

5.3. Capillary pressure estimates

A twin experiment can also be used to estimate parameters of the capillary pressure curve. This is done using a slightly different experiment than for the relative permeability curves. The cores used for estimation of the Capillary pressure must be far shorter than the ones used for the estimation of relative permeabilities. This is because the effect of the capillary pressure are mostly visible and the outflow end of the core, so decreasing the size of the core makes the effects at the edge better distinguishable. So for this experiment a core of 6 centimetres is used.

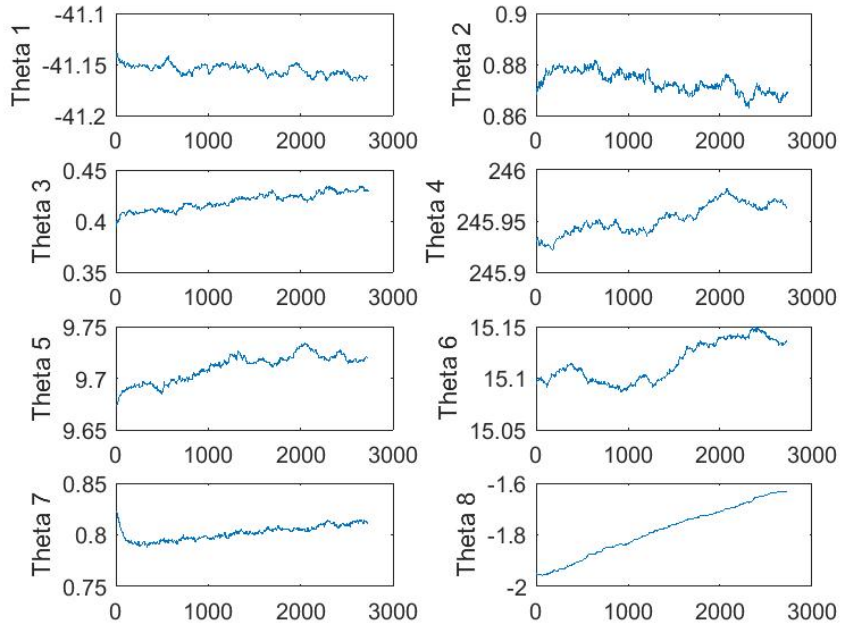


Figure 5.18: Realisation of the MCMC for B-splines with in total 8 degrees of freedom

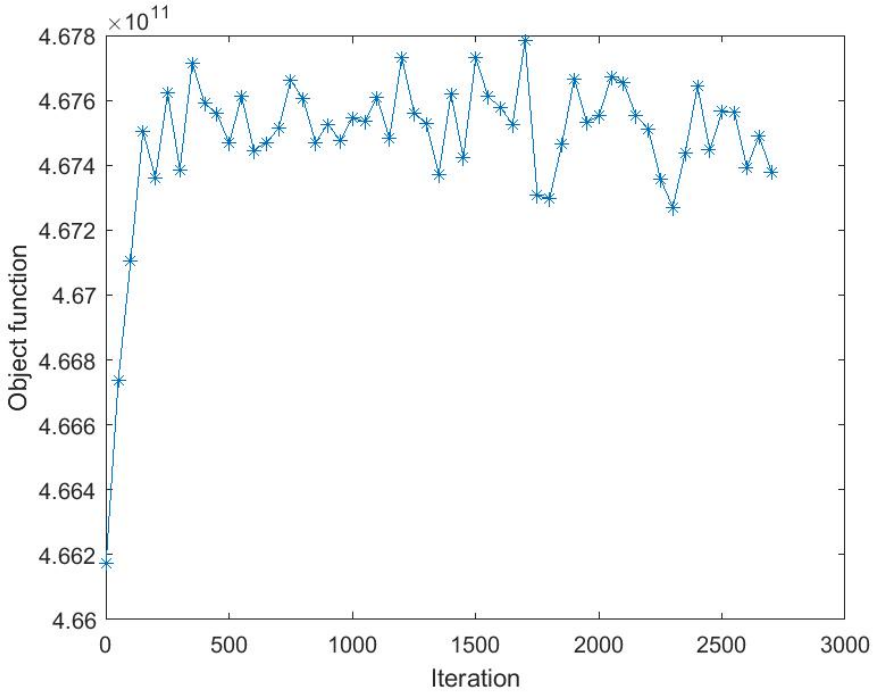


Figure 5.19: The error of the model result as a function of the Markov Chain

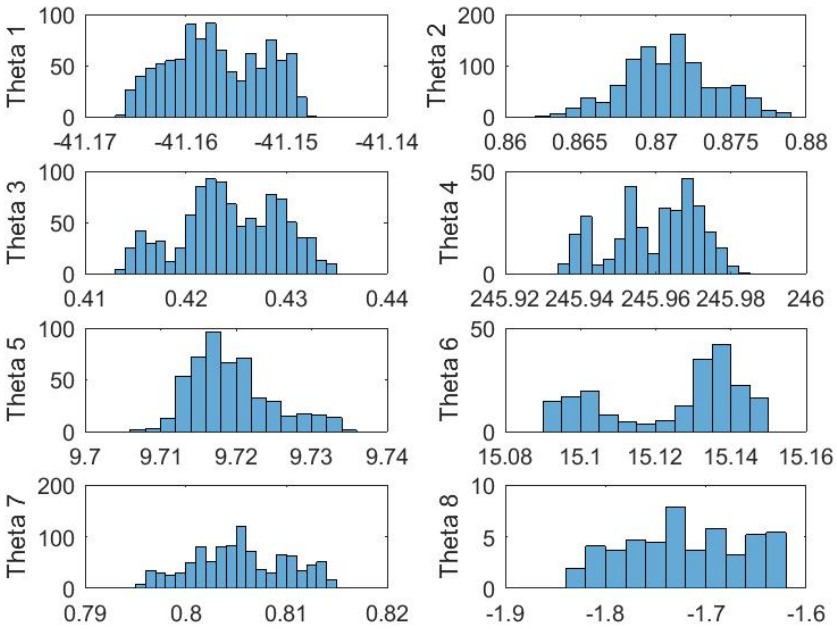
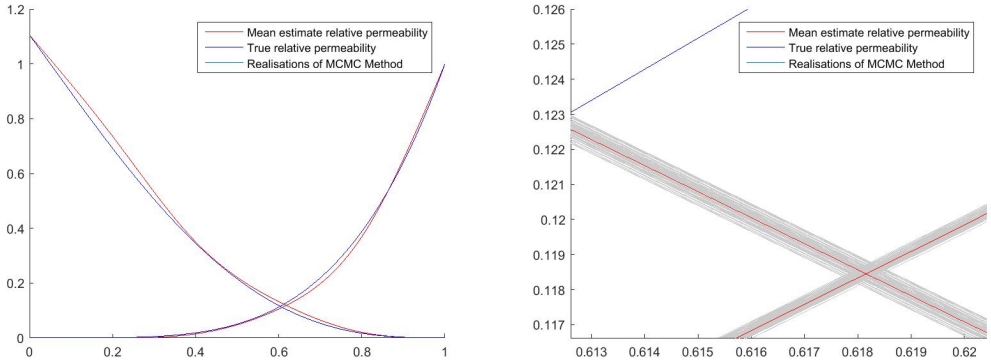


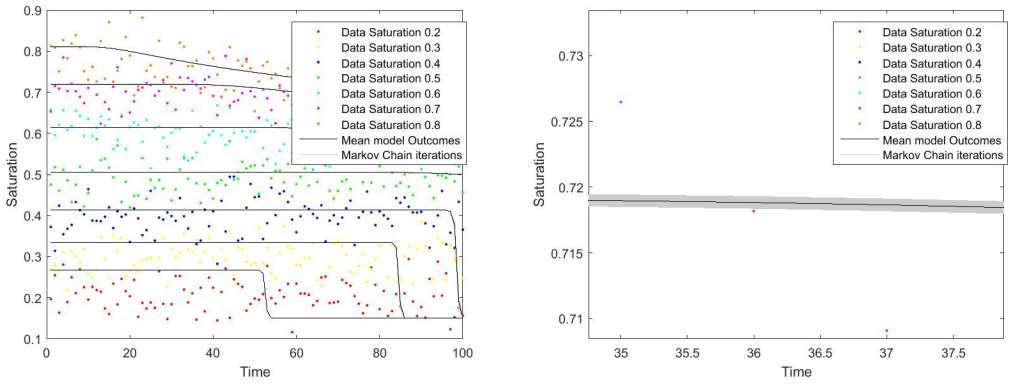
Figure 5.20: Histograms of the parameters for the MCMC method estimates for the B-spline with 8 degrees of freedom in total



(a) Oil relative permeability curves

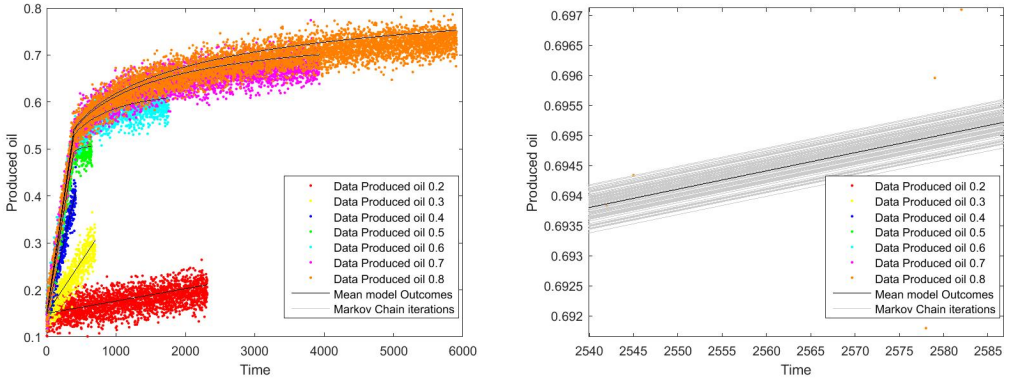
(b) Zoomed oil relative permeability curves

Figure 5.21: The relative permeability curves for the B-spline with 8 degrees of freedom with a hundred realisations from the MCMC method



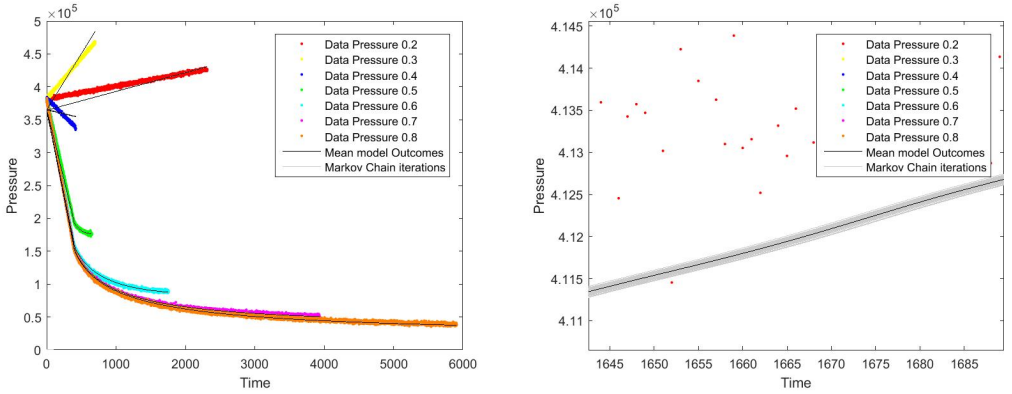
(a) Produced oil signal (b) Zoomed produced oil signal

Figure 5.22: The model outcome for the relative permeability curves for the B-spline with 8 degrees of freedom with a hundred realisations from the MCMC method



(a) saturation profile (b) Zoomed saturation profile

Figure 5.23: The model outcome for the relative permeability curves for the B-spline with 8 degrees of freedom with a hundred realisations from the MCMC method



(a) Pressure model outcome (b) Zoomed pressure model outcome

Figure 5.24: The model outcome for relative permeability curves for the B-spline with 8 degrees of freedom with a hundred realisations from the MCMC method

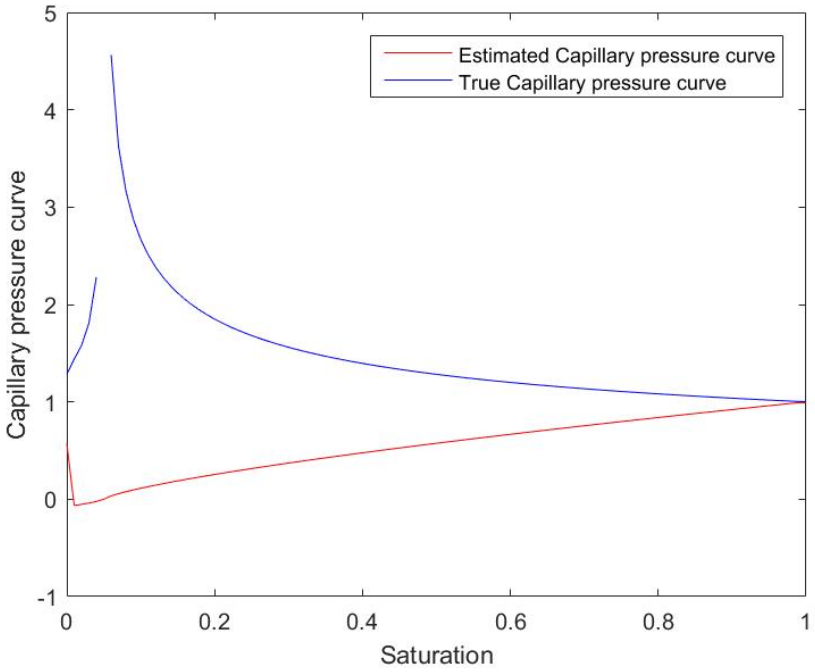


Figure 5.25: Capillary pressure curves

For this experiment the capillary pressure is modeld as

$$P_c(S_o) = \left(\frac{1 - S_{or}}{S_o - S_{or}}\right)^{\theta}. \tag{5.7}$$

Again the core is flooded with different saturations, $0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8$. And the measured data is, the pressure drop, the saturation profile when the steady state is reached and the amount of produced oil.
The resulting caplillary pressure curves are given in figure 5.25. This estimate is of very low quality since the true value was $\theta = \frac{1}{3}$. Which is of no surprise when taking a look at the object function given in figure 5.26. The object function has no minimum at the real value $\theta = \frac{1}{3}$.
The next question is, why is there no minimum at $\theta = \frac{1}{3}$? It is possible that there is to much noise, so the true minimum is drowned out by the measurement errors. To see whether this could be the reason for the bad estimate, the twin experiment is done, only without adding measurement errors to the true data. Indeed adapted the experiment returns $\theta = 0.3333$. The object function that was returned was of the order of 10^{-10}. going back to the original experiment, the situation is not very unlikely. The model outcome of the experiment is given in figures 5.27 and 5.28. Note that indeed the the model outcomes seem to go through the data clouds quite nicely. So there is indeed no reason for the optimisation to expect this result to be incorrect. So for this exact experiment, the calibration is unable to find the parameter value.
Another interesting artefact about the object function having no minimum had at the true parameter value is that the object function goes below what should be its minimal value. Since the parametrisation used is the one used in the model, the true value should give the minimum for the object function. That the object function hits values below this true minimum is a numerical artefact.

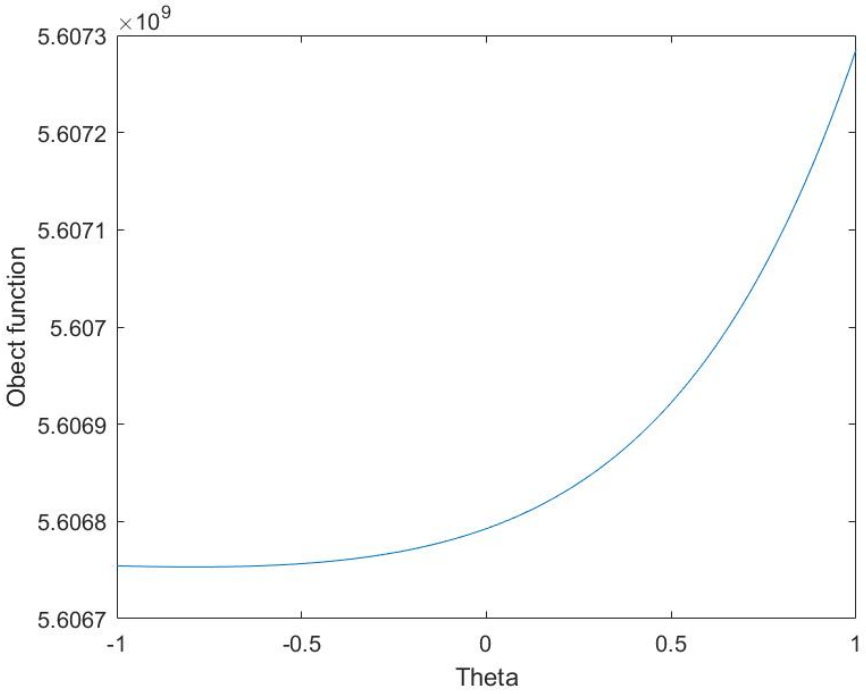


Figure 5.26: Object function capillary pressure curves

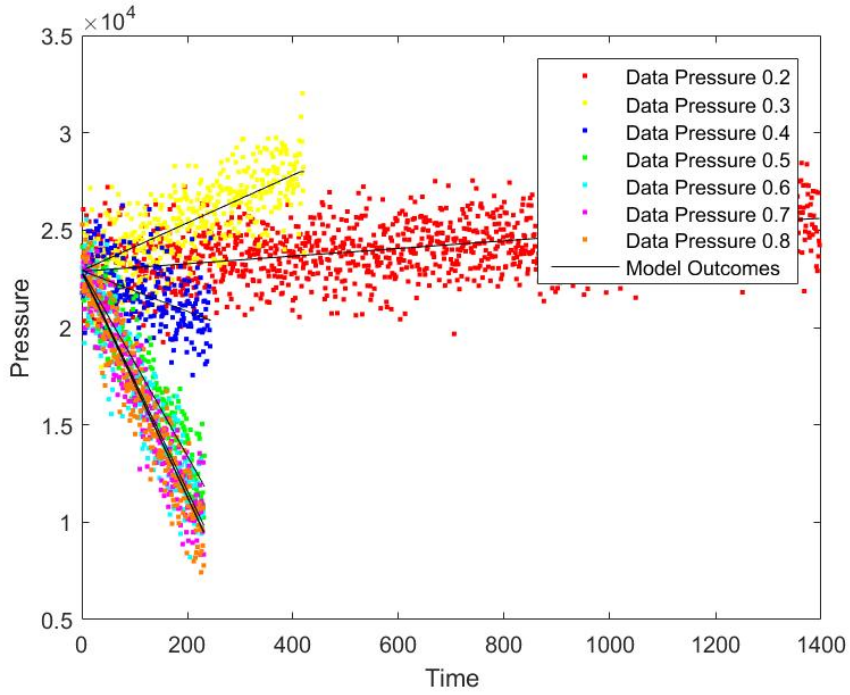


Figure 5.27: Capillary pressure curves

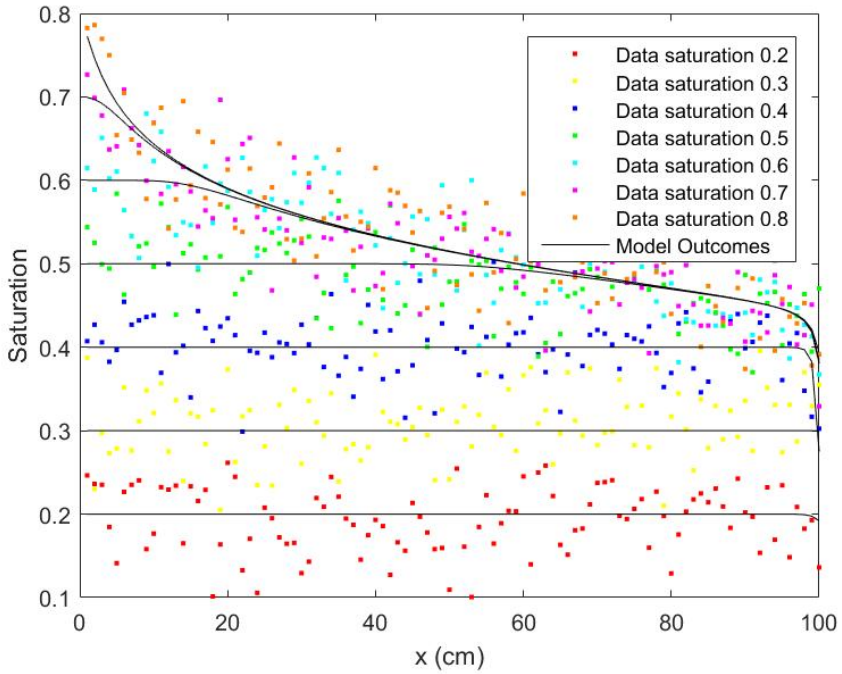


Figure 5.28: Object function capillary pressure curves

6 Conclusions

The different assessed parametrisations gave a wide variety of results. As mentioned before, it is not straight forward to quantify the quality of a parametrisation for the relative permeability curve. Even when the curve itself is very close to the true curve, the resulting model outcome may be way off the data. Even so the B-spline parametrisation is the most flexible of the parametrisations reviewed for this project. Therefore it is more suitable than the other parametrisations mentioned in this report. There are however many other parametrisations that allow for the same flexibility and have the correct characteristics to be suitable as relative permeability curves. Through the first experiment, it has become clear that the error made by choosing a certain parametrisation is far greater than the error made because the exact state cannot be measured. Even for the 1 dimensional case in figure 5.17b, the uncertainty retrieved from the MCMC was far smaller than the actual error made. So for the relative permeability curves the error made by choosing a unqualified parametrisation is easily greater than the error made in the estimate due to measurement errors. The experiments in chapter 5 show that is very possible to model the relative permeability curve with a relative small amount of degrees of freedom. What matters far more than the amount of free parameters, is the choice in parametrisation.
Modelling the uncertainty can be done for, a variety of parametrisations. The MCMC method will give an estimate of the full distribution, but it may take very long to get there. Especially when the initial estimate is not very good, the convergence may take thousands and thousands of model realisations. And even when the Markov Chain has converged, it takes a lot of iterations of the Markov Chain to map out a high dimensional distribution.
The experiment done in section 5.3, gives to little information to estimate the capillary pressure curve. Since the simplest parametrisation is used with only one degree of freedom, it means that this experiment really gives no information that helps retrieve the parameter.

6.1. Discussion

In this section some comments are made on the experiments done in chapter 5. Each parametrisation has a trade off between number of degrees of freedom and certainty of the estimate. The Corey curves allow for very little freedom but the curve always agrees with the prior knowledge on the relative permeabilities in this model. A much more free parametrisation such as the B-splines can take more shapes. This is a valuable characteristic of a parametrisation in this case. Because in reality the shape of the relative permeability curve is unknown and only some characteristics are known, the flexibility of the parametrisation is important. When the parametrisation is less flexible, the estimated error is more distorted. The lack of freedom in the parametrisation creates a false idea of certainty. It is important to realise that the choice in parametrisation will have a great impact on the estimated uncertainties.
When looking at the estimated uncertainties, there is one extra point of consideration to take into account. There are several methods to estimate the uncertainty of the estimate. The MCMC method maps the full distribution of the parameters, however, it will take a lot of computational power to do this. For the low dimensional cases, it appeared that the full distribution was very similar to a Gaussian. A method such as the EnKF can give a fit of the uncertainty under the assumption of Gaussian parameters, and is computationally far cheaper than the MCMC. The amount of extra work that goes into estimating the full distribution may not

outweigh the added precision, of estimating the full distribution. To give an example of how great this difference can be, the calibration done for 8 degrees of freedom takes 64 iterations of Newton. This means that the full model is evaluated 675 times. The evaluations can be to determine the gradient of the object function or to do the line search. For the MCMC for 8 dimensions the model was evaluated 3000 times. And for the MCMC the initial guess was the result of the calibration. So doing the MCMC on its own would result in many more iterations than that. Since because of the transformation, it is not very transparent to what initial guess is suitable, it is hard to find a good initial guess.
Another issue with using the MCMC is convergence. For the high dimensional case it will take a lot of iterations of the Markov Chain to converge. There are creteria to test if the Chain has converged, but in high dimensional cases these do not always agree[18]. This is problematic, because it is very hard to tell whether the Markov Chain has converged. There for it is hard to asses the quality of the estimate.

6.2. Future research

There is a multitude of directions for future research. Other parametrisations may be better for the relative permeability curves. Also, it would be interesting to see whether different data types influences the quality of the solution.
The amount of data that is used to estimate the parameters influences the certainty that can be obtained on the curves. It is interesting to seeing how this influence compares to the influence of the parametrisations. For instance, it is interested to compare the precision when the same set of water saturations is pumped through the core multiple times. This would give two realisations of every data point used in the standard experiment. Also, using more variation in water saturation should give an interesting comparison. Testing this using simulations is far cheaper than actually doing core experiments and see the combined result.
Another interesting experiment would be to estimate the relative permeability and the capillary pressure using one method. The current practise is often that the relative permeability is estimated first and then using the best estimate of the relative permeability curves, the capillary pressure is estimated using data from a shorter core experiment. This makes sense since the capillary pressure has little to no effect in experiments on longer cores. All of the information on the capillary pressure is situated at the outflow boundary. For a longer core experiment this is hardly distinguishable. For a short core, 6 centimetres for this project this effect is visible. However, the uncertainty of the relative permeability is not taken into account in this approach. This leads to a gross underestimate of the true uncertainty of the capillary pressure curve.
To asses the effect of the lost uncertainty, the capillary pressure curve that is estimated with a set relative permeability curves, is compared to the estimate of the capillary pressure curve that is estimated with uncertain relative permeability curve. So step one is to estimate the capillary pressure for the mean of the estimate of the capillary pressure curve. Then both curves are estimated simultaneously. These two estimates are then compared to objectively asses the effect of throwing away the known uncertainty of the relative permeability curve. The expected result is that both estimates have a greater uncertainty when being estimated simultaneously.

Bibliography

[1] UnderstandingtheMHAlgorithm.pdf, 1995.

[2] Unconstrained Nonlinear Optimization Algorithms - MATLAB & Simulink, 2016. URL http://www.mathworks.com/help/optim/ug/unconstrained-nonlinear-optimization-algorithms.html.

[3] Micah Altman, Jeff Gill, and Michael P McDonald. Numerical Issues in Statistical Computing for the Social Scientist. *Wiley Series in Probability and Statistics*, page 323, 2004. ISSN 0040-1706. doi: 10.1198/tech.2005.s270. URL http://www3.interscience.wiley.com/cgi-bin/bookhome/107063597.

[4] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian\nBayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, 2002. ISSN 1053587X. doi: 10.1109/78.978374. URL http://ieeexplore.ieee.org/xpl/login.jsp?tp={&}arnumber=978374{&}url=http://ieeexplore.ieee.org/iel5/78/21093/00978374.

[5] Emmanuel Blanchard. Parameter Estimation Method using an Extended Kalman Filter. *Proceedings of the Joint North America, Asia-Pacific ISTVS Conference and Annual Meeting of Japanese Society for Terramechanics Fairbanks, Alaska, USA, June 23-26, 2007*, 2007.

[6] Rh Brooks and At Corey. Hydraulic properties of porous media. *Hydrology Papers, Colorado State University*, 3(March):37 pp, 1964. URL http://www.citeulike.org/group/1336/article/711012.

[7] Se Buckley and Mc Leverett. Mechanism of fluid displacement in sands. *Trans. AIME*, 146(1337):107–116, 1942. ISSN 0081-1696. doi: 10.2118/942107-G. URL http://www.onepetro.org/mslib/app/Preview.do?paperNumber=SPE-942107-G{&}societyCode=SPE$\delimiter"026E3B2$nhttp://scholar.google.com/scholar?hl=en{&}btnG=Search{&}q=intitle:Mechanism+of+Fluid+Displacement+in+Sands{#}0.

[8] Daniela Calvetti and Erkki Somersalo. *Introduction to Bayesian scientific computing*, volume 2. 2007. ISBN 9780387733937. doi: 10.1007/978-0-387-73394-4. URL http://books.google.com/books?id=HIG0CRf6msoC.

[9] Shi Chen, Gaoming Li, Alvaro Marco Peres, and Albert Coburn Reynolds. A Well Test for In-Situ Determination of Relative-Permeability Curves. *SPE Annual Technical Conference and Exhibition*, 2005. ISSN 1094-6470. doi: 10.2118/96414-MS. URL http://www.onepetro.org/doi/10.2118/96414-MS.

[10] CIA. The World Factbook — Central Intelligence Agency, 2016. URL https://www.cia.gov/library/publications/the-world-factbook/geos/nl.html.

[11] K. H. Coats. IMPES Stability: The CFL Limit. *SPE Journal*, 8(3):291–297, 2003. ISSN 1086-055X. doi: 10.2118/85956-PA.

[12] Petros Dellaportas and Gareth O Roberts. Introduction to MCMC. *Lect. Notes Statistics*, 6(1):1–5, 2003. URL http://www.stat.umn.edu/{~}galin/Handbook/HandbookChapter1.pdf.

[13] A Doucet, S Godsill, and C Andrieu. On sequential {Monte Carlo} sampling methods for {Bayesian} filtering. *Statistics and Computing*, 10(3):197–208, 2000. ISSN 09603174. doi: 10.1023/A:1008935410038. URL {\T1\textless}GotoISI{\T1\textgreater}://WOS:000088440800002$\delimiter"026E3B2$nhttp://link.springer.com/content/pdf/10.1023/A:1008935410038.

[14] Geir Evensen. The Ensemble Kalman Filter: Theoretical formulation and practical implementation. *Ocean Dynamics*, 53(4):343–367, 2003. ISSN 16167341. doi: 10.1007/s10236-003-0036-9.

[15] Geir Evensen. *Data Assimilation, The Ensemble Kalman Filter*. 2007. ISBN 9783540383000. doi: 10.1007/978-3-540-38301-7.

[16] Geir Evensen and Peter Jan van Leeuwen. An ensemble Kalman smoother for nonlinear dynamics. *Monthly Weather Review*, 128(6):1852–1867, 2000. ISSN 0027-0644. doi: 10.1175/1520-0493(2000)128<1852:AEKSFN>2.0.CO;2.

[17] R. Fletcher. A new aproach to variable metric algorithm. *The Computer Journal*, 13(3):317–322, 1970. ISSN 0010-4620. doi: 10.1093/comjnl/13.3.317.

[18] Jeff Gill. Is partial-dimension convergence a problem for inferences from MCMC algorithms? *Political Analysis*, 16(2):153–178, 2008. ISSN 10471987. doi: 10.1093/pan/mpm019.

[19] N Kantas, a Doucet, S S Singh, and J M Maciejowski. *An Overview of Sequential Monte Carlo Methods for Parameter Estimation in General State-Space Models*, volume 44. IFAC, 2009. ISBN 9783902661470. doi: 10.3182/20090706-3-FR-2004.00129. URL http://publications.eng.cam.ac.uk/16156/.

[20] Jeffrey M. Lane and Richard F. Riesenfeld. A geometric proof for the variation diminishing property of B-spline approximation. *Journal of Approximation Theory*, 37(1):1–4, 1983. ISSN 10960430. doi: 10.1016/0021-9045(83)90111-9.

[21] Heng Li, Shengnan Chen, Daoyong Yang, and Paitoon Tontiwachwuthikul. Ensemble-Based Relative Permeability Estimation Using B-Spline Model. *Transport in Porous Media*, 85(3):703–721, 2010. ISSN 01693913. doi: 10.1007/s11242-010-9587-7.

[22] Jurgen H Schon. Physical Properties of Rocks: Fundamentals and Principles of Petrophysics - Juergen H. Schön - Google Boeken, 1996. URL https://books.google.nl/books?hl=nl{&}lr={&}id=uckHBgAAQBAJ{&}oi=fnd{&}pg=PP1{&}dq=physical+motivation+relative+permeability+curves{&}ots=R9-xQAoduh{&}sig=QX{_}e3wffCCyoPOinfV1UEXERgfw{#}v=onepage{&}q=relativepermeability{&}f=false.

[23] D. F. Shanno. Conditioning of quasi-Newton methods for function minimization. *Math. Comp*, 24(111):647, 1970. ISSN 0025-5718. doi: 10.1090/S0025-5718-1970-0274029-X.

[24] WILLIAM CARLISLE THACKER. The Role of the Hessian Matrix in Fitting Models to Measurements. *Journal of Geophysical Research*, 94(C5):6177, 1989. ISSN 0148-0227. doi: 10.1029/JC094iC05p06177.

[25] S Thomas. Enhanced oil recovery-an overview. *Oil & Gas Science and Technology-Revue ...*, 63(1):9–19, 2008. ISSN 12944475. doi: 10.2516/ogst. URL http://ogst.ifpenergiesnouvelles.fr/articles/ogst/abs/2008/01/ogst07042/ogst07042.html.

[26] G.N. Vanderplaats. *Numerical optimization*. 1995. ISBN 0387987932. doi: 10.1007/BF01068601. URL https://books.google.com/books?id=epc5fX0lqRIC{&}pgis=1.

[27] C Vuik, P Van Beek, F Vermolen, and J Van Kan. Numerieke Methoden voor Differentiaalvergelijkingen. page 148, 2004.

[28] Kv Yuen. A Relationship between the Hessian and Covariance Matrix for Gaussian Random Variables. *Bayesian Methods for Structural Dynamics and Civil Engineering*, pages 0–5, 2010. URL http://books.google.com/books?hl=en{&}lr={&}id=0iDSezuV9mIC{&}oi=fnd{&}pg=PR7{&}dq=Bayesian+Methods+for+Structu+r+al+Dynamics+and+Civil+Engineering{&}ots=TIu-U82QPi{&}sig=shjqB66m8MprJGu{_}AljfJ-KFhNM.