NTNU
Innovation and Creativity

Ensemble Kalman Filtering for State and Parameter Estimation on a Reservoir Model

John Petter Jensen

Master of Science in Engineering Cybernetics
Submission date: June 2007
Supervisor: Bjarne Anton Foss, ITK
Co-supervisor: Petter Tøndel, Sintef
John-Morten Godhavn, Statoil

Norwegian University of Science and Technology
Department of Engineering Cybernetics

Problem Description

Background:
Ensemble Kalman Filtering (EnKF) is a technique that has shown successful for history matching dynamic models with a high number of states, in particular distributed models such as metrological and ocean models, and more recently, reservoir models for oil and gas fields. An ensemble of typically 100 models is updated with field data. Both dynamic states (pressure, saturation, etc) and parameters (porosity, permeability) for a large number of cells are estimated. Some reservoir knowledge is built into the method by the selection of noise covariance matrices. The covariance used in the Kalman gain is estimated from the differences within the ensemble. The updated models may be used for reservoir surveillance, production and injection planning, etc.

In control theory this technique will be included in the term state and parameter estimation for a highly nonlinear and large scale system. Interesting aspects as observability (high number of states and parameters and small number of measurements with not too good accuracy) and constraint handling then arise. These aspects have not been investigated before to our knowledge. An updated dynamic model can be used for short term planning and control of production and injection. Events like production choke failures, changing processing capacity, changing receiving pressure (topside), water break through, sand production, can be responded to by feedback control with an accurate updated model and adequate instrumentation. The use of an commercial simulation like ECLIPSE in a nonlinear MPC raise challenges like robustness, computation time, discontinuities (these will be addressed in another MSc thesis by Patrick Meum). The long term goal is to develop a combined online solution, including both state and parameter estimation (history matching of reservoir model by EnKF) and nonlinear MPC for auto- or semiautomatic choking of production and injection wells using the updated reservoir model for predictions.

Tasks:
1) Literature review on EnKF and specifically on its application on reservoir models. Compare with other known estimation techniques for large scale systems.
2) Discuss observability of and constraint handling for EnKF for large scale systems
3) Implement and test method on simple ECLIPSE simulation model: Let one instant of the model represent the true reservoir and simulate this with a production and injection plan. Implement an EnKF and investigate how well the method works.
4) Sketch how EnKF can be combined with nonlinear model predictive control (MPC) in a combined estimation and control scheme for advanced control of wells and reservoir.

Assignment given: 09. January 2007
Supervisor: Bjarne Anton Foss, ITK

Preface

Oil production is a popular area for research these days. Due to high oil prices and demand a great deal of funding is available for research. Production optimization is important to enhance the recovery of oil from the reservoir. The technology that is put into the reservoirs allows for advanced control of the production. The basis for this control is models of the reservoir. Building and maintaining these models is probably one of the most challenging areas.

This field is both challenging and interesting, and it also ties many disciplines together. That is why I chose to work with continuous model updating. EnKF is a new method, which is not that well established compared to other methods. This is both a risk and a challenging task, but in the end it was a rewarding experience. After looking back at this assignment, I feel that the task was a little ambitious. Getting into the history matching, the statistics, the EnKF method, the implementation and reservoir modelling is a lot to cover. This resulted in a little shortage of time to work with the simulations and results. In the end I think I managed to tie the ends together and learn a lot about the EnKF and its use for history matching. In the paper a lot of interesting pointers are given and the EnKF is in general presented from a little different angle. This is since my background is in control engineering, while the authors of most EnKF material has a mathematical background. I am satisfied with how I brought the control engineering terms and "way of thinking" into the EnKF methodology.

Acknowledgments

This work could not have been done without the help of others. I would like to thank my advisers at Statoil, John-Morten Godhavn, Petter Tøndel and Per Arne Slotte. Also my professor Bjarne A. Foss has assisted me through the whole process and given me many good inputs. I really appreciate the help i received from G. Nævdal at IRIS. He helped me getting into grips with the implementation and we had many interesting discussions on the topic. I would also like to thank IRIS for letting me use their EnKF implementation and Statoil for providing me with resources needed to complete the work. Finally i would like to than my fellow classmates Patrick Meum and Håvard Torpe and the rest of the people at the process control department at the Statoil research center. They have all contributed to this paper and I have been lucky to work in such a good environment.

Abstract

In reservoir management it is important with reservoir models that have good predictive abilities. Since the models initially are based on measurements with high uncertainties it is important to utilize new available data. Ensemble Kalman Filter (EnKF) is a new method for history matching that has received a lot of attention the last couple of years. This method is sequential and continuously update the reservoir model states (saturations, pressures etc.) and parameters (permeabilities, porosities etc) as data become available.

The EnKF algorithm is derived and presented with a different notation, similar to that of the Kalman Filter (KF) used in control engineering. This algorithm is also verified on a simple linear example to illustrate that the covariance of the EnKF approaches that of the linear KF in case of an infinite ensemble size.

In control theory this method falls under the category of parameter and state estimation of nonlinear large scale systems. Interesting aspects as observability and constraint handling arises, and these are linked to the EnKF and the reservoir case. To determine if the total problem is observable is a nearly impossible task, but one can learn a lot from introducing this concept.

The EnKF algorithm was implemented on a simple "shoe box" reservoir model and four different problem initializations were tested. Although decent results were achieved from some of the simulations other failed completely. Some strange development in the ensemble when little information is available in the measurements was experienced and discussed.

An outline was presented for a reservoir management scheme where EnKF is combined with Model Predictive Control (MPC). Some challenges was pointed out and these involve computation time, predictive ability, closed-loop behavior etc.

Contents

List of Figures

Chapter 1

Introduction

Reservoir management is important to increase overall profitability. Models of the reservoirs are important assets in reservoir management, but such models are only valuable if they have good predictive abilities. Before the production starts the models are based on seismic data, well tests etc and there are high uncertainties associated with this data. In general the predictive value of such models is limited and tends to deteriorate over time. To prevent this some form of model updating have to be done at regular intervals. The update consider the new production data from the system to update the model. This procedure is known *History Matching* and is a well established field in reservoir engineering. At the beginning this procedure was done manually, but as this was quite time consuming a lot of research is done to automate this procedure. History matching is a difficult task, with models containing many parameters subject to high uncertainty and with few measurements available. Typically the new commercial reservoir models contains thousands of cells, all consisting of a number of dynamic and static parameters. Therefore there is a demand for automatic methods that are computationally efficient. A new method for History Matching is the Ensemble Kalman Filter (EnKF). This method was first introduced by G. Evensen in 1994 [18] for use on oceanic models, while the first history matching application was given by G. Nævdal et al. in 2002 [36]. This method has shown promising results on several history matching cases, is computational efficient compared to more traditional methods and computes sequentially. The last property provides continuous model update, meaning it uses all production data as it becomes available to update the model and tries to improve the models predictive abilities. All these qualities make EnKF suitable for integration with reservoir management.

In recent years there has been an interest in closed-loop reservoir management. The closed-loop is used with optimal reservoir control. A requirement for optimal control, is a model with predictive abilities. This is especially important for long-term reservoir management. Some work has been done to incorporate continuous model updating in this closed loop scheme. Still there are still many challenges related to coupling these two methods into a complete automatic reservoir management system. Such procedures are however well established in control engineering theory. Interesting aspects in this matter is observability and constraint handling. A lot of the closed-loop system analysis is however only performed on smaller systems. This paper seeks to address these terms in a reservoir management perspective.

Next the layout of this paper will be presented. In Chapter 2 some statistic fundamentals are described. This acts as a basis for understanding the filter theory presented in Chapter 3. In Chapter 3 the EnKF algorithm will be presented with a basis in the linear Kalman Filter (KF) and the nonlinear Extended Kalman Filter (EKF). This algorithm is the basis for the further work done with the EnKF. Chapter 4 outline a property of the EnKF measurement perturbation scheme, using a simple linear system and a comparison with the linear KF. These Chapter (2-4) considers the EnKF as a method independent of application. In the rest of the report the EnKF is considered for reservoir model updating. In Chapter 5 an outline of the previous work done with EnKF on continuous reservoir model updating is given. Further the methodology for performing a history match using the EnKF is described. To put the EnKF in some perspective a comparison with traditional methods for history matching is presented at the end of this chapter. In Chapter 6, observability and constraint handling is outlined and discussed with basis in the EnKF. Chapter 7 presents the implementation of EnKF for history matching on a simple "Shoe box" reservoir model. Chapter 8 presents an outline of a closed-loop scheme combining both Model Predictive Control (MPC) and EnKF for optimal reservoir management with continuous model update. To conclude the report the conclusions is presented in Chapter 9 and an outline of the further work in Chapter 10. At the end one can find the references and various appendices.

Chapter 2

Statistic Fundamentals

To obtain an understanding of the Ensemble Kalman Filter (EnKF) it is important to understand some basic statistic principles. The statistics given in this chapter is a short overview of some of the fundamentals, aimed at providing the reader with a basis for understanding the filter theory. First some basic terms in statistics is presented. After that an introduction to the Gaussian probability distribution is given. Afterward a Monte Carlo approximation to distributions using samples is outlined. The latter forms the basic for introducing the EnKF. For a more detailed description of the statistics the reader should refer to the three books that this introduction is based on. First the work by A. H. Jazwinski[29] is a book explaining principles of stochastic processes and filtering theory. R. G. Brown and P. Y. C. Hwang[8] provides an introduction to random signals and applied Kalman Filtering. Last G. Evensen [16] presents some statistics in his introduction to data assimilation using the EnKF. In addition some references on Monte Carlo statistical methods is referred to later in this chapter.

2.1 Random variables

A real random variable is a real finite-valued function $X(\cdot)$ defined on Ω if, for every real number, the inequality

$$X(\omega) \leq x \tag{2.1}$$

defines a set ω whose probability is defined. The function

$$F_X(x) \stackrel{\triangle}{=} Pr\{X(\omega) \leq x\} \tag{2.2}$$

is called the cumulative distribution function (CDF).

2.2 Probability density function

The probability density function (PDF) states the probability that a random variable X will take a particular value x. The PDF $f_X(x)$ must satisfy (2.3a) and (2.3b).

$$f_X(x) \geq 0 \quad \forall\ x \tag{2.3a}$$

$$\int_{-\infty}^{\infty} f_X(x)dx = 1 \tag{2.3b}$$

2.3 Expected value, Variance and Covariance

Expected value

The expected value for a random variable X is given both in the discrete and the continuous case by

$$\begin{array}{ll} \textit{Discrete} & \textit{Continuous} \\ E[X] = \sum_{i=1}^{n} p_i x_i & E[X] = \int_{-\infty}^{\infty} x f_X(x) \end{array} \tag{2.4}$$

where $f_X(x)$ is the PDF for the continuous case and p_i is the probability of $X = x_i$ for the discrete case.

Variance

The variance of a random variable is given in (2.5).

$$VAR(X) = E\left[(X - E[X])^2\right] = E[X^2] - E[X]^2 \tag{2.5}$$

In a qualitative sence, the variance of Xi is a measure of the dispersion of X about its mean. Another property of the random variable X is the standard deviation (σ), which is defined as the square root of the variance.

Covariance

The covariance is defined in (2.6) for two random variables X and Y.

$$COV[X, Y] = E\left[(X - E[X])(Y - E[Y])\right] \tag{2.6}$$

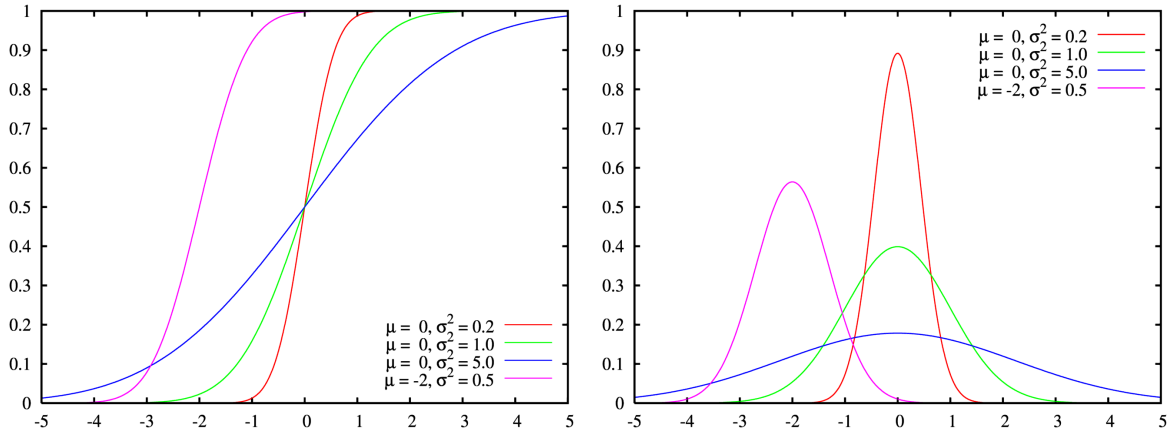
Here the joint probability function can be defined as $f(X, Y)$. If X and Y are independent $f(X, Y) = f(X)f(Y)$ and accordingly $COV[X, Y] = 0$. Qualitatively the covariance describe the dependency between the two random variables X and Y.

2.4 Probability distributions

Random variables have a certain distribution given by their PDF. There exist many distributions for random variables. the most common is the normal or Gaussian distribution. The Gaussian distribution PDF is given by

$$f_X(x) = \frac{1}{\sqrt{2\pi}\sigma}\left[-\frac{1}{2\sigma^2}(x - E[X])^2\right] \tag{2.7}$$

The normal distribution has a distribution function and PDF as illustrated in Figure 2.1 Here the CDF and PDF is plotted for different standard deviations σ given a Gaussian distribution.
$X \sim N(E[X], \sigma^2)$ is a short note for telling that a random variable X is normal distributed with an expected value of $E[X]$ and standard deviation σ. If $\boldsymbol{X}$ is a vector of Gaussian distributed random variables with mean $E[\boldsymbol{X}]$ and a covariance matrix $\boldsymbol{Q}$ then this can be denoted
$\boldsymbol{X} \sim N(E[\boldsymbol{X}], \boldsymbol{Q})$. The diagonal elements of Q denotes the variance for each of the random variables in $\boldsymbol{X}$ and the off-diagonal elements represent the covariance between the variables.



(a) Cumulative distribution function (b) Probability density function

Figure 2.1: CDF and PDF for a Gaussian distribution

2.5 Approximations using samples from a distribution

In this section an approximation to the expected value, variance and covariance is proposed. This approximation is based on using an ensemble of samples to describe a probability distribution. The theory used is based on the description given in J. S. Liu [31] and C. P. Robert and G. Casella [38]. First a description of an integral approximation using a Markov Chain Monte Carlo method is used. Afterward the sample approximation for the expected value, variance and covariance is outlined.

Markov Chain Monte Carlo method

To understand the need for using approximations to derive the expected value, variance and covariance an integral is observed. Solving the integral given in 2.8 is an essential part of many scientific problems.

$$I = \int_D g(\boldsymbol{x}) d\boldsymbol{x} \tag{2.8}$$

Here D is often a region defined in a high-dimensional space and $g(\boldsymbol{x})$ is the function of interest. Solving this integral using numerical integration can become very complex at dimensions higher than 3-4. Consider a case with 5 dimensions where there is a need for 10 grid points in each direction to have a proper representation of the function. To solve this a grid with 10^5 points would have to be kept for reference. Storing this alone would require a lot of hard drive space and to compute the problem 10^5 grid points would need to be evaluated.

To cope with problems of this magnitude an alternative calculation of the integral is available. Assume that N independent and identically distributed (i.i.d.) random samples $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}$ are drawn uniformly from D. Then the following approximation to I can be obtained

$$\hat{I}_N = \frac{1}{N}\left[g(\boldsymbol{x}^{(1)}) + \ldots + g(\boldsymbol{x}^{(N)})\right] \tag{2.9}$$

The average of many independent random variables with common mean and finite variances tends to stabilize at their common mean. This translates into

$$\lim_{N\to\infty} \hat{I}_N = I, \text{ with probability 1.} \tag{2.10}$$

The convergence rate is defined from the central limit theorem as follows

$$\sqrt{N}(\hat{I}_N - I) \rightarrow N(0, \sigma^2) \tag{2.11}$$

where $\sigma^2 = VAR[g(\boldsymbol{x})]$. The "error term" of the Monte Carlo approximation is $O(N^{-1/2})$. Note that the error term is independent of the dimensionality of $\boldsymbol{x}$, but two intrinsic difficulties arise

- When the region D is large in high-dimensional space, the variance σ^2 can be formidably large
- One may not be able to produce uniform random samples in an arbitrary region D

To compensate for this the idea of importance sampling can be employed. More details on this can be found in Robert and Casella or Liu. The MCMC method proposed here can be used to approximate the expected value, the variance and the covariance for a given sample.

Expected Value

Given a sample of N independent realizations of $\boldsymbol{x}$ where $\boldsymbol{x}$ has dimension m and the distribution $f(\boldsymbol{x})$. The expected value can be approximated by

$$E[\boldsymbol{X}] \simeq \bar{\boldsymbol{x}} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i \tag{2.12}$$

which is the sample mean. The expected value is in other word the value that is the expected outcome if infinitely many data are present.

Sample Variance

The sample variance can in a similar way be approximated as shown in (2.13).

$$VAR[\boldsymbol{X}] = E\left[(\boldsymbol{X} - E[\boldsymbol{X}])^2\right] \simeq \overline{(\boldsymbol{x} - \bar{\boldsymbol{x}})^2} = \frac{1}{N-1}\sum_{i=1}^{N}(\boldsymbol{x}_i - \bar{\boldsymbol{x}})^2 \tag{2.13}$$

where $N-1$ is used rather than N to provide an unbiased estimator for the variance.

Sample covariance

The covariance is derived using the same properties as above and is given by

$$COV[\boldsymbol{X}, \boldsymbol{Y}] = E\left[(\boldsymbol{X} - E[\boldsymbol{X}])(\boldsymbol{Y} - E[\boldsymbol{Y}])\right] \tag{2.14}$$

$$\simeq \overline{(\boldsymbol{x} - \bar{\boldsymbol{x}})(\boldsymbol{y} - \bar{\boldsymbol{y}})} = \frac{1}{N-1}\sum_{i=1}^{N}(\boldsymbol{x}_i - \bar{\boldsymbol{x}})(\boldsymbol{y}_i - \bar{\boldsymbol{y}}) \tag{2.15}$$

This section has described the basic statistical definitions which forms a basis for the filter presented in this paper. The focus now moves to the Ensemble Kalman Filter and the structure of this filter will be presented in the next section.

Chapter 3

Theoretical formulation of the EnKF

The original Kalman Filter (KF) first presented by R. E. Kalman in 1960 [30] was designed for linear models. An early attempt to adapt this filter to nonlinear problems was done using the Extended Kalman Filter (EKF), which is based on linearization of the nonlinear model using the Jacobian. This is not suitable for large scale problems or problems that are too nonlinear. The Ensemble Kalman Filter (EnKF) was introduced by Evensen in 1994 [18] to handle large nonlinear oceanic models, and has had promising results in many areas.

To introduce the EnKF a linear estimation problem is outlined and the basic linear KF is presented as a solution. After the focus moves to a nonlinear estimation problem and the EKF is presented as a possibility for solving such cases. These filters however have shown some weaknesses on problems like the reservoir parameter estimation in this paper, but is excellent as an introduction for the parameter estimation problem. Last a problem description for a nonlinear problem is outlined and an EnKF algorithm is proposed as a solution to the problem.

3.1 Linear Kalman Filter

The Kalman Filter (KF) is a good basis for understanding the EnKF. A good description of the Kalman filter is given in the book Brown and Hwang [8].

The problem

Consider a discrete stochastic system with the true state $\boldsymbol{x}_k$ at time k represented by a linear dynamical model according to (3.1).

$$\boldsymbol{x}_{k+1} = \boldsymbol{\Phi}\boldsymbol{x}_k + \boldsymbol{\Delta}\boldsymbol{u}_k + \boldsymbol{q}_k \tag{3.1a}$$

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{v}_k \tag{3.1b}$$

Here $\boldsymbol{\Phi}$ is a linear model matrix, $\boldsymbol{y}_k$ are the measurements at time k and $\boldsymbol{C}$ is the linear measurement matrix. $\boldsymbol{\Delta}$ is the input matrix and $\boldsymbol{u}_k$ are the inputs at time k. The true initial state is given by $\boldsymbol{x}_0$ and it is normal distributed $N(0, \boldsymbol{P}_0)$, where $\boldsymbol{P}_0$ is the initial covariance matrix for the stochastic process $\boldsymbol{x}$. $\boldsymbol{q}_k$ is the unknown model error at time k and $\boldsymbol{v}_k$ is the measurement error at time k. These errors are assumed normal distributed as shown below

$$\boldsymbol{q}_k \sim N(0, \boldsymbol{Q}_k) \tag{3.2a}$$

$$\boldsymbol{v}_k \sim N(0, \boldsymbol{V}_k) \tag{3.2b}$$

where $\boldsymbol{Q}_k$ and $\boldsymbol{V}_k$ are the covariance matrices for respectively the model noise and the measurement noise.

The Kalman filter equations

The problem described in (3.1) is considered. Here $\boldsymbol{x}_k$ denotes the system state at time t_k and $\boldsymbol{x}_{k+1}$ the state at some time t_{k+1} where $t_{k+1} > t_k$. Let $\hat{\boldsymbol{x}}_{k+1}^-$ be the a priori estimate of the process state, based on the previous information available up until this point (t_{k+1}). Similarly let $\hat{\boldsymbol{x}}_k$ be the best estimate of the system state with filtered information up to time t_k. This can be written in mathematical terms as

$$\hat{\boldsymbol{x}}_{k+1}^- = E(\boldsymbol{x}_{k+1}|\boldsymbol{y}^*) \tag{3.3}$$

$$\hat{\boldsymbol{x}}_k = E(\boldsymbol{x}_k|\boldsymbol{y}^*) \tag{3.4}$$

where $\boldsymbol{y}*$ are all the available measurements up to time t_k. The model for a priori prediction of the states forward in time can be derived using (3.1):

$$\hat{\boldsymbol{x}}_{k+1}^- = \boldsymbol{\Phi}\hat{\boldsymbol{x}}_k + \boldsymbol{\Delta}\boldsymbol{u}_k \tag{3.5}$$

The a priori covariance matrix is updated using the following matrix relationship

$$\boldsymbol{P}_{k+1}^- = \boldsymbol{\Phi}\boldsymbol{P}_k\boldsymbol{\Phi}^T + \boldsymbol{Q}_k \tag{3.6}$$

where

$$\boldsymbol{P}_{k+1}^- = E\left[(\hat{\boldsymbol{x}}_{k+1}^- - \boldsymbol{x}_{k+1})(\hat{\boldsymbol{x}}_{k+1}^- - \boldsymbol{x}_{k+1})^T|y*\right] \tag{3.7}$$

$$\boldsymbol{P}_k = E\left[(\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k)(\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k)^T|y*\right] \tag{3.8}$$

The filtering step is where the a posteriori estimates are calculated. The a posteriori state is updated using the following equation

$$\hat{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{C}\boldsymbol{x}_k^-) \tag{3.9}$$

where the Kalman gain K_k is given by

$$\boldsymbol{K}_k = \boldsymbol{P}_k^-\boldsymbol{C}^T(\boldsymbol{C}\boldsymbol{P}_k^-\boldsymbol{C}^T + \boldsymbol{V}_k)^{-1} \tag{3.10}$$

The a posteriori covariance matrices are updated as follows

$$\boldsymbol{P}_k = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{C})\boldsymbol{P}_k^- \tag{3.11}$$

The filter can the be summarized in Algorithm 1.

input: $\hat{\boldsymbol{x}}_0$, $\boldsymbol{P}_0$, $\boldsymbol{V}_k$, $\boldsymbol{Q}_k$

$k = 0$

while *true* **do**

Prediction step

$\hat{\boldsymbol{x}}_{k+1}^- \leftarrow \boldsymbol{\Phi}\hat{\boldsymbol{x}}_k + \boldsymbol{\Delta}\boldsymbol{u}_k$ // Forecast the state ahead

$\boldsymbol{P}_{k+1}^- \leftarrow \boldsymbol{\Phi}\boldsymbol{P}_k\boldsymbol{\Phi}^T + \boldsymbol{Q}_k$ // Forecast the covariance matrix ahead

Filtering step

$\boldsymbol{K}_k \leftarrow \boldsymbol{P}_k^-\boldsymbol{C}^T(\boldsymbol{C}\boldsymbol{P}_k^-\boldsymbol{C}^T + \boldsymbol{V}_k)^{-1}$ // Compute the kalman gain

$\hat{\boldsymbol{x}}_k \leftarrow \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{C}\boldsymbol{x}_k^-)$ // Update state estimate with measurement

$\boldsymbol{P}_k \leftarrow (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{C})\boldsymbol{P}_k^-$ // Update the covariance matrix

$k \leftarrow k + 1$

end

Algorithm 1: Basic Linear Kalman Filter (KF)

Here the two steps of prediction and filtering are illustrated. For the Kalman filter the initial estimate for the state ($\hat{\boldsymbol{x}}_0$) and the state covariance ($\boldsymbol{P}_0$) is needed together with the model error and measurement covariance (respectively $\boldsymbol{Q}_k$ and $\boldsymbol{V}_k$). These variables are used to tune the Kalman filter response. If the initial state and/or the state covariance is close to its true value then the settling time will naturally be shorter. Both the measurement and the model error covariance tuning adjust the Kalman gain $\boldsymbol{K}_k$. When the measurement covariance is large this suggests that there is a lot of error in the measurement. As a result the Kalman gain will be smaller and the observed effect of the analysis step will decrease. If the model error covariance is large the Kalman gain will increase and push toward a stronger influence from the analysis step. These two tuning parameters have to be considered together and in the end it is the ratio between them that matter. More details about the tuning and examples can be found in Brown and Hwang[8].

3.2 Extended Kalman Filter

The Extended Kalman Filter (EKF) may be applied to nonlinear dynamics. A friendly exposition of the EKF can be found both the book by Jazwinski [29] and the book by Brown and Hwang [8]. This filter uses an approximate linearized equation for prediction of the error statistics. The nonlinear system is defined by

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k) + \boldsymbol{q}_k \tag{3.12a}$$

$$\boldsymbol{y}_k = \boldsymbol{g}(\boldsymbol{x}_k) + \boldsymbol{v}_k \tag{3.12b}$$

The forecast and analysis steps are the same as for the linear Kalman filter in Algorithm 1. The only difference is that the model and measurement matrices are linearized in the analysis part at each step from the system given in (3.12). The linearized model and measurement matrices are shown in Equation 3.13.

$$\boldsymbol{\Phi}_k = \left.\frac{\partial \boldsymbol{f}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}_k} \tag{3.13a}$$

$$\boldsymbol{C}_k = \left.\frac{\partial \boldsymbol{g}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}_k} \tag{3.13b}$$

input: $\hat{\boldsymbol{x}}_0$, $\boldsymbol{P}_0$, $\boldsymbol{V}_k$, $\boldsymbol{Q}_k$

$k = 0$
while *true* **do**
Prediction step
$\hat{\boldsymbol{x}}^-_{k+1} \leftarrow \boldsymbol{f}(\hat{\boldsymbol{x}}_k)$ // Forecast the state ahead
$\boldsymbol{P}^-_{k+1} \leftarrow \boldsymbol{\Phi}_k \boldsymbol{P}_k \boldsymbol{\Phi}_k^T + \boldsymbol{Q}_k$ // Forecast the covariance matrix ahead
Filtering step
$\boldsymbol{K}_k \leftarrow \boldsymbol{P}_k^- \boldsymbol{C}_k^T (\boldsymbol{C}_k \boldsymbol{P}_k^- \boldsymbol{C}_k^T + \boldsymbol{V}_k)^{-1}$ // Compute the kalman gain
$\hat{\boldsymbol{x}}_k \leftarrow \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k(\boldsymbol{y}_k - \boldsymbol{C}_k \boldsymbol{x}_k^-)$ // Update state estimate with measurement
$\boldsymbol{P}_k \leftarrow (\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{C}_k)\boldsymbol{P}_k^-$ // Update the covariance matrix
$k \leftarrow k + 1$
end

Algorithm 2: Extended Kalman Filter (EKF)

The result of this is that the system matrices used for updating the covariances and the gain are Linear Time Variant (LTV). The EKF is presented in Algorithm 2. The EKF presents an approximate equation for the error covariance evolution. Thus the properties of the model dynamics will strongly influence the suitability of the EKF.

There are some problems regarding the usage of the KF and EKF with high dimensional and nonlinear dynamics. Using a KF or EKF on high dimensional problems poses some demands of storage and computation time. Given a model with m unknowns in the state vector, then the error covariance matrix will have m^2 unknowns. The update of the error covariance matrix according to (3.6) requires the cost of $2m$ model integrations. As a result the KF and EKF is not very suitable for high-dimensional systems. Another issue is the one mentioned earlier regarding the linearization done in the EKF. The linearization leads to poor error covariance updates and in some cases unstable growth. To cope with this higher order approximations may be used, but this leads to a higher storage requirement and more calculation time. In general there is a need for a better way to update the covariance equation when working with nonlinear, high-dimensional problems.

3.3 Ensemble Kalman Filter

The idea behind the EnKF is to provide a filter that is suitable for large-scale nonlinear systems. The basic and extended Kalman filter have proven difficult to adapt to such systems due to computation time and handling of nonlinear dynamics. Often the difficulty lies in computing the error covariance matrix. In the EnKF the covariance matrix estimate ($\boldsymbol{P}$) is predicted and analyzed using the ensemble statistics.

The EnKF was introduced by G. Evensen in 1994 [18] for the purpose of handling large-scale nonlinear ocean models. The EnKF has been developed and examined further for various applications in many papers. A good source for articles and developments of the EnKF can be found online [2]. This page is established as a reference page for users of the EnKF made by G. Evensen and Nansen Environmental and Remote Sensing Center (NERSC). At this page one can also find a Fortran 90 EnKF implementation and some examples.

The method was originally used with oceanic forecasting models and, in addition to the work of Evensen, the ocean model described by K. Brusdal et. al. [9] has also showed good potential. Another application where EnKF has showed promising results is the marine ecosystem case presented by both M. Eknes and G. Evensen [15] and J. I. Allen et. al. [4]. Lately the properties of the EnKF

has showed promising results in oil reservoir modelling. The latter application is the focus of this paper and will be discussed in Chapter 5. The theoretical formulation of the EnKF will be described in this section. A more detailed exposition of the EnKF can be found in the book by Evensen [16].

3.3.1 Problem formulation

The EnKF is usually considered for large scale problems. These problems is often arranged in a grid, where each grid block contains a number of dynamic and static variables. It is then appropriate to define these variables as a function of space and time. The model state can then be defined as $\boldsymbol{x}(\boldsymbol{z}, t_k) \in \mathcal{R}^{n_x}$ and consists of n_x dynamic variables at each location $\boldsymbol{z}$ in space and at time t_k. Some of the static variables are usually poorly known. Let these poorly know parameters be denoted $\boldsymbol{\theta}(\boldsymbol{z}) \in \mathcal{R}^{n_\theta}$ consisting of n_θ static variables and is defined in space as the dynamic parameters. Consider the following problem

$$\boldsymbol{x}(\boldsymbol{z}, t_{k+1}) = \boldsymbol{f}(\boldsymbol{x}(\boldsymbol{z}, t_k), \boldsymbol{\theta}(\boldsymbol{z})) + \boldsymbol{q}(\boldsymbol{z}, t_k) \tag{3.14a}$$

$$\boldsymbol{\theta} = \boldsymbol{\theta}_0(\boldsymbol{z}) + \boldsymbol{\theta}'(\boldsymbol{z}) \tag{3.14b}$$

$$\boldsymbol{y} = g(\boldsymbol{x}(\boldsymbol{z}, t_k), \boldsymbol{\theta}(\boldsymbol{z})) + \boldsymbol{v}(t_k) \tag{3.14c}$$

where $\boldsymbol{f}(\boldsymbol{x}(\boldsymbol{z}, t_k), \boldsymbol{\theta}(\boldsymbol{z}))$ is the nonlinear model operator and $\boldsymbol{q}(\boldsymbol{z}, t)$ is the model noise. The model state is initially given as $\boldsymbol{x}(\boldsymbol{z}, t_0)$. An initial guess of the poorly known parameters are given by $\boldsymbol{\theta}_0(\boldsymbol{x}) \in \mathcal{R}^{n_\theta}$. The measurements $\boldsymbol{y} \in \mathcal{R}^{n_y}$ may be direct point measurements of the solution or complex parameters nonlinearly related to the model state.

The error term $\boldsymbol{q}(\boldsymbol{z}, t_k)$, which is included in (3.14), represent the model errors. Without these errors the system would be overly specified and have no solution. But introducing these errors implies that there are infinitely many solutions to the system. To cope with the infinite possibilities an assumption that the errors are normal distributed with expected values being zero and the covariances known.

3.3.2 Ensemble covariances

In general the error covariance matrices for the predicted and filtered ensemble are given by

$$\boldsymbol{P}_k^- = \overline{(\hat{\boldsymbol{x}}_k^- - \boldsymbol{x}_k)(\hat{\boldsymbol{x}}_k^- - \boldsymbol{x}_k)^T} \tag{3.15}$$

$$\boldsymbol{P}_k = \overline{(\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k)(\hat{\boldsymbol{x}}_k - \boldsymbol{x}_k)^T} \tag{3.16}$$

Since the true state is in general hard to acquire one can instead redefine the ensemble covariance in regards to the ensemble means $(\overline{\hat{x}_k}, \overline{\hat{x}_k^-})$ as shown in (3.17. 3.18).

$$\boldsymbol{P}_k^- = \overline{(\hat{\boldsymbol{x}}_k^- - \overline{\hat{\boldsymbol{x}}_k^-})(\hat{\boldsymbol{x}}_k^- - \overline{\hat{\boldsymbol{x}}_k^-})^T} \tag{3.17}$$

$$\boldsymbol{P}_k = \overline{(\hat{\boldsymbol{x}}_k - \overline{\hat{\boldsymbol{x}}_k})(\hat{\boldsymbol{x}}_k - \overline{\hat{\boldsymbol{x}}_k})^T} \tag{3.18}$$

Introducing this enforces an interpretation that the ensemble mean is the best estimate and that the spreading of the ensemble around defines the error in the ensemble mean. Now the error covariance can be represented using an appropriate ensemble of model states. How to select this ensemble will be described later in this chapter. Consider the error term from the Monte Carlo integration given in Section 2.5. The description of the error term also hold for the ensemble covariance. As the size of the ensemble N increases, the errors in the MC sampling will decrease proportional to $1/\sqrt{N}$. Evensen [16] concluded that the information contained by a full probability density function can be exactly represented by an infinite ensemble of model states.

3.3.3 Ensemble representation

Earlier the system states $\boldsymbol{x}$ and poorly know parameters $\boldsymbol{\theta}$ were defined. These can be put together in a matrix $\boldsymbol{A}(\boldsymbol{z}, t_k) \in \mathcal{R}^{n \times N}$, where $n = n_x + n_\theta$, holding the N ensemble members at time t_k. In mathematical terms this become

$$\boldsymbol{A}(\boldsymbol{z}, t_k) = \begin{bmatrix} \boldsymbol{x}^{(1)}(\boldsymbol{z}, t_k) & \boldsymbol{x}^{(2)}(\boldsymbol{z}, t_k) & \cdots & \boldsymbol{x}^{(N)}(\boldsymbol{z}, t_k) \\ \boldsymbol{\theta}^{(1)}(\boldsymbol{z}, t_k) & \boldsymbol{\theta}^{(2)}(\boldsymbol{z}, t_k) & \cdots & \boldsymbol{\theta}^{(N)}(\boldsymbol{z}, t_k) \end{bmatrix} \tag{3.19}$$

The ensemble mean can now be written as

$$\overline{\boldsymbol{A}}(\boldsymbol{z}, t_k) = A(\boldsymbol{z}, t_k) \cdot \mathbf{1}_N \tag{3.20}$$

where $\mathbf{1}_N$ is a vector consisting of $N \times 1$ elements of value $1/N$. The ensemble covariance in space can then be estimated as

$$\boldsymbol{P}(\boldsymbol{z}_1, \boldsymbol{z}_2, t_k) = \frac{1}{N-1}(\boldsymbol{A}(\boldsymbol{z}_1, t_k) - \overline{\boldsymbol{A}}(\boldsymbol{z}_1, t_k))(\boldsymbol{A}(\boldsymbol{z}_2, t_k) - \overline{\boldsymbol{A}}(\boldsymbol{z}_2, t_k))^T \tag{3.21}$$

The measurements can also be formulated using an ensemble representation. For a system consisting of J measurements, J vectors consisting of N perturbed measurements is defined as

$$\boldsymbol{y}_j^l = \boldsymbol{y}_j^t + \boldsymbol{v}_j^l, \qquad l = 1, \ldots, N \quad j = 1, \ldots, J \tag{3.22}$$

where $\boldsymbol{y}_j^t$ is the "true" measurement from the original process. These variables can be stored in a measurement matrix $\boldsymbol{\mathcal{Y}}_j$ and a measurement perturbation matrix $\boldsymbol{\mathcal{E}}_j$ as in (3.23)-(3.24) for each of the system measurements $j \in [1 \ldots J]$.

$$\boldsymbol{\mathcal{Y}}_j = (\boldsymbol{y}_j^1, \boldsymbol{y}_j^2, \ldots, \boldsymbol{y}_j^N) \tag{3.23}$$

$$\boldsymbol{\mathcal{E}}_j = (\boldsymbol{v}_j^1, \boldsymbol{v}_j^2, \ldots, \boldsymbol{v}_j^N) \tag{3.24}$$

For the $j \in [1 \ldots J]$ measurement the covariance can be estimated using the ensemble measurement perturbations as follows

$$\boldsymbol{V}(t_k)_j = \frac{\boldsymbol{\mathcal{E}}_j \boldsymbol{\mathcal{E}}_j^T}{N-1} \tag{3.25}$$

The argument for using this perturbation is further investigated in Section 4.1 and shows that in the case of infinite ensemble size the EnKF will converge to a KF for a linear system.

3.3.4 EnKF algorithm

The ensemble can be expressed as a function of time only by stacking the states and parameters. Given the a finite set of space parameters $\boldsymbol{z} = [\boldsymbol{z}_1 \ldots \boldsymbol{z}_g]$, S states and P poorly known parameters at each location. Each state $x_i, i \in [1, S]$ and parameter $\theta_j, j \in [1, P]$ can then be expressed for all locations in space as follows.

$$\boldsymbol{\mathcal{X}}_i(t_k) = \begin{bmatrix} x_i^{(1)}(z_1, t_k) & x_i^{(2)}(z_1, t_k) & \cdots & x_i^{(N)}(z_1, t_k) \\ \vdots & \vdots & \vdots & \vdots \\ x_i^{(1)}(z_g, t_k) & x_i^{(2)}(z_g, t_k) & \cdots & x_i^{(N)}(z_g, t_k) \end{bmatrix} \tag{3.26a}$$

$$\boldsymbol{\theta}_j(t_k) = \begin{bmatrix} \theta_j^{(1)}(z_1, t_k) & \theta_j^{(2)}(z_1, t_k) & \cdots & \theta_j^{(N)}(z_1, t_k) \\ \vdots & \vdots & \vdots & \vdots \\ \theta_j^{(1)}(z_g, t_k) & \theta_j^{(2)}(z_g, t_k) & \cdots & \theta_j^{(N)}(z_g, t_k) \end{bmatrix} \tag{3.26b}$$

The $\boldsymbol{A}$ matrix can now be redefined from (3.19) containing all ensemble states and parameters *for all locations in space*. This matrix works as an augmented state vector to be used in the EnKF. Combining (3.26a) and (3.26b) the ensemble matrix can be derived as

$$\boldsymbol{A}_k = \boldsymbol{A}(t_k) = \left[\begin{array}{c} \boldsymbol{\mathcal{X}}_1(t_k) \\ \boldsymbol{\mathcal{X}}_2(t_k) \\ \vdots \\ \boldsymbol{\mathcal{X}}_S(t_k) \\ \hline \boldsymbol{\theta}_1(t_k) \\ \boldsymbol{\theta}_2(t_k) \\ \vdots \\ \boldsymbol{\theta}_P(t_k) \end{array}\right] \tag{3.27}$$

The same notation can be used for the measurements using (3.23)

$$\boldsymbol{Y}_k = \boldsymbol{Y}(t_k) = \left[\begin{array}{c} \boldsymbol{\mathcal{Y}}_1(t_k) \\ \boldsymbol{\mathcal{Y}}_2(t_k) \\ \vdots \\ \boldsymbol{\mathcal{Y}}_J(t_k) \end{array}\right] \tag{3.28}$$

The matrix containing the measurements from the true system is stored in $\boldsymbol{Y}_k^t = [\boldsymbol{y}_1^t \boldsymbol{y}_2^t \dots \boldsymbol{y}_J^t]^T$. Similarly this can be done for the ensemble measurement perturbation using (3.24)

$$\boldsymbol{E}_k = \boldsymbol{E}(t_k) = \left[\begin{array}{c} \boldsymbol{\mathcal{E}}_1(t_k) \\ \boldsymbol{\mathcal{E}}_2(t_k) \\ \vdots \\ \boldsymbol{\mathcal{E}}_J(t_k) \end{array}\right] \tag{3.29}$$

For a general nonlinear system the measurement is given according to (3.14c). The estimated states and parameters for all the ensembles are denoted $\hat{\boldsymbol{A}}_k$. Let the measurements of the estimated ensemble at time t_k be given by the following nonlinear relationship

$$\mathcal{M}(\hat{\boldsymbol{A}}_k) = [\hat{\boldsymbol{y}}^{(1)} \dots \hat{\boldsymbol{y}}^{(N)}] = [g((\hat{\boldsymbol{x}}_k^-)^{(1)}, (\hat{\boldsymbol{\theta}}_k^-)^{(1)}) \dots g((\hat{\boldsymbol{x}}_k^-)^{(N)}, (\hat{\boldsymbol{\theta}}_k^-)^{(N)})] \tag{3.30}$$

For convenience the estimate for the states, parameters and measurements are stacked in a big ensemble matrix as shown below.

$$\hat{\boldsymbol{\mathcal{A}}}_k = \hat{\boldsymbol{\mathcal{A}}}(t_k) = \left[\begin{array}{c} \hat{\boldsymbol{A}}_k \\ \mathcal{M}(\hat{\boldsymbol{A}}_k) \end{array}\right] \tag{3.31}$$

The Ensemble Kalman filter updates the estimate of the states and variables by updating the ensemble matrix $\bar{\hat{\boldsymbol{\mathcal{A}}}}_0$. Note that $\mathcal{M}(\hat{\boldsymbol{A}})$, in the ensemble matrix from (3.31), is treated as a diagnostic variable in the system. This is because its update is not used in the next step of the filter. The EnKF also updates the mean of the ensemble matrix $\bar{\hat{\boldsymbol{\mathcal{A}}}}_k$. The superscript $^-$, as defined previously, denotes that an estimate is a priori. To initialize the filter one must provide the model and measurement covariances, respectively $\boldsymbol{Q}_k$ and $\boldsymbol{V}_k$. In addition one must specify an initial ensemble, which will be discussed in further detail later. Let $\boldsymbol{H}_k$ be the measurement index matrix at each time step k defined such that

$$\boldsymbol{H}_k \hat{\boldsymbol{\mathcal{A}}}_k = \mathcal{M}(\hat{\boldsymbol{A}}_k) \tag{3.32}$$

Using this notation the Ensemble Kalman Filter is summarized in Algorithm 3. There are three sub functions, which involves the statistics, in the presented algorithm and they will be described in more detail in the next section.

input: $\boldsymbol{Q}_k$, $\boldsymbol{V}_k$
$\hat{\boldsymbol{\mathcal{A}}}_0 \Leftarrow$ `computeInitialEnsemble(...)`
while *true* **do**

Prediction

$$
\begin{aligned}
d\boldsymbol{q} &\leftarrow \texttt{computeModelNoise}(\boldsymbol{Q}_k) \\
\hat{\boldsymbol{\mathcal{A}}}_k^- &\leftarrow \boldsymbol{f}(\hat{\boldsymbol{\mathcal{A}}}_{k-1}) + d\boldsymbol{q} \\
\boldsymbol{P}_k^- &\leftarrow \overline{(\hat{\boldsymbol{\mathcal{A}}}_k^- - \bar{\hat{\boldsymbol{\mathcal{A}}}}_k^-)(\hat{\boldsymbol{\mathcal{A}}}_k^- - \bar{\hat{\boldsymbol{\mathcal{A}}}}_k^-)^T}
\end{aligned}
$$

Filtering

$$
\begin{aligned}
\boldsymbol{E}_k &\leftarrow \texttt{computeMeasurementNoise}(\boldsymbol{V}_k) \\
\boldsymbol{Y}_k &\leftarrow \boldsymbol{Y}_k^t + \boldsymbol{E}_k \\
\boldsymbol{K}_k &\leftarrow \boldsymbol{P}_k^- \boldsymbol{H}_k (\boldsymbol{H}_k \boldsymbol{P}_k^- \boldsymbol{H}_k^T + \boldsymbol{V})^{-1} \\
\hat{\boldsymbol{\mathcal{A}}}_k &\leftarrow \hat{\boldsymbol{\mathcal{A}}}_k^- + \boldsymbol{K}_k (\boldsymbol{Y}_k - \boldsymbol{H}_k \hat{\boldsymbol{\mathcal{A}}}_k^-) \\
\bar{\hat{\boldsymbol{\mathcal{A}}}}_k &\leftarrow \bar{\hat{\boldsymbol{\mathcal{A}}}}_k^- + \boldsymbol{K}_k (\boldsymbol{Y}_k - \boldsymbol{H}_k \bar{\hat{\boldsymbol{\mathcal{A}}}}_k^-) \\
\boldsymbol{P}_k &\leftarrow \overline{(\hat{\boldsymbol{\mathcal{A}}}_k - \bar{\hat{\boldsymbol{\mathcal{A}}}}_k)(\hat{\boldsymbol{\mathcal{A}}}_k - \bar{\hat{\boldsymbol{\mathcal{A}}}}_k)^T}
\end{aligned}
$$

end

Algorithm 3: Ensemble Kalman Filter (EnKF)

3.3.5 Managing the uncertainties

The EnKF is based on statistics and consequently uncertainties comes into play in many parts of the algorithm. Generating ensembles or adding model/measurment noise is all built on statistic properties.

Adding of model and measurement noise is important for the filter to work. The statistical theory behind the measurement noise perturbation is discussed further in Section 4.1. The model noise is used to describe model uncertainty in the filter. This is a characteristic of EnKF, namely that it solves both the parameter and the state estimation problem. The adjustment of the model noise is something that vary with the different problem cases and is based on, among other, the severeness of the nonlinearities of the problem. Some questions around the model noise for large scale system will also be addressed in Section 10.3.

Constructing the initial ensemble is an essential part of the kalman filter. The initial ensemble contains information about the initial states, parameters and their uncertainties. There are various strategies for creating these samples representing the uncertainties. Evensen [16] suggested three sampling strategies that might be applied

1. Sampling all variables randomly. The mean and variance will vary within the accuracy that can be expected for a given ensemble size.
2. Sampling all variables randomly, but correcting the sample such that it will have the correct mean and variance.
3. Using the improved sampling scheme presented by Evensen.

From running several tests Evensen concluded that there was a slight improvement from strategy 1 to strategy 2. The third strategy and more complex strategy gave a better result than the two

previous even though it is not so computationally expensive. The improved sampling scheme seeks to generate ensembles with full rank and better conditioning than using a random sample. Evensen [17] suggested that since the improved sampling scheme also generates a better conditioning of the ensemble during the forward integration it might be useful when computing the model noise. The improved sampling can be used in the following two ways

- Reduce the computation time by decreasing the ensemble size and still obtain the same results
- Improve the EnKF results even further

In this paper the first two sampling strategies will be used for the comparison of linear Kalman filter and EnKF in Section 4.

A common factor of these sampling strategies is that the samples are created using the mean and the covariance. To create samples based on these variables one need some more information. An assumption that the ensemble is Gaussian distributed is made. In practice this means that the probability density function (pdf) of the a priori ensemble is assumed to be Gaussian distributed for the update to be correct. The Gaussian assumption result in the EnKF solving for the mean and not the mode. In the case of a Gaussian distribution these values coincides. One reason this assumption is made is that the mean is easier to estimate using a small ensemble size. Nævdal [35] tested the EnKF on a scalar case with skew-normal initial distribution of the state. The results are shown in Figure 3.1. On the left is the initial ensemble, with a skew-normal distribution. On the

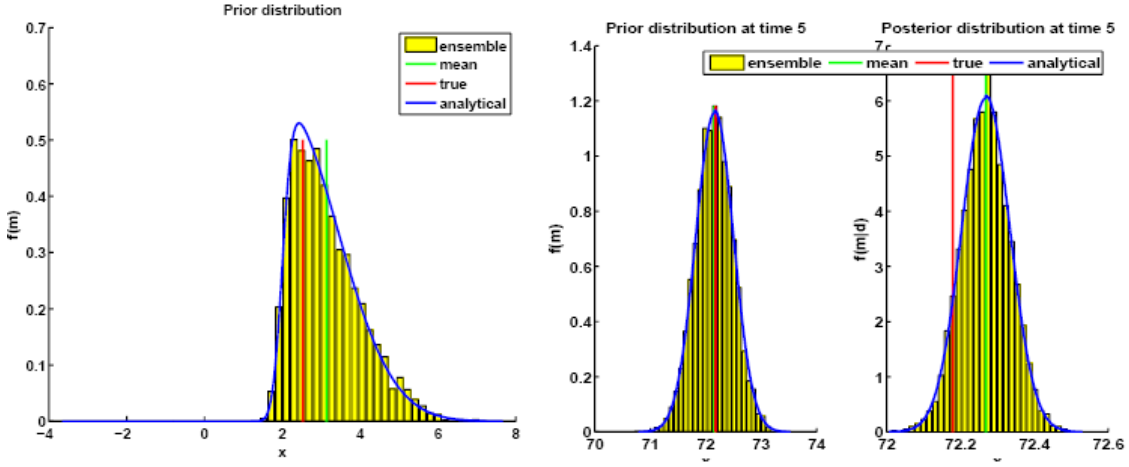


Figure 3.1: Example illustrating EnKF on a system with non-gaussion prior

right, the prior and posterior ensemble is plotted at time step 5. For the initial ensemble the mean is not a good estimate of the true solution since the distribution is non-Gaussian. This error, as the EnKF assimilates data, gets smaller and the mean becomes a better estimate. In general the EnKF have proven to work well with a large number of nonlinear dynamic models. Evensen [16] gave some examples both where the EnKF inherited some non-Gaussian structures in the analyzed ensemble and where the EnKF failed.

The implementation for a large scale systems in this paper uses a more sophisticated sampling scheme. This scheme takes into consideration a 3-dimensional correlation between the elements in the distribution. More details on this is to be found in Section 7.3. Given this introduction to the EnKF it will be interesting to compare it with a linear KF to test some of the basic principles.

Chapter 4

Comparison of a linear Kalman Filter and EnKF

The purpose of this chapter is two compare the performance of a linear Kalman Filter (KF) and EnKF. There are two reasons for doing this comparison. First, this comparison acts as an argument for using the measurement perturbation scheme as described in Section 3.3.3. The alternative to this scheme would be letting all ensemble members see the same measurement. This section seeks to prove that by using this scheme, the EnKF converges toward the KF for a linear case. Secondly this comparison is works as a simple way of verifying the EnKF implementation that is described in Section 7.3. Since this is a simple linear example it is possible to quickly implement an algorithm as a reference. Setting up the provided EnKF code on the other hand took quite a lot of effort, since this is adapted to reservoir cases. The exercise however proved to be useful for understanding the complex implementation. In this section there will first be made a theoretical comparison and afterward an implementation of each of the two filters are made to verify this.

4.1 Theoretical comparison

When comparing KF and EnKF it is natural to examine the covariance matrices. The analysis scheme in the KF uses the definitions of (3.16). Burgers et al. [10] showed that it is essential that the observations are treated as random variables with a mean equal to the system observation and covariance equal to V^{true}. The ensemble of observations is given in (3.22). Comparing the filtering step in the KF and the EnKF one can see that the relation between the apriori and aposteriori ensemble is identical, apart from the use of $\boldsymbol{P}_k^-$ and $\boldsymbol{P}_k$. Given an ensemble of size N and the a posteriori update of the state and the state mean from Algorithm 3, the following relationship can be derived for ensemble member l.

$$\hat{\boldsymbol{x}}^l - \overline{\hat{\boldsymbol{x}}}^l = (\boldsymbol{I} - \boldsymbol{K}\boldsymbol{H})((\hat{\boldsymbol{x}}^-)^l - (\overline{\hat{\boldsymbol{x}}^-})^l) + \boldsymbol{K}(\boldsymbol{y}_l - \overline{\boldsymbol{y}}) \tag{4.1}$$

Here the Kalman gain is the same as in the formula from both Algorithm 1 and 3. Using this relationship and considering the ensemble covariance from the EnKF the following derivation can

be done.

$$\begin{aligned}
\boldsymbol{P}_k &= \overline{(\hat{x}_k - \overline{\hat{x}_k})(\hat{\boldsymbol{x}}_k - \overline{\hat{\boldsymbol{x}}_k})^T} \\
&= \overline{\Big((\boldsymbol{I} - \boldsymbol{K}\boldsymbol{H})(\hat{\boldsymbol{x}}_k^- - \overline{\hat{\boldsymbol{x}}_k^-}) + \boldsymbol{K}(\boldsymbol{y}_k - \overline{\boldsymbol{y}}_k)\Big)(\ldots)} \\
&= (\boldsymbol{I} - \boldsymbol{K}\boldsymbol{H})\overline{(\hat{\boldsymbol{x}}_k^- - \overline{\hat{\boldsymbol{x}}_k^-})(\hat{\boldsymbol{x}}_k^- - \overline{\hat{\boldsymbol{x}}_k^-})^T}(\boldsymbol{I} - \boldsymbol{K}\boldsymbol{H})^T + \boldsymbol{K}\overline{(\boldsymbol{y}_k - \overline{\boldsymbol{y}}_k)(\boldsymbol{y}_l - \overline{\boldsymbol{y}})^T}\boldsymbol{K}^T \\
&= (\boldsymbol{I} - \boldsymbol{K}\boldsymbol{H})\boldsymbol{P}_k^-(\boldsymbol{I} - \boldsymbol{K}\boldsymbol{H})^T + \boldsymbol{K}\boldsymbol{V}\boldsymbol{K}^T \\
&= \boldsymbol{P}_k^- - \boldsymbol{K}\boldsymbol{H}\boldsymbol{P}_k^- - \boldsymbol{P}_k^-\boldsymbol{H}^T\boldsymbol{K}^T + \boldsymbol{K}(\boldsymbol{H}\boldsymbol{P}_k^-\boldsymbol{H}^T + V)\boldsymbol{K}^T \\
&= (\boldsymbol{I} - \boldsymbol{K}\boldsymbol{H})\boldsymbol{P}_k^-
\end{aligned} \tag{4.2}$$

Here the result is the minimum error covariance as used in the KF scheme. The conclusion here is that the EnKF analysis step converges to that of the KF when the ensemble size converges to infinity. For this to hold the observations (y) must be treated as random variables.

4.2 Implementation

Given the theoretical comparison of EnKF and KF it is natural to test this with a small linear system and a pure state estimation problem. Consider the following system

$$\begin{aligned}
\boldsymbol{x}_{k+1} &= \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{q}_k \\
\boldsymbol{y}_k &= \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{v}_k
\end{aligned} \tag{4.3}$$

where

$$\begin{aligned}
\boldsymbol{A} &= \begin{bmatrix} 0.5 & 0.3 \\ 0.2 & 0.1 \end{bmatrix} & \boldsymbol{C} &= \begin{bmatrix} 1 & 0.5 \end{bmatrix} \\
E[\boldsymbol{q}_k\boldsymbol{q}_k^T] &= \begin{bmatrix} 0.01 & 0 \\ 0 & 0.02 \end{bmatrix} & E[\boldsymbol{v}_k\boldsymbol{v}_k^T] &= \begin{bmatrix} 0.01 \end{bmatrix}
\end{aligned} \tag{4.4}$$

The state is initialized as

$$\boldsymbol{x}_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{4.5}$$

The system is analyzed using both a linear KF according to Algorithm 1 and an EnKF according to Algorithm 3. The filters seek to estimate the two states in the system. The KF and EnKF state was initialized as follows:

$$\hat{\boldsymbol{x}}_0 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \tag{4.6}$$

The covariance matrix for this system in both the KF and the EnKF is shown in Figure 4.1

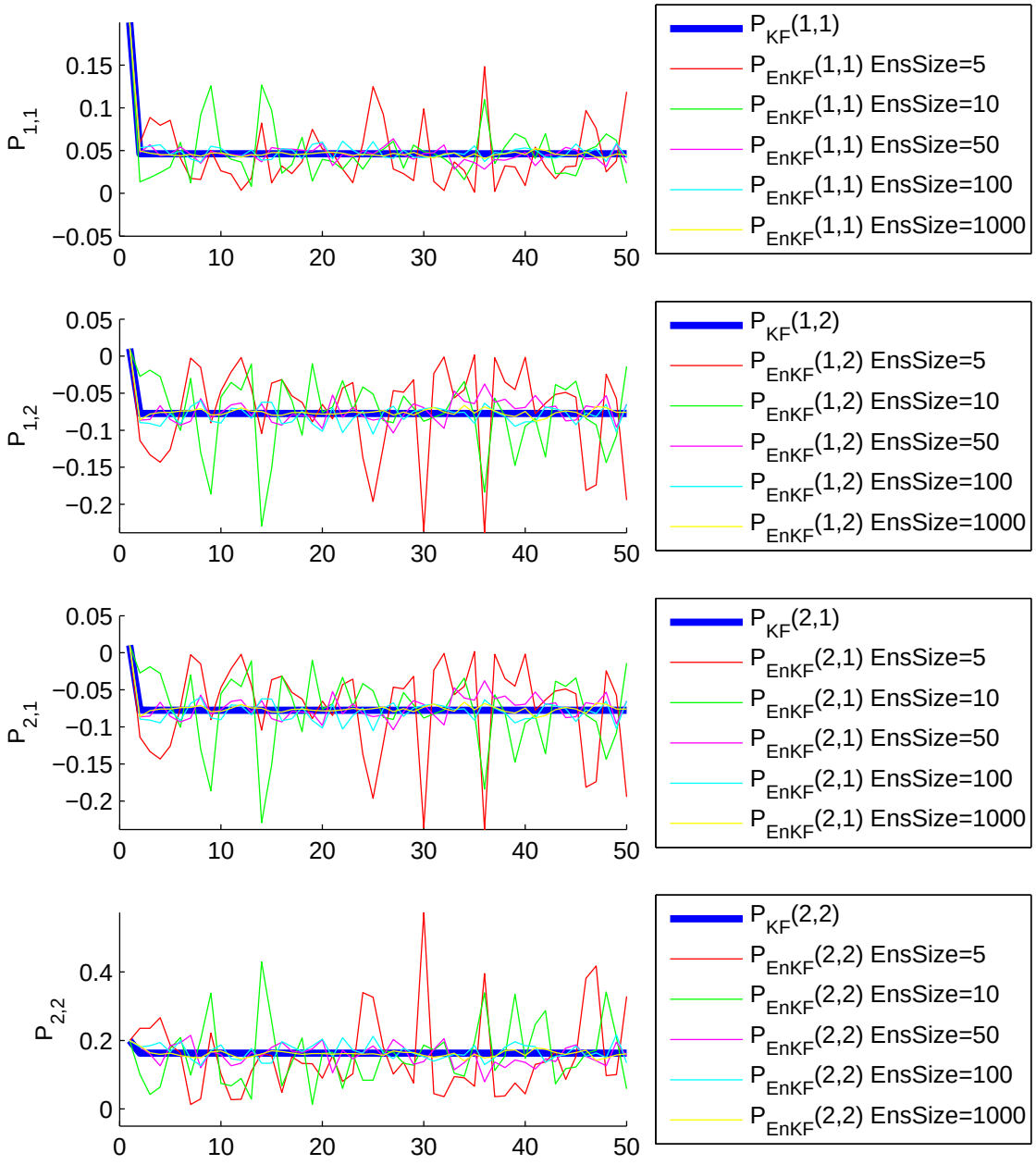


Figure 4.1: Comparison of KF and EnKF - Covariance Matrix

The state estimates for this system in both the KF and the EnKF are shown in Figure 4.2 From both Figure 4.1 and 4.2 one can observe that as the ensemble size grows larger the EnKF

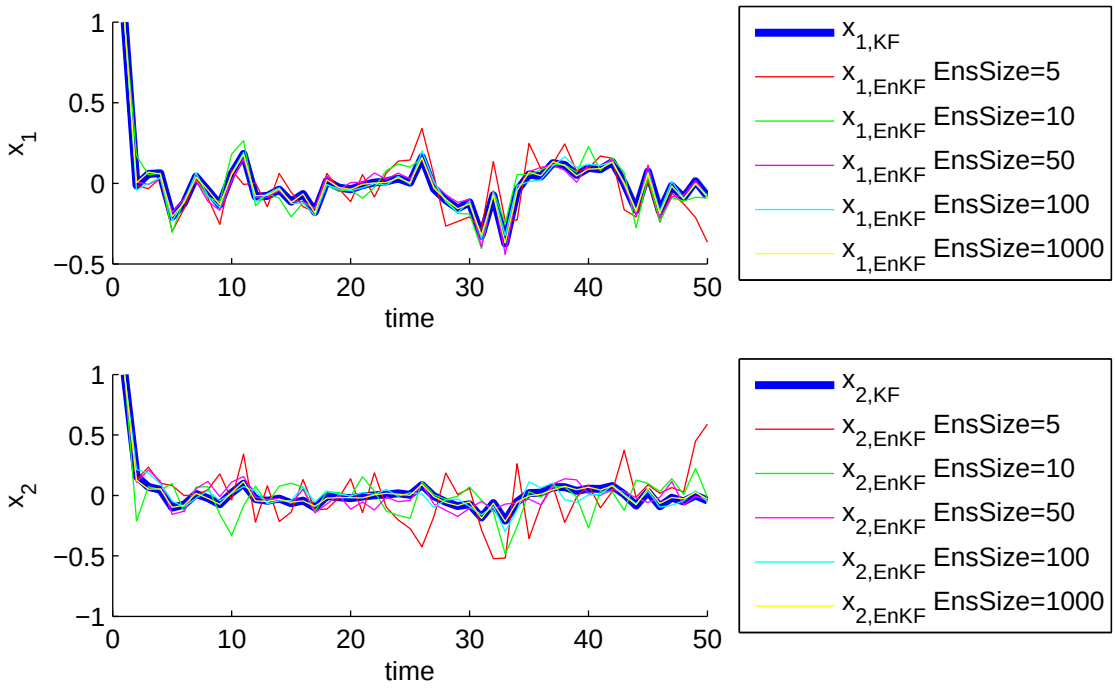


Figure 4.2: Comparison of KF and EnKF - States

behavior approaches that of the KF. This supports the theory presented in the previous section. The MATLAB [1] code for the filter that was used in these tests can be found in Appendix A. The same tests were run with the IRIS EnKF implementation (see Section 7.3) and similar results were shown. As described earlier setting up these tests took quite a lot of effort since the IRIS code was originally built for reservoir problems and some smaller modifications had to be made to make it work. The consistency of the IRIS implementation was proven and a lot of experience with the code was gained by the author.

Chapter 5

EnKF for continuous reservoir model updating

The EnKF was as mentioned earlier first introduced in association with oceanic forecasting models. But later effort have been put into adapting this method for continuous reservoir model updating. Currently a lot of effort is put into developing this method as a contestant for automatic history matching. However, there is still some work left to be done. This chapter is meant as an introduction to EnKF for continuous reservoir model updating. To start of an outline of the previous work done on the subject is given. Following this is an outline of the methodology of setting up and running the EnKF as a parameter estimation tool. Last a short comparison with some other automatic history matching methods is given.

5.1 Previous work

In recent years there has been an interest in both mathematical and statistical methods for history matching. Most of these methods have considered the problem as a parameter estimation problem. In 1994 Evensen [18] introduced the EnKF as parameter and state estimation method. The EnKF differs from the other methods because it estimates both the parameters and the states. Previous methods usually perform history matching by minimizing an objective function. This objective function describes the difference between the observed and simulated measurements. This problem usually becomes highly nonlinear and the solution lies in a space with dimension equal to the number of parameters. The usage of EnKF for history matching was first proposed by Nævdal et al. [36] and following his work many others have contributed to the discussion of EnKFs potential for history matching. Some of the papers that discuss EnKF applications in regards to reservoir simulation models are listed below.

- Nævdal et al. (2002) [36]: Used EnKF for permeability estimation on a simple reservoir application and showed good results.

- Nævdal et al. (2003) [37]: Continued showing good results using EnKF. Now estimating the permeability of the whole reservoir using a simplified 2D reservoir model of a North Sea field.

- Gu and Oliver (2004) [23]: Examined the use of EnKF for state and parameter estimation on the PUNQ-S3 reservoir test case and got good results. Some issues regarding an overshoot in the porosity and permeability was presented.

- Brouwer et al. (2004) [7]: Investigated EnKF in a closed loop setting with optimal reservoir control. They showed good results on a simple 2-dimensional model.
- Gao,Zafari and Reynolds (2005) [21]: Compared EnKF with another method called Randomized Maximum Likelihood (RML). Both methods showed similar results. One major difference between the method is that RML is an adjoint based method whereas the EnKF does not require much effort in coupling with the reservoir model.
- Liu and Oliver (2005) [32]: Used the EnKF both for history matching and for facies estimation in a reservoir simulation model. This is a very difficult problem both to implement and solve. EnKF showed many good properties for this type of problem. The same year they compared these results to that of a traditional gradient-based minimization method and EnKF outperformed the traditional method [33].
- Wen and Chen (2005) [48]: Provided an improved EnKF algorithm on a two dimensional reservoir. Examined the effect of variation in the ensemble size.
- Zafari and Reynolds (2005) [50]: Tested the method on some nonlinear problems to validate the EnKF. EnKF was proven to have difficulties on multi-modal distributions and that the Gaussian assumption of EnKF is very critical. They also tested the improved algorithm of Wen and Chen and found it to be inconsistent.
- Skjervheim et al. (2005) [44]: Suggested a method based on EnKF to incorporate 4D seismic data in continuous model updating.
- Haugen et al. (2006) [24]: Tested EnKF for history matching on a North Sea field case. They showed how EnKF can improve the model parameters for history matching and also discussed the updating of the reservoir states. They showed promising results, but concluded that work should be done to be able to estimate other reservoir parameters using the EnKF.
- Evensen et al. (2007) [19]: Also showed how EnKF can be used in assisting history matching of a North Sea reservoir model. They also attempted to estimate WOC and GOC using EnKF and showed an decrease in uncertainty regarding the parameters.

This time line shows the evolution of EnKF from an early theoretical point of view toward a more mature tool for assisting history matching. This paper will focus on the EnKF implementation on a test case reservoir, hence not all of the applications in the newer literature will be tested in the implementation part (Section 7). A closer look on some of the newest applications of EnKF will be given in the further work chapter in Section 10.

5.2 The EnKF methodology

The method of history matching using EnKF is a complex procedure which require collaboration between many areas of expertise. This description seeks to put the procedure into a system and explain each of the steps. Solving a parameter estimation problem for a reservoir can be organized in the following procedure

1. Create an initial model of the reservoir based on geological interpretation
2. Parameterize the estimation problem (Static and dynamic variables, measurements, etc.)
3. Define the problem in mathematical terms (Put the parameters into the mathematical form of Section 3.3)
4. Create the initial ensemble
5. Run the filter
6. Interpret the results

If the results are not satisfactory one have to analyze them in order to determine where in the procedure it went wrong and redo the procedure from that point. The steps in this procedure will now be discussed in more detail.

The creation and maintenance of a reservoir model is a complex task. This process includes geological modelling, seismic reservoir characterization, reservoir simulation, and conditioning models to all available data. To implement EnKF as a tool for assisting model updating there are many challenges. A very important aspect is to have an understanding for the reservoir geology. By this it is understood that all initial information about the reservoir have to be taken into consideration. It is important to map out all available information and to create an initial reservoir model which describes the geology in the reservoir. Before a reservoir starts producing one can never be one hundred percent sure of what is underneath the surface. And this is where continuous model updating comes into play. This underlines that before the EnKF comes into play a lot of work has to be done from the reservoir engineers to create a good model. After this has been done the information has to be incorporated in the EnKF configuration.

The first step is to create a parameterization of the problem. The idea is to find the parameters in the model wherein the uncertainty lies. As described by G. Evensen [19] the structural model is assumed accurate, since there is no current way of estimating structural parameters using EnKF. To specify the uncertainty in the model good communication is critical between the reservoir engineers and the people setting up the EnKF. So what parameters can be estimated. Evensen listed some examples of parameters that can be estimated in reservoir models:

- Fluid Contacts (GOC, WOC etc.)
- Porosity and permeability fields
- Fault transmissivities
- Vertical transmissivities

The fluid contacts are parameters where initial information usually is derived from the drilling of wells through the specific areas. This often leads to a large uncertainty in these important parameters and a need for better estimates is often present. Porosity and permeability fields describes the reservoirs

ability to transport fluids. These are the main parameters to adjust to match both observed rates and the timing of water and/or gas breakthrough. The transmissivities are important parameters in for instance reservoirs containing many faults and little information about pressure. In addition to the parameters also the states in the reservoir can be estimated. These usually are the pressure P and the saturations of gas, oil and water (S_g, S_o and S_w).

In addition to the static and dynamic variables, the measurements that should be used have to be defined. It is important to choose the measurements wisely as the measurements are the only information available to tune the model. But in general the measurements are given by facilities and there is no real choice. An understanding for history matching and concepts of observability can however assist in planning where to incorporate measurements in the facilities. This will be discussed further in Chapter 6.

Once the parameters and states to be estimated are defined an augmented state vector for the filter can be derived as

$$\begin{bmatrix} x_i^{(l)}(z_1,t_k) \\ \vdots \\ x_i^{(l)}(z_g,t_k) \\ \theta_j^{(1)}(z_1,t_k) \\ \vdots \\ \theta_j^{(1)}(z_g,t_k) \end{bmatrix} \tag{5.1}$$

where $x_i^{(l)}(z_o,t_k)$ denotes the states for ensemble l in grid block o at time k. Similarly $\theta_j^{(l)}(z_o,t_k)$ denotes the poorly known parameters to be estimated. The measurements to use in the filter updating is also to be defined. Usually these consists of rate and pressure measurements. Now the model is defined and the ensemble can be written in the form presented in (3.31).

When the problem has been defined the next step is to initialize the EnKF. This is a very important part of the EnKF as will be illustrated next. Evensen [16] stated that the solution for the static parameters is only searched for in the space spanned by the initial ensemble of parameters. An example to illustrate this is given in Appendix B. This poses that one can effectively reduce the degrees of freedom by spanning a smaller space in which the filter searches for the solution. On the other hand it is important that consideration is put into the problem reduction so that the solution does not fall outside the span. In practice the problem reduction can be done by assuming the fields of permeability and porosity to be smooth. This smoothness can be described in the prior statistics using vertical and horizontal correlations.

After the problem has been parameterized and the ensemble initialized the EnKF is ready to be run. The EnKF algorithm can be implemented with different options that might be turned on or off. These options are in general available to implement extra features that might improve the EnKFs ability to solve the problem. Some of the improvements will be discussed in the implementation section in Chapter 7.

After the EnKF has been run over a period in time the results needs to be interpreted and verified. G. Evensen et al. [19] asked the question *"What are we solving for?"*. As discussed earlier the EnKF solves for the mean of the posterior pdf. In addition to solving for the mean, the uncertainty is described in the standard deviation. In some cases it can also be interesting to observe the solution in the individual ensemble. One example can be if the mean does not give a solution that is realistic, a realistic solution might be caught in one of the ensemble members. Also in case something goes wrong in the parameter estimation one can interpret information from the individual samples. In any case this comprehension of the results call for a collaboration with the reservoir engineers. Both the work by G. Evensen et al. and V. Haugen et al. [24] describes interpretation of

results from a North Sea Field Case.

In general one can say that performing a history matching using the EnKF is a complex task which requires both in depth knowledge of the reservoir and EnKF properties. There are also some advantages in using EnKF compared to more traditional history matching methods. This will be described in the next part.

5.3 Comparison with traditional history matching

The EnKF is a new approach to history matching. The history matching problem for reservoir models on the other hand has been around for quite a while. In this section some of the differences between the classical method for automatic history matching and EnKF will be outlined.

There are four categories in which the history matching methods can be placed under

1. Manual history matching
2. Gradient-based algorithms
3. Genetic algorithms
4. Statistical methods

The manual history matching is the first method, where the reservoir engineers would look at the data from a reservoir and try to tune the parameters manually to improve the fit with the measurements. The gradient based algorithm methods for history matching involves minimizing an objective function, thereby minimizing the difference between the observed and simulated measurements. These methods in general use the following loop:

1. Run the flow simulator for the complete history matching period
2. Evaluate the cost function
3. Update the static parameters and go to the first step

The search in these methods usually search a space with dimension equal to the number of parameters. This is a very complex problem and is usually highly nonlinear with many local minima. Up until 1972 most of this work was done using a perturbation method according to Dougherty [14]. A lot of work have been put into creating good algorithms for the history matching optimization problem, and the gradient calculation will be discussed later. The genetic algorithms are algorithms that also solves the problem of minimizing an objective function. These evolutionary algorithms are usually good for solving large scale problems with many local minima. On the other hand there are no guarantees in the solution of these algorithms. R.W. Schulze-Riegert et al. [43] did some work with evolutionary algorithms and also showed how prior information can improve convergence. The last class of methods is the statistical methods and the EnKF falls under this category. In the rest of this section the comparison between EnKF and other methods will focus on comparison with the gradient-based methods. Note that these method will be referred to as traditional history matching.

One of the differences between EnKF and traditional history matching lies in what the method solves for. The methods that search for the minimum of a cost function solves for the mode of the posterior pdf. The EnKF on the other hand solves for the mean of the pdf since the mean is easier to estimate with a small ensemble size. This, as described in the theory earlier, imposes a Gaussian assumption to the pdf. Effectively EnKF simplifies the problem by searching for the mean and by limiting the search to the space spanned by the ensemble. This makes EnKF more computationally

efficient, but requires greater care in initializing the problem. Similar methods for restricting the search in gradient methods has also been applied and is referred to as Regularization. C.B. Chung et al. [12] explained how a priori information could be incorporated into history matching.

The computation of the pdf also raises the question of uncertainty. In EnKF the uncertainty is calculated directly by the standard deviation of the ensemble and the measurements. In gradient based methods the uncertainty has to be calculated by different means. B.A. Foss [20] proposed a method for calculating the uncertainty in the parameters by using the Hessian of the objective function. This might become a complex task for large objective functions in big reservoir models. A newer method to quantify the uncertainty in history matching is Randomized Maximum Likelihood (RML). Liu and Oliver [33] compared the EnKF with this such a RML method. They tested both methods ability to history match geologic facies, which is a very difficult problem. In their experiments the EnKF outperformed the traditional method both in calculation time and results.

Another thing that separates the EnKF from traditional history matching lies in the computation of the gradient. The traditional methods requires the computation of the gradient of the objective function values. Even though the newer adjoint based methods have reduced the computation significantly the gradient computations requires some simulations of the reservoir model for the whole the horizon. The task that requires most computation time is just simulation of the reservoir hence extra simulations should be avoided. J.R.P. Rodrigues [39] proposed an implementation of the forward and adjoint based methods for derivative calculation in a full-featured adaptive implicit black-oil simulator. In EnKF the gradient is not calculated which saves a lot of extra runs of the simulator compared to the traditional gradient-based methods. In general the computation time of the EnKF is equal to N simulations over the whole horizon plus some overhead for the filter calculations and model restarts. The EnKF unlike most of the gradient-based methods is a recursive algorithm. This means that when new data arrives one can just compute the next step and there is no need to run a full new optimization over the horizon.

The early versions of the gradient-based methods were like the EnKF independent of the simulator source code. When using adjoint methods for computation of gradients, access to the source code is needed. This means that implementing gradient-based methods is a choice of long run time or implementing extra features on a reservoir simulator source code basis. The EnKF on the other hand is very easy to adapt to different simulators. There is only a need for an interface to make the filter communicate with the reservoir model. One example of this is described in the implementation part of this paper in Section 7.3.

To summarize this comparison, the traditional gradient-based methods have been around for a long time. Several modifications have been done since the description of these methods in 1972 by Dougherty. The EnKF is a new method which have shown promising results on many reservoir cases. The advantage of the EnKF methods is that it is computationally efficient and easy to implement compared to the traditional methods. It incorporates uncertainty calculation in a smart and understandable way using ensemble statistics. The EnKF also computes the estimates sequentially and updates both the static and dynamic parameters. Which forms an optimal starting point for computing prediction and makes EnKF suitable for operational reservoir monitoring and prediction (V. Haugen et al. [24]).

Chapter 6

Observability and constraint handling for EnKF

OBSERVABILITY is a term often used in control engineering. Observability relates to system identification and deals with whether or not the initial state of a system can be observed from the output. In the linear case it exists a well established method to define observability. The first part of this chapter presents the linear theory and a simple example as a background for understanding nonlinear observability. After that the focus moves to the nonlinear case. Proving global observability for a nonlinear system is difficult, and certainly not practical for large scale systems. A criteria for local nonlinear observability have been developed and a lot of research has been done in this area. In this chapter a definition of nonlinear local observability is given. Further this is related to the EnKF and various problem reducing strategies already implemented in the EnKF are related to improving the observability. The purpose of establishing this connection between observability and EnKF is to outline a different angle to approach the continuous reservoir model updating problem using EnKF. Constraint handling is another term often used in control theory. Constraints appears in many problems. The most common forms of constraints are as limitations in variables, inputs and/or outputs. In addition relationships between these parameters (usually in form of a model or specifications) can act as constraints. An outline of how EnKF is suited for constraint handling will be given last in this chapter.

6.1 Linear observability theory

To start of the linear case is investigated. Goodwin et al. [22] gave the following definition of observability

> The state $x_0 \neq 0$ is said to be "unobservable" if, given $x(0) = x_0$ and $u[k] = 0$ for $k \geq 0$, then $y[k] = 0$ for $k \geq 0$. The system is said to be completely observable if there exists no nonzero initial state that it is unobservable

also an alternative definition is found in Chen [11]

> The state equation (6.1) is said to be observable if for any unknown initial state $\boldsymbol{x}(0)$, there exists a finite $t_1 > 0$ such that the knowledge of the input $\boldsymbol{u}$ and the output $\boldsymbol{y}$ over $[0, t_1]$ suffices to determine uniquely the initial state $\boldsymbol{x}(0)$. Otherwise the equation is said to be unobservable.

Consider the following linear time invariant system

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \tag{6.1a}$$
$$\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{u} \tag{6.1b}$$

and the observability matrix

$$\boldsymbol{Q}_o = \begin{bmatrix} \boldsymbol{C}^T & (\boldsymbol{C}\boldsymbol{A})^T & \dots & (\boldsymbol{C}\boldsymbol{A}^{n-1})^T \end{bmatrix} \tag{6.2}$$

where n is the number of states. A method to check if a linear system is observable is given in Theorem 1.

Theorem 1. *The system* (6.1) *is observable if and only if* $Rank(\boldsymbol{Q}_o) = n$. *Where* $\boldsymbol{Q}_o$ *is defined as* (6.2).

A simple example

To illustrate the definitions a simple example is considered. A car is driving along a straight road. The goal is to observe both the position and velocity, what measurements are needed to determine this? Lets create a simple linear system for the car

$$\begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix} \tag{6.3a}$$
$$\boldsymbol{y} = \begin{bmatrix} a & b \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix} \tag{6.3b}$$

where x is the position and v is the velocity. a and b are two constant used to vary the measurement y. That is if $a = 0, b = 1$ the velocity is measured and if $a = 1, b = 0$ the position is measured. For simplicity the input in this system is not considered and only the relationship between position and velocity is illustrated. Using (6.2) the following observability matrix is obtained

$$\boldsymbol{Q}_o = \begin{bmatrix} a & 0 \\ b & a \end{bmatrix} \tag{6.4}$$

From this one can conclude that if the position or a linear combination of the position and velocity is measured the system is observable. On the other hand if only the velocity is measured the system is not observable. This is pretty intuitive. If the position and time is known, the velocity is easily calculated. On the other hand if the velocity is known one can integrate to get the change in position over the time, but without a position reference somewhere one can not know the exact position.

6.2 Large scale nonlinear observability

So how can the observability defined in the previous section relate to the EnKF and the reservoir estimation case? First consider a general nonlinear problem

$$\dot{\boldsymbol{x}}(t) = f(t, \boldsymbol{x}(t), \boldsymbol{u}) \tag{6.5a}$$
$$\boldsymbol{y} = h(t, \boldsymbol{x}(t)) \tag{6.5b}$$

where $(t, x) \in S \subset \mathcal{R}^1 \times \mathcal{R}^n,\ u \in R_u \subset \mathcal{R}^l,\ y \in R_y \subset \mathcal{R}^m$, and $t \in [t_0, T]$. The observability problem is: under what conditions can $\boldsymbol{x}_0$ be uniquely determined from $\boldsymbol{y}(t)$ and $\boldsymbol{u},\ t \in [t_0, T]$? An overview of different approaches to investigate observability in nonlinear systems have been suggested by Hwang and Seinfeld [27] and more details and references around the theory can be found here. In addition Hedrick and Girard at University of California (Berkley) gave a good summary of nonlinear observability using Lie derivatives in their lecture notes [25]. First they define distinguishability as in Theorem 2.

Theorem 2. *Two states $\boldsymbol{x}_0$ and $\boldsymbol{x}_1$ are distinguishable if there exists an input function $\boldsymbol{u}^*$ such that: $y|_{\boldsymbol{x}_0} \neq y|_{\boldsymbol{x}_1}$*

Afterward the local observability around a state $\boldsymbol{x}_0$ is defined in Theorem 3

Theorem 3. *The system* (6.5) *is locally observable at $\boldsymbol{x}_0$ if there exists a neighbourhood of $\boldsymbol{x}_0$ such that every $\boldsymbol{x}$ in thath neighbourhood other than $\boldsymbol{x}_0$ is distinguishable from $\boldsymbol{x}_0$.*

Given these definitions, a method to test local observability was outlined by Hedrick and Girard. This test involves using Lie derivatives and creating a nonlinear version of an observability matrix. To generate this matrix, one must compute the derivative of $h(t, \boldsymbol{x}(t))$ with respect to both the state $\boldsymbol{x}$ and the input $\boldsymbol{u}$.

It is clear that the method described above becomes unpractical for large scale systems. In the reservoir case using a commercial reservoir simulator it is very difficult to obtain this matrix. But the idea of identifying if the problem is locally observable is something to consider. If the reservoir model is locally observable in an space $E \in \mathcal{R}^{n_x + n_\alpha}$ around the true solution and our initial ensemble does not span outside E it should be possible to find the solution. So the question the becomes, what can one do to ensure observability?

Consider a general problem of the same size as the shoe box problem described earlier. The system has approximately 9000 parameters to be identified using only eight measurements. This alone sounds like an impossible problem and the need for more information about the system is obvious to ensure local observability. So where can one introduce more information to the problem? Some suggestions are made below

- Through the model:
  The model contains relationships between the input,unknown parameters, states and output, hence it limits the degrees of freedom.

- Through excitation of the system:
  Different excitations of the model may trigger different model properties in the measurements. As seen in Theorem 2 and 3 the input $\boldsymbol{u}$ is important for observability. A more complex system input will potentially make more states distinguishable.

- Searching only in a subspace of the total space spanned by the parameters:
  An example of this has been described earlier by the EnKF searching only in the space spanned by the initial ensemble. This is based on some prior knowledge of the system parameters. If a smaller space close to the true solution is searched in, the space is more likely to lie within the locally observable region E.

- Add more measurement:
  More measurements also have the potential to make more states distinguishable according to Theorem 2. However, not all measurements provide more information to the system. A very easy example is that if one measure both the oil rate and the gas rate. Adding a measurement of the GOR does not add more information to the system since this is only a combination of the previous two.

In Figure 6.1 an observability region E is illustrated together with the true solution x_0. The actions previously described tries to either increase E or to search in a smaller region closer to x_0.

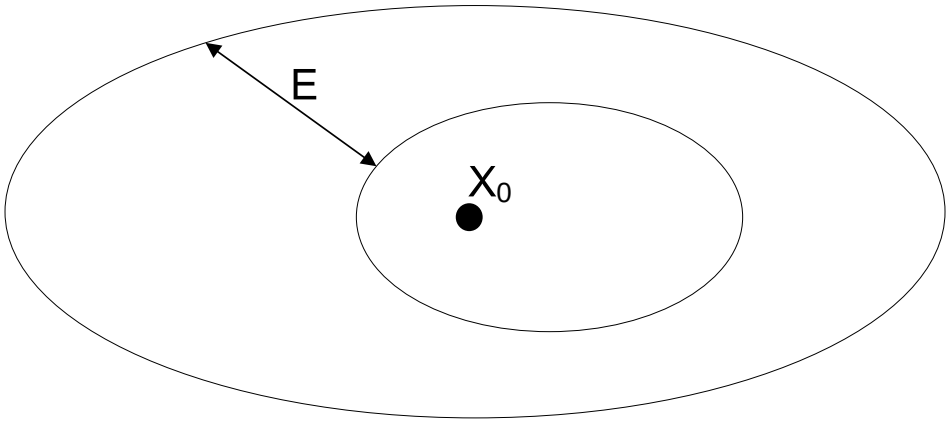


Figure 6.1: Observability region

The EnKF update would be interesting to analyze with regard to observability. The idea is to investigate how the Kalman gain is determined. The Kalman gain decides how much the variables are updated from the information in the measurements. To investigate this the state vector and Kalman gain equation are repeated and analyzed. Remember the ensemble state vector $\boldsymbol{\mathcal{A}}_k^l$ at step k for ensemble member l, which was expressed as follows

$$\boldsymbol{\mathcal{A}}_k^l = \begin{bmatrix} \boldsymbol{\theta}_k^l \\ \boldsymbol{x}_k^l \\ \mathcal{M}(\boldsymbol{x}_k^l, \boldsymbol{\theta}_k^l) \end{bmatrix} \tag{6.6}$$

The EnKF gain is given by

$$\boldsymbol{K}_k = \boldsymbol{P}_k^- \boldsymbol{H}_k^T (\boldsymbol{H}_k \boldsymbol{P}_k^- \boldsymbol{H}_k^T + \boldsymbol{V})^{-1} \tag{6.7}$$

Next the information the different matrices in (6.7) contains is described in (6.8). Note that when writing *state* it is understood the augmented state containing bot the dynamic and the poorly known

static variables ($\boldsymbol{A}_k^l = [\boldsymbol{\theta}_k^l \ \boldsymbol{x}_k^l]^T$ from 3.27 for ensemble member l).

$$\boldsymbol{K}_k = \left[\begin{array}{c|c} \begin{array}{c}\text{Covariance:}\\ \text{state}\\ \updownarrow \\ \text{state}\end{array} & \begin{array}{c}\text{Covariance:}\\ \text{state}\\ \updownarrow \\ \mathcal{M}(\text{state})\end{array} \\ \hline \begin{array}{c}\text{Covariance:}\\ \text{state}\\ \updownarrow \\ \mathcal{M}(\text{state})\end{array} & \begin{array}{c}\text{Covariance:}\\ \mathcal{M}(\text{state})\\ \updownarrow \\ \mathcal{M}(\text{state})\end{array} \end{array}\right] \boldsymbol{H}_k^T \left[\begin{array}{cc} \text{Covariance:} & \text{Covariance:} \\ \mathcal{M}(\text{state}) & \text{y} \\ \updownarrow & \updownarrow \\ \mathcal{M}(\text{state}) & \text{y} \end{array}\right]^{-1} \tag{6.8}$$

So the covariance matrix ($\boldsymbol{P}_k^-$) expresses the covariance for the state ($\boldsymbol{x}$ and $\boldsymbol{\theta}$), between the state and the computed nonlinear measurements ($\mathcal{M}(\boldsymbol{x}_k, \boldsymbol{\theta}_k)$) and for the computed nonlinear measurements. The matrix $(\boldsymbol{H}_k \boldsymbol{P}_k^- \boldsymbol{H}_k^T + \boldsymbol{V})^{-1}$ holds information about the measurements (y) covariance (V) and computed nonlinear measurements covariance. When computing the EnKF gain by (6.8) its content can be interpreted as (6.9)

$$\boldsymbol{K}_k = \left[\begin{array}{c} \begin{array}{c}\text{Covariance:}\\ \text{state}\\ \updownarrow \\ \mathcal{M}(\text{state})\end{array} \left(\begin{array}{cc} \text{Covariance:} & \text{Covariance:} \\ \mathcal{M}(\text{state}) & \text{y} \\ \updownarrow & \updownarrow \\ \mathcal{M}(\text{state}) & \text{y} \end{array}\right)^{-1} \\ \hline \\ \text{measurement gain} \\ \\ \end{array}\right] \tag{6.9}$$

The measurement gain is of no interest since the updated computed measurements are disregarded before the next step. This shows that there are two factors that determine how much each state will be updated by each measurements error.

1. How much correlation there is between the states and the computed nonlinear measurements
2. A factor describing the uncertainty in the model compared to the uncertainty in the measurements

This means that the EnKF will have more update in the states (x and θ) that has the strongest influence on the computed nonlinear measurements, than in the once with less influence. Some regions in the field are more likely to influence the measurements than other. Consider a simple example of the permeability field in a 2-dimensional reservoir with one producer and one injector. Then the area around and between the wells are more important than the area closer to the corners. This can be illustrated as in figure 6.2. The blue area illustrates the area where the permeability is of less importance for the history match. This is of course dependent of the reservoir properties, but in general there are often some areas where the parameters are of less importance than others. This means that one should assume the standard deviation to first decrease in the white are of the figure. It can not be concluded that the blue regions wont be estimated properly since the system is highly nonlinear and information from these parameters may be propagated through several steps of

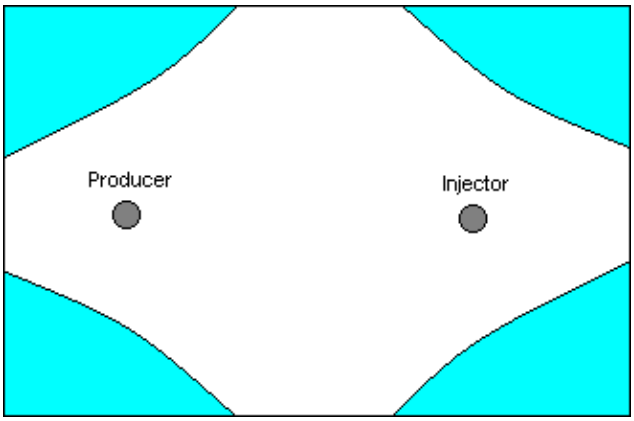


Figure 6.2: Observability region

the EnKF algorithm. This analysis can not conclude whether parameters can be identified or not, but is meant as an illustration to how the EnKF works. It can also give some pointers of where to expect the parameter estimation to be harder.

This section has tried to introduce the strict definition of observability into the parameter estimation problem in reservoir models. Traditionally the success of this problem have been described using the term identifiability. A.T. Watson et al. [47] said the following about identifiability

> Since the number of parameters to be estimated in a reservoir history match is potentially quite large, it is important to determine which parameters can be estimated with reasonable accuracy from the available data. This aspect can be called determining the identifiability of the parameters.

However, Watson et al. also described the desired knowledge regarding identifiability. That is to answer the two following questions

1. Which parameters can be determined uniquely if the measurements were exact?
2. Given the expected level of error in the measurements, how accurately is it expected to be able to estimate the parameters?

The first question is related to a definition similar to that of observability. Watson et al. describes obtaining this information in general as impossible for large scale systems. This relates to the conclusions of the observability analysis in this chapter. However, the observability theory presented in this paper is good guidelines on how to reduce the problem according to both questions presented above.

6.3 Constraint handling

The most common forms of constraints are limitations in variables, inputs and/or outputs. In addition, relationships between these parameters (usually in form of a model or specifications) can act as constraints. In the parameter estimation problem there is mostly one type of constraint that is interesting, that is the constraint in the variables (parameters and states). One of the strengths in traditional history matching using gradient-based method is constraint handling, and how this easily can be implemented in the algorithm. When using genetic algorithms or EnKF constraint handling requires more thought.

The simplest way of enforcing constraints on the static variables in the EnKF is to put upper and lower bounds on them. Then every time the variables are updated the algorithm sets the parameters that break the constraints equal to respectively the upper or lower limit. The question is then how does this interfere with the method. Consider the case of where porosity is assumed to be low, say a mean of 0.1. Let the uncertainty in this parameter be large, for instance 0.05. The probability density function (pdf) in this parameter will then be as illustrated in Figure 6.3a and the pdf with a lower bound at 0 will be as in Figure 6.3b. This can then relate to the discussion in Section

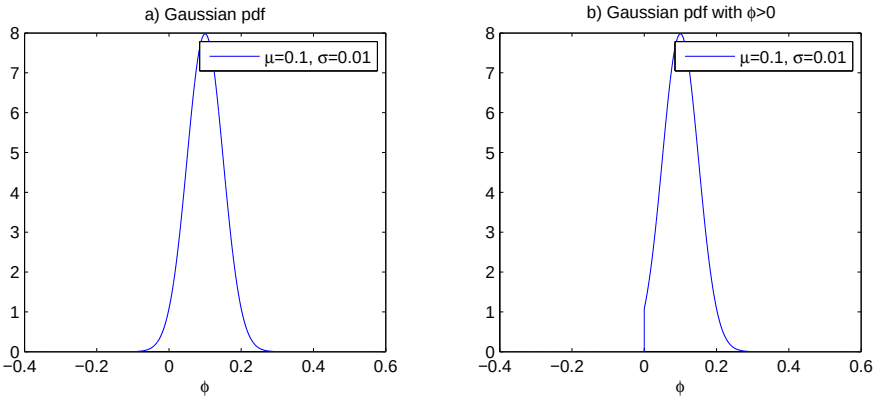


Figure 6.3: Probability density function with (a) and without (b) constraint

3.3.5 since the distribution is no longer Gaussian. However, it is assumed to give sufficient results for the EnKF on a reservoir case. In general constraint handling will be complex to implement correctly in EnKF and only an approximate implementation for constraints on the parameters has been suggested.

Constraint handling is also interesting when coupling EnKF with optimal reservoir control. This will be discussed further in the Chapter 8 were some challenges regarding this are posed.

Chapter 7

EnKF applied on a simple reservoir model

One of the goals for this paper is to implement and test the EnKF as a parameter and state estimator for a reservoir model. For this paper a simple shoe box reservoir model was used to test the EnKF method. The EnKF algorithm used was made by the International Research Institute of Stavanger (IRIS). The purpose of this section is to investigate the properties of the EnKF on a large scale 3-dimensional reservoir case. To start of a short description of the reservoir model is given. Afterward the implementation of EnKF is discussed before the various simulations are outlined and the results are presented and discussed.

7.1 Model description

The reservoir model used in this paper is implemented in the commercial oil reservoir simulator ECLIPSE [42] developed by Schlumberger. The model is a simple, cubic, shoe box model which is 2-phase and consists of only water and oil. Two high permeable layers are separated by one with a lower permeability making the water flooding time vary in the different layers. The model is also created with a general flow pattern in each layer going from x, y grid block $(1, 1)$ to $(15, 15)$ as illustrated in figure 7.1. The reservoir geometry is illustrated in Figure 7.2

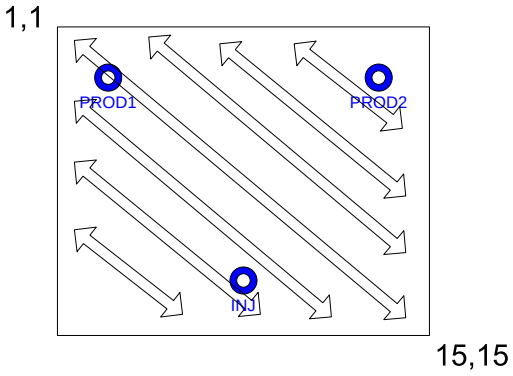


Figure 7.1: The shoe box reservoir flow pattern

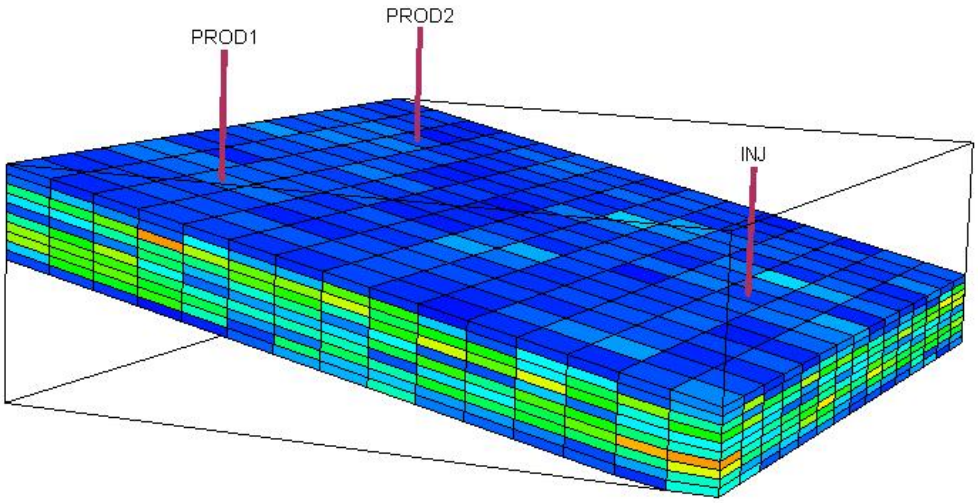


Figure 7.2: The shoe box reservoir layout

and the number of grid cells is found by

$$\bar{m} \times \bar{n} \times \bar{k} = 15 \times 15 \times 10 = 2250 \tag{7.1}$$

which yields 4500 pressure and saturation variables. A more detailed description of the reservoir geometry is given in Table 7.1. The producers and injectors are located in the reservoir as shown in

Variable	Description	Value	Units
$\bar{m}$, $\bar{n}$	grid blocks in x and y direction	15	-
$\bar{o}$	grid blocks in z direction	10	-
Δx	grid block width in x direction	94	m
Δy	grid block width in y direction	80	m
Δz	grid block heigth	5.6	m
$P_{o,init}$	initial pressure	260	bar

Table 7.1: Shoe box reservoir properties

Table 7.2. The reservoir is set to produce with a liquid rate at 2000 Sm^3/d and an upper bound for

Well	Description	x	y	z
PROD1	Producing well 1	4	13	[2 . . . 7]
PROD2	Producing well 2	12	13	[2 . . . 7]
INJ	Injector well	8	3	[3 . . . 8]

Table 7.2: Shoe box well placements

the oil rate at 1500 Sm^3/d for each producer. The BHP is bounded below at 100 bar. For the water injection, the rate is controlled at 4000 Sm^3/d with an upper BHP limit at 400bar. The resulting rates and pressures from the simulation of the shoe box reservoir is illustrated in Figure 7.3. Here it

is observed that both wells have their water breakthrough at around 500 days of production. Before the wells produce water there is pressure buildup in the wells and when the wells start producing water the pressure drops. The model is set up with 7 measurements for the whole field and 8

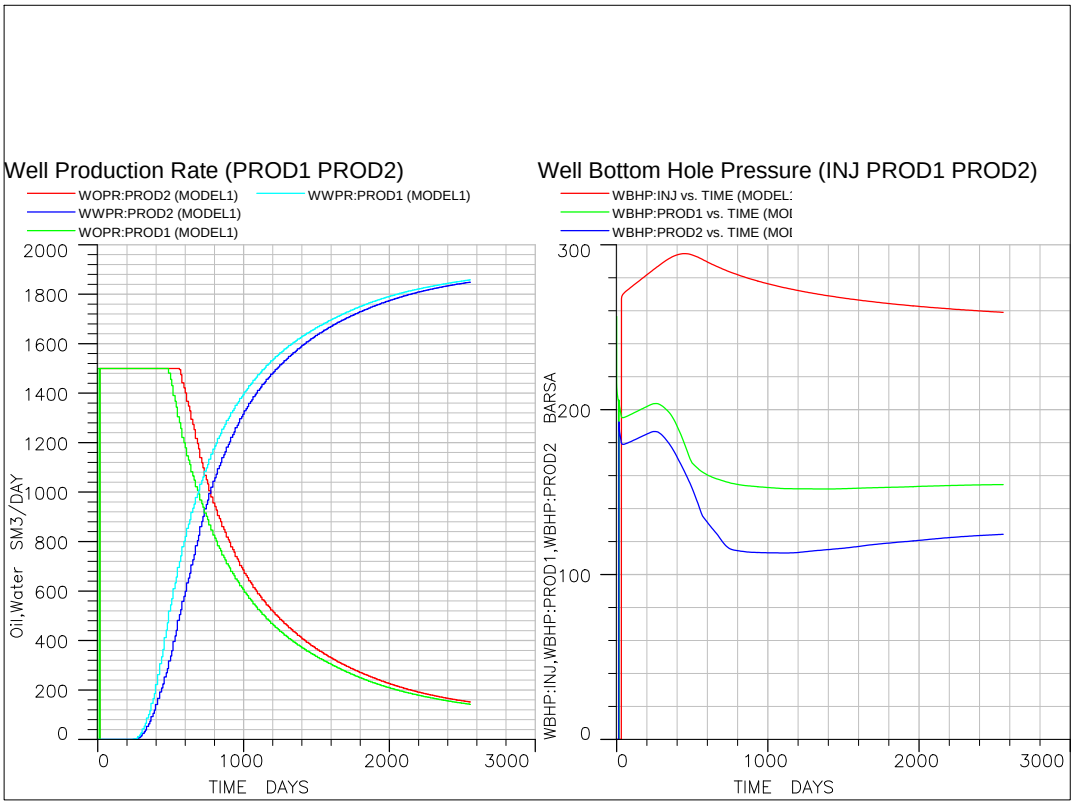


Figure 7.3: The shoe box rates and BHP

measurements that are on a well basis. The measurements available are described in more detail in Table 7.3. For the well measurements the wells that have the specific measurement is listed in parenthesis in the description. This model is the basis for the implementation and testing of the EnKF. Now that the model is available the focus moves to the problem specification.

Measurement	Description	Unit
WWIR	Well Water Injection Rate (INJ)	Sm^3/d
WWIT	Well Water Injection Total (INJ)	Sm^3
WOPR	Well Oil Production Rate (PROD1, PROD2)	Sm^3/d
WOPT	Well Oil Production Total (PROD1, PROD2)	Sm^3
WWPR	Well Water Production Rate (PROD1, PROD2)	Sm^3/d
WWPT	Well Water Production Total (PROD1, PROD2)	Sm^3
WBHP	Well Bottom Hole Pressure (PROD1, PROD2, INJ)	bar
WWCT	Well Water Cut (PROD1, PROD2)	-
FOPR	Field Oil Production Rate	Sm^3/d
FOPT	Field Oil Production Total	Sm^3
FWPR	Field Water Production Rate	Sm^3/d
FWPT	Field Water Production Total	Sm^3
FWCT	Field Water Cut	-
FWIR	Field Water Injection Rate	Sm^3/d
FWIT	Field Water Injection Total	Sm^3

Table 7.3: Shoe box reservoir measurements

7.2 Problem specification

A simple shoe box reservoir model has been outlined for these tests. The true solution measurements is then computed using the real porosity and permeability parameters. The true parameters is shown in Appendix C. Then the EnKF is initialized with an ensemble of static variables and history matched with the true solution. These tests assumes that there is no model error. The testing of systems with model error is left out for further work, see Section 10.1.

In this paper the focus is on estimating the porosity and permeability fields. The filter estimates the full 3-dimensional permeability field ($k_h(\boldsymbol{z})$), which is permeability in x-, y- and z-direction for each grid block. Also the porosity for each cell ($\phi(\boldsymbol{z})$) is estimated. In the shoe box model the dynamic parameters are pressure ($P(\boldsymbol{z})$) and water saturation ($S_w(\boldsymbol{z})$). Given these variables the augmented state vector can be derived as follows

$$\begin{bmatrix} \boldsymbol{P} \\ \boldsymbol{S}_w \\ \boldsymbol{k}_h \\ \boldsymbol{\phi} \end{bmatrix}_j \tag{7.2}$$

Here it can be seen that both the dynamic and static parameters can be updated in the augmented state vector. In the test case used in this paper only the static parameters are initialized with uncertainty in the EnKF. This is because it is assumed that there is no model error and therefore the covariance matrix for the states equals zero. The assumption of no model error is due to the fact that the true solution is computed with the same model as the EnKF uses. So in practice, the filter only searches for the static parameters.

There are a total of 6 variables to be estimated in each cell of the reservoir model. The porosity and permeability together count for 4 parameters in each grid block and the saturation and pressure count for the other two. Thus there are 9000 unknowns in the permeability and porosity fields, which count for many degrees of freedom (DOF).

In the shoe box model the number of DOF can be reduced by assuming these fields to be smooth, so that each grid node no longer consists of totally independent variables. This can be done by specifying correlation in horizontal and vertical direction in the prior statistics. Initially the correlation was specified to be 5 cells in each direction. This parameter is subject to tuning parameter and its properties will be discussed in more detail later.

Another specification that has to be done for the EnKF problem is the measurements. For the shoe box reservoir model the following measurements are used in the EnKF filtering equation.

- PROD1 - WOPR
- PROD1 - WWPR
- PROD1 - WBHP
- PROD2 - WOPR
- PROD2 - WWPR
- PROD2 - WBHP
- INJ - WWIR
- INJ - WBHP

This gives us eight measurements in total, where five are rate and three are pressure measurements. These measurements are subject to noise in a real case and an artificial noise is added to the measurements. For this case the noise in the rates have a variance of 100 Sm^3/day and the pressures 15 bar. Given this specification it is time to consider the implementation of the EnKF and how this cooperates with the model.

7.3 Implementation

Creating a full scale system for an EnKF on a reservoir model is a big task. In this project a Matlab EnKF implementation with connection to the reservoir simulator ECLIPSE was used. It was developed at the International Research Institute of Stavanger (IRIS). A big part of the implementing EnKF on the "shoe box" reservoir was examining this code and verifying it. The IRIS code consists of many options and settings to tweak the performance for different scenarios. A short description of the Code will be given in this Section. More info can also be found in the manual [46].

As mentioned earlier one of the benefits of using EnKF is that it is independent of the reservoir simulator. Still the EnKF code needs some sort of mechanism to communicate with the simulator. In this case the commercial simulator ECLIPSE was used. So for the EnKF to work an interface with ECLIPSE is needed. The main file describing an ECLIPSE model is the `*.DATA` file containing the model setup. This file can include others using the `INCLUDE` keyword. The data file consists of the following sections.

- `RUNSPEC`
- `GRID`
- `EDIT`
- `PROPS`

- `REGIONS`
- `SOLUTION`
- `SUMMARY`
- `SCHEDULE`

The sections that are critical for EnKF to work are the `GRID`, `SUMMARY` and `SCHEDULE` sections. The `GRID` section is where the static variables in the reservoir model is stored. The EnKF will need to update these parameters according to the assimilation results as it proceeds. This has to be done by writing text files and including them in the `*.DATA` file. The `SUMMARY` section is where the measurements are defined. Here the output from the reservoir simulator is specified from a long list of measurements and properties for the wells, the groups or the whole field available. The `SCHEDULE` section is where the report dates for the simulator is specified.

When the simulator runs it generates restart files at the end of each section. The EnKF code need to be able to:

1. Read these restart files
2. Compute the updated states and static variables
3. Update the restart files with the filtered data.

This also requires an interface to read and write to the output files from the simulator and requires extensive knowledge of the output formats. When an interface for communication has been built, one can focus on the implementation of the filter. As mentioned earlier a simple linear EnKF has been proposed in Appendix A. When transferring the simple example to a filter that handle large problems some new problems arise. The two first an most obvious things is memory management and simulation time. When working with big data sets it is important to keep only the needed variables in the memory and to do the computations efficiently. A feature that is useful is the ability to restart if something unexpected happened or a simulation for some reason crashed. All these implications of introducing large scale data needs to be kept in mind while creating the filter, which leads to a more extensive code.

The filter is implemented according to Algorithm 3, but built into a superstructure containing more functionality. The prediction step contains the communication with ECLIPSE and the functionality described above is realized. In addition the implementation for adding model noise is added. The construction of noise will be addressed later. In the filtering step the measurements are first computed by reading the true solution. The measurement perturbation is then added using a standard Gaussian random generator. The computation of the covariances and Kalman gain is straight forward matrix operations. When updating the ensemble one have to retrieve the nonlinear measurements from the model. If one choose the formulation suggested in (3.31) the recovery of the measurements is just a matter of picking out the right variables from the state vector.

Looking back at the generation of noise. In Section 5.2 a strategy for decreasing the degrees of freedom in the solution space was suggested. This involves adding correlation in the noise fields created for the static variables. The noise generator used in this implementation is built to generate a random 3-dimensional field satisfying a Gaussian variogram in three dimensions. Basically this is to generate normal distributed stochastic variables with covariance matrix $\boldsymbol{Q}_k$. Evensen [16] described one algorithm for creating a variogram.

Other functionalities are also available in the code. Some of these are listed below:

- Specifying upper and lower bounds on parameters
- Specifying if model noise is to be added prior to or after the simulation - Argument for adding noise prior to the simulation is to dampen the noise in the relationship between the static parameters and measurements. As described earlier, the experiments in this paper will not consider the usage of model noise.
- Choice of using log or true value of the permeability - The logarithm of the permeability is often used since the assumption of Gaussian distribution becomes more appropriate.

Given this implementation the setup of the different simulations is the next focus of the next section.

7.4 Simulations

In this paper three strategies for selecting the initial ensemble was investigated. To test the strategies a simulation of the ECLIPSE model containing the true parameters were used as production data. The production data was subject to noise as described in Section 7.1. Afterward the initialization of the ensemble was made and the EnKF applied. The results of the filter were then compared to the true parameters and production data.

The different initializations are presented below. A visualization of these initializations is illustrated in appendix D, showing the permeability and porosity fields and their respective standard deviation. The difference in the initializations is in how they define the static variables in the initial ensemble. The dynamic parameters (pressure and saturation) are initialized as the true values without any uncertainties.

Initialization 1 - True mean

In this realization the static variables in the initial ensemble is defined with a mean that is equal to that of the true static parameters and a standard deviation of 0.1 for the porosity and 0.5 for the permeability (log). The initial ensemble should then span the solution. The correlation length was initialized with a mean of $[5, 5, 5]$ cells in the $[x, y, z]$ direction and standard deviation of 1 cell. This will initially give a scattered ensemble but spanned around the true solution.

Initialization 2 - Flat mean

Here the initial ensemble is defined flat in each layer. Where the mean and standard deviation is based on measurements made in the wells. The mean for each layer is the mean of the measurements at each well in that layer and the standard deviation is based on the variation in these measurements. The correlations are specified as in Initialization 1.

Initialization 3 - Offset mean

This initialization is based on the true static variables, but the mean here has an offset of respectively 0.05 and 0.2 for the porosity and permeability. The standard deviations and the correlations are specified as in Initialization 1.

The simulation tests

These three initializations was used as a basis in the tests in this paper. The first three tests that were carried out are

Simulation 1: Simulation with initialization 1
As described above

Simulation 2: Simulation with initialization 2
As described above

Simulation 3: Simulation with initialization 3
As described above

When the results from these three simulations were analyzed it was clear that the correlation lengths were wrong compared to the true case. This will be discussed in the next section. The idea was then to run the same Initialization 2 over again, but with a correlation length closer to that of the true solution. The following simulation was then done.

Simulation 4: Testing of the effect of the correlation:
Initialization as in Simulation 2, but correlation with a mean of $[2, 2, 0]$ cells in the $[x, y, z]$ direction and standard deviation of 2 cells.

A presentation of the results will be given in the next section together with a discussion on some of the observations.

7.5 Results

The three original simulations (1-3) was first conducted, and the results from these simulations are presented. Afterward the next simulation (4) was run and the results presented. The history matching results from these simulations can be found in Appendix E. In these plots the red line represents the true solution. The green lines represent the simulation of the model with the initial static variables. The blue lines is the filtered static variables, after the EnKF was run over the time horizon, used in the model for the whole horizon. Twenty ensemble members was chosen for these plots. In addition some of the static parameters will be presented in this chapter. The full simulation results containing both the history matching and the static parameters can be found in the Electronic Appendix, see Appendix G. In this appendix the static parameter is shown with the evolution of the mean, the standard deviation and two ensemble members. These static variables plots are shown for 5 steps in time and in addition the true solution is shown.

Simulation 1-3

To compare the history match in more detail lets consider the water production rate (WWPR) from PROD1 and PROD2 for each of the three first simulations in Figure 7.4. 20 ensemble members are

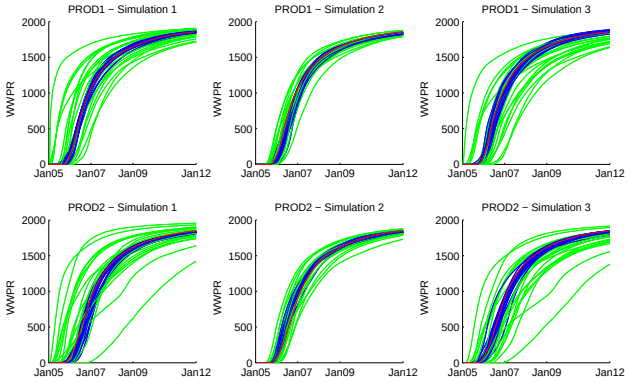


Figure 7.4: Well water production rate for simulation 1,2 and 3

plotted with the initial ensemble parameters (in greeen) and for the assimilated parameters after the EnKF was run over the period (in blue). Here it is seen that the WWPR uncertainties from the initial ensemble span the true solution. It can also be observed that the WWPR for the ensemble after the 7 years of simulation has a lot less uncertainty and fits the true solution better. This result is common for all of the three simulations. Where in general the EnKF has produced new realizations of the model which produce a better history match for the WWPR with less uncertainty.

The wells bottom hole pressure (WBHP) is another measurement to consider. This measurement is investigated further in Figure 7.5 for both the two producers and the injection well for simulation 1-3. Here it is also observed that the updated parameters in blue provides better results than the initial in green.

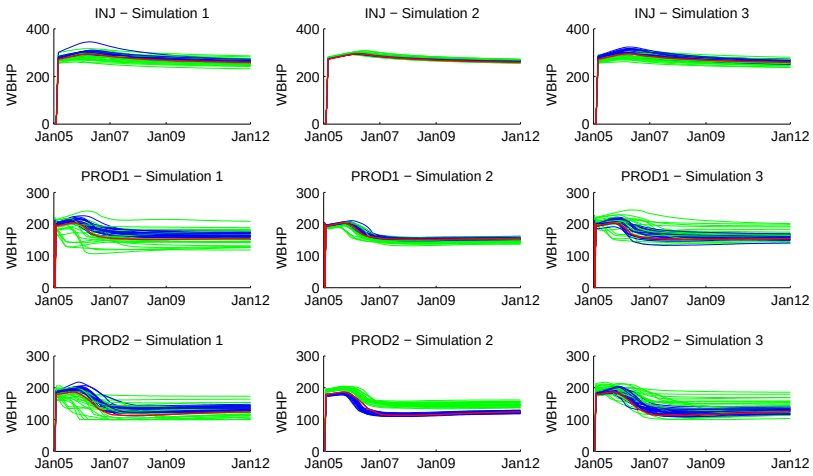


Figure 7.5: Well bottom hole pressure for simulation 1,2 and 3

There are however a couple of exceptions. In simulation 1 ensemble number 11 overestimates the BHP in INJ, which can be seen by the blue line lying above the rest. This is made up for in the EnKF by updating the states to get a better history match. Since the EnKF can both update the static and dynamic parameters it somehow fails for this ensemble and chooses to update the state when the error lies in the static variables. The updates made to the pressure states can be seen in Figure 7.6. The plot to the left shows the history match plot, in the middle the average

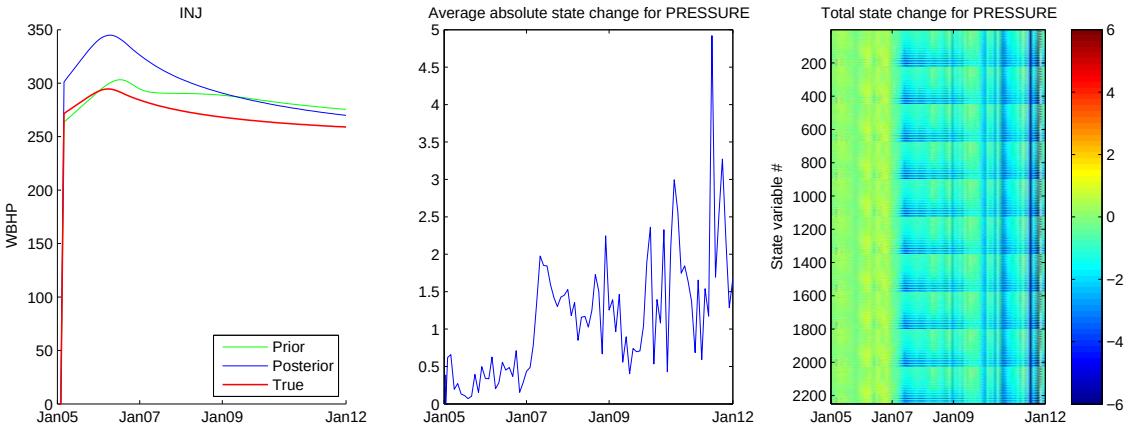


Figure 7.6: Simulation 1 - Changes made to the Pressure in ensemble member 11

absolute change in the pressure state is shown and to the right the changes for all variables over the whole period. The same plot can be made for ensemble member 1 which has much better history matching. In Figure 7.7 one can see that less adjustments to the state on average is done. The last

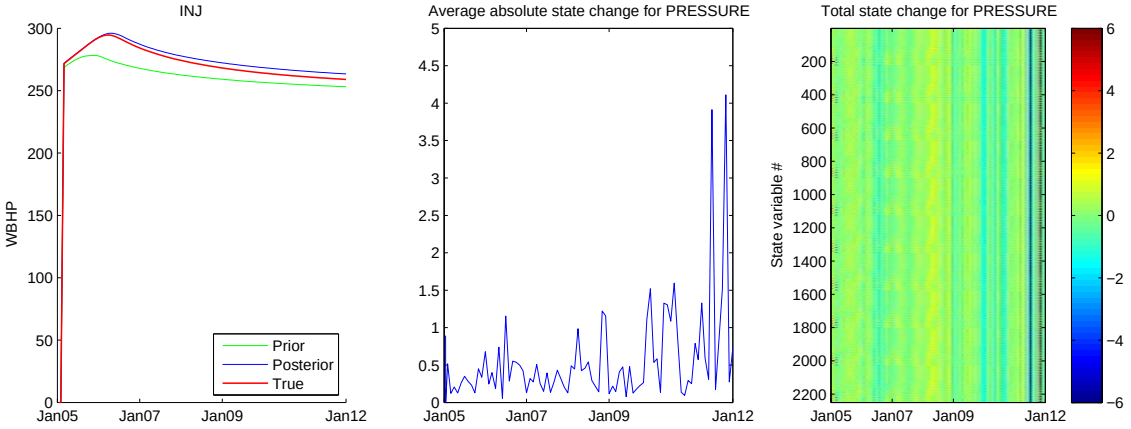


Figure 7.7: Simulation 1 - Changes made to the Pressure in ensemble member 1

two years one observe an big increase in the update done in the pressure variables. This is due to the fact that the standard deviation in the static parameters increases over time. That effect will be explained later in this section.

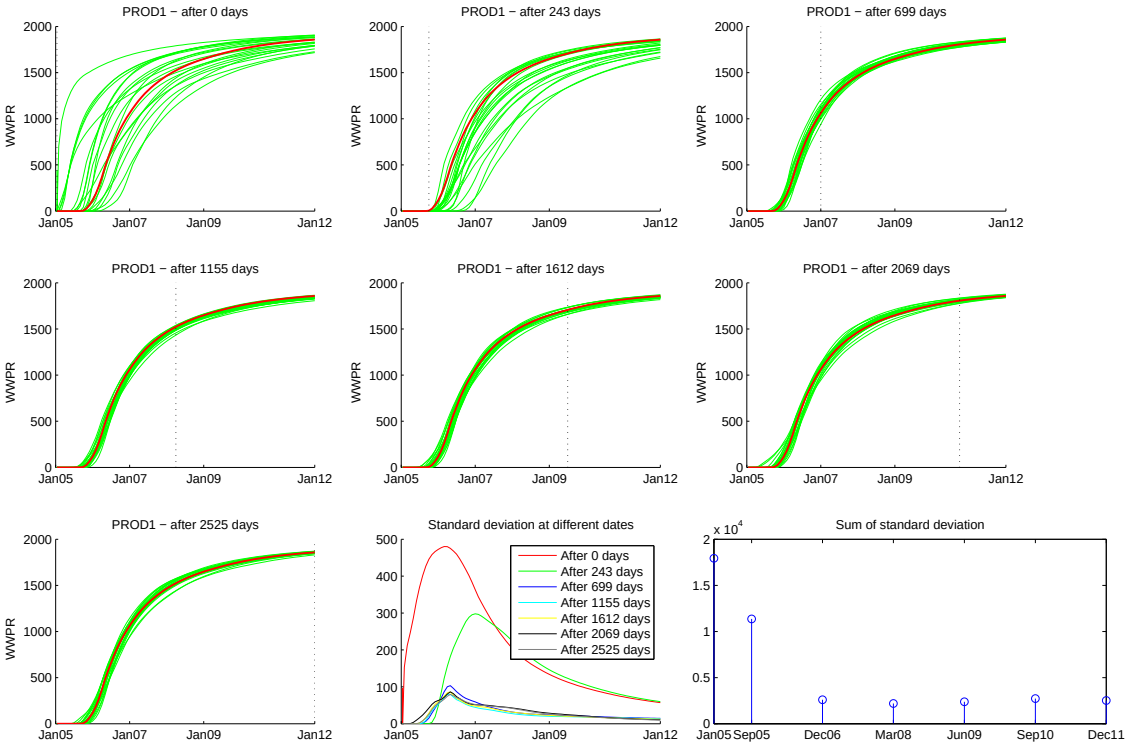


Figure 7.8: Simulation 1 - Uncertainty in the WWPR

In addition the BHP in PROD2 from Simulation 2 in Figure 7.5 shows another interesting result. Here the initial ensemble overestimates the bottom hole pressure. From the first look one can ask, why does this plot converge toward the true solution when the initial ensemble does not span the solution? The reason why offsets in the results can be detected and dealt with is that a better history match might be spanned in the static parameters even though it is not spanned in the measurements.

Lets consider where the information about the system is obtained. To investigate this the water production rate for PROD1 is examined in Figure 7.8. The first 7 plots shows the true WWPR in red and the WWPR for 20 ensemble members in green. The simulation was run with the parameters acquired after respectively 0, 243, 699, 1155, 1612, 2069 and 2525 days of running the EnKF, illustrated by black dotted line in the plots. Here it is observed that most of the information is obtained before and under the water breakthrough. After a period without much change in the uncertainty, it is a little strange that the uncertainty increases a little in the last period. This will be discussed further when looking at the static variables. Another interesting perspective on this plot is the predictive abilities of the EnKF. From such a simple example with so few measurements the EnKF is not able to predict the water breakthrough with much certainty before it actually happens. To achieve this more measurements would be needed.

After looking at some of the history matching for these three simulations it is time to look at the static variables. The complete results are found in the Electronic Appendix. In this section some interesting parameters and their results are considered. The following discussion is based on the permeabilities in the x-direction in layer 3, similar results as these was acquired for the other permeabilities and porosities in the different layers. The permeability evolution for PERMX in layer 3 from simulation 1-3 is shown in respectively Figure F.1, F.2 and F.3 in Appendix F. In general

one can observe that for Simulation 1 and 3 the mean is a good estimate and the standard deviation decreases in the first period of the simulation (i.e. from 13 days to 608 days). In the later part of the simulation the standard deviations increases a little in the edges of the layer. However for Simulation 2 the standard deviation increases over the whole period. Some details on these results are now investigated.

First consider Simulation 1 and 3 where one can observe that the standard deviation in figure F.1(b) and F.3(b) decreases during the time of the water breakthrough. What is strange on the other hand is that the standard deviation in the ensemble later increases in some parts. Figure 7.9 shows the average standard deviation for both the porosity and the permeability for Simulation 1. It

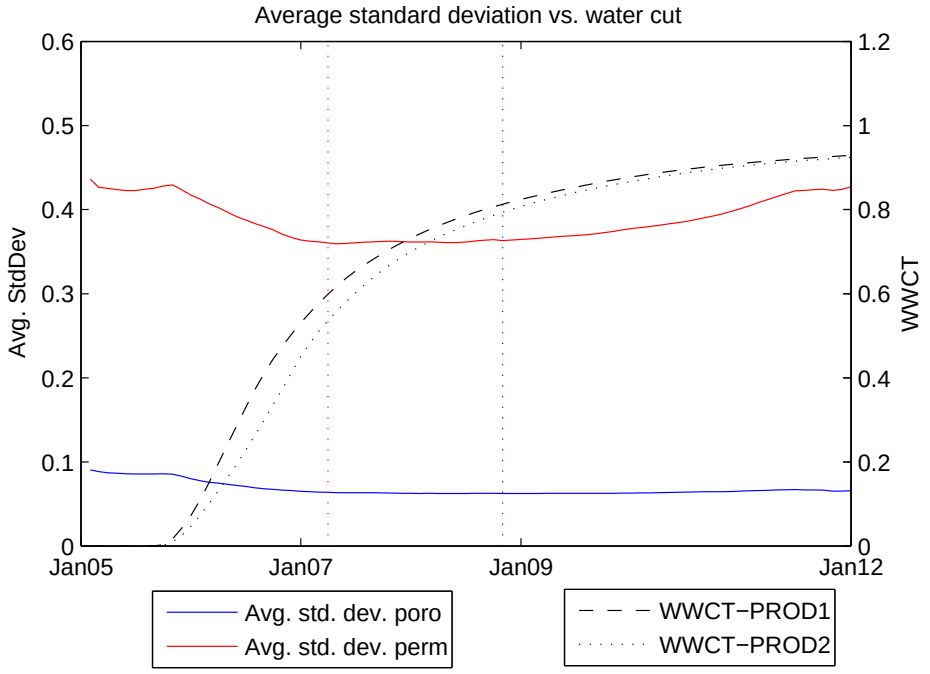


Figure 7.9: Simulation 1 - Changes in the standard deviation

can be seen that on average there is a positive influence on the variance in the static variables from when the water breakthrough occurs and until about midway through the simulation period. On first though, one assumes that when measurements are assimilated using the EnKF, the standard deviation should not increase. But it might fluctuate a little due to measurement noise. What is seen in the simulations is that the uncertainty increases when there is little or no information in the measurements, this will be discussed at the end of this chapter. It is also observed that with the excitation used in these simulations most of the dynamic information is in the water breakthrough. The similar plot for Simulation 3 can be found in Figure F.6 in Appendix F. The results for Simulation 3 shows approximately the same results as for Simulation 1. The average standard deviation only gives a simplified picture, since it says that the total change in standard deviation is very small. There are some regions of the layers that have become more certain (in terms of ensemble standard deviation) while other have become less. The last analysis is to inspect the mean of the ensemble, which represents the most probable parameters according to the EnKF estimation. For Simulation 1 the mean in Figure F.1(a) stays more or less the same for the first period, which is analogous with the initial mean being correct. In Simulation 3 the mean in Figure F.3(a) is initialized a little to

high, but as time proceeds the mean value decreases to something similar of the true solution. For both Simulations the evolution of the mean drifts off toward the end of the simulation period. This is caused by the same effect that affect the standard deviation as mentioned earlier.

When considering Simulation 2 the standard deviation increases during the water breakthrough as observed in F.1(b) and F.5. Also the mean values of the ensemble static variables shows something which totally different from the true solution. This shows that here the EnKF fails completely.

An attempt to explain why these results was obtained will now be given. In general one can say that the EnKF does a fairly good job for Simulation 1 and 3, but suffers from an increase in uncertainty toward the end. Simulation 2 on the other hand seems to fail completely and no valid information can be found in either the mean or the ensemble members. When looking at the different ensemble members and how they evolve under the update it is interesting to see the smoothness in the solution. Especially this can be observed in Simulation 2 which is initialized with a flat mean, while in Simulation 1 and 3 this is suppressed by having the original smoothness in the mean. This is illustrated in in Figure 7.10, where one can observe the initialization in Simulation 2 for two of the ensemble members together with the true value all for PERMX in layer 3. If the

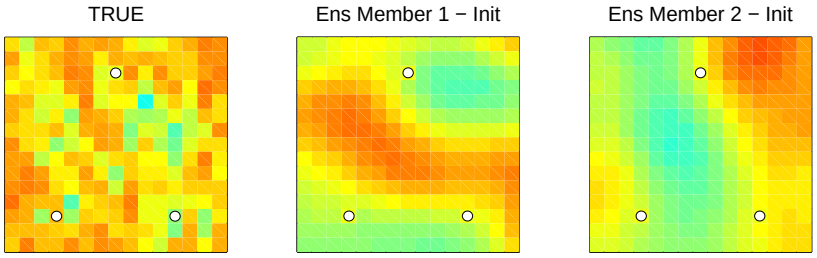


Figure 7.10: Simulation 2 - Initialization in 2 ensemble members

filter correlation is far of from the true solution, the true solution might not be spanned in the initial ensemble. So for these three initializations the correlation has been initialized to high and the filter has problems converging toward the true solution, especially when the initial ensemble does not contain any information about the correlations in the true system (Simulation 2).

Simulation 4

As outlined after discussing the results from Simulation 1, 2 and 3 the conclusion was that the correlation length was to long when initialized with a mean of $[5, 5, 5]$ in the $[x, y, z]$ direction and a standard deviation of 1. By looking at the true solution the correlation is a value closer to a mean of $[2, 2, 0]$ with a standard deviation of 2. This was then used to create the initial ensemble in Initialization 2. A similar simulation was also done for Initialization 1 and 3, but the results where not so different from the previous simulations. This is as expected since the correct correlation was present in their initial ensembles in Simulation 1 and 3. These results will not be presented in the report, but for reference they are available in the Electronic Appendix.

In Simulation 4 a flat mean was used as initialization and the new correlation length was applied when creating the initial ensemble of static variables. This simulation crashed after running approximately 2/3 of the time. The parameter estimation results are shown in Figure F.4. Here it is observed that both the mean and the ensemble members diverge. This is illustrated in Figure 7.11 where the standard deviation change for porosity and permeability is plotted Here the spread

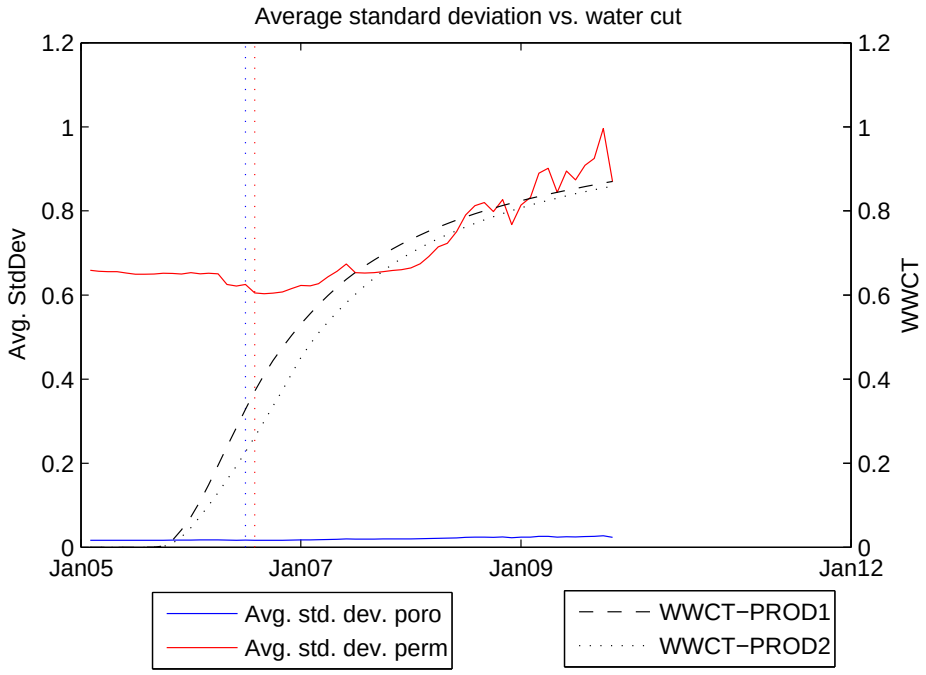


Figure 7.11: Simulation 4 - Changes in the standard deviation

in the ensemble decreases a little as effect of the water breakthrough but after some time it starts to increase rapidly.

The history match uncertainty for PROD1 water production rate as the filter proceeds is presented in figure 7.12 Here one can see that the uncertainty in the water production rate decreases as the EnKF assimilates new data. This is only plotted for the first 1155 days since the history matches after the deviation in the static variables became to large are infeasible. One can also observe the history matching for the other measurements in the Electronic Appendix. Note that the history match is made from the parameters assimilated after 1155 days which corresponds to the March 1st 2008.

The reason for the behavior in Simulation 4 is among others that the solution space is to large and there is not enough information in the measurements, hence the system is not observable. One reason for the observed behavior might be that the covariance matrix estimate is very poor. In Section 6.2 a discussion is made on the EnKF update and more specific on the Kalman gain in regards to parameter update. Recalling from (6.9), one can imagine a case where the covariance between the state and the computed measurement will be completely wrong. This will happen if the ensemble parameters are very different from each other, while the computed measurements give approximately the same solution.

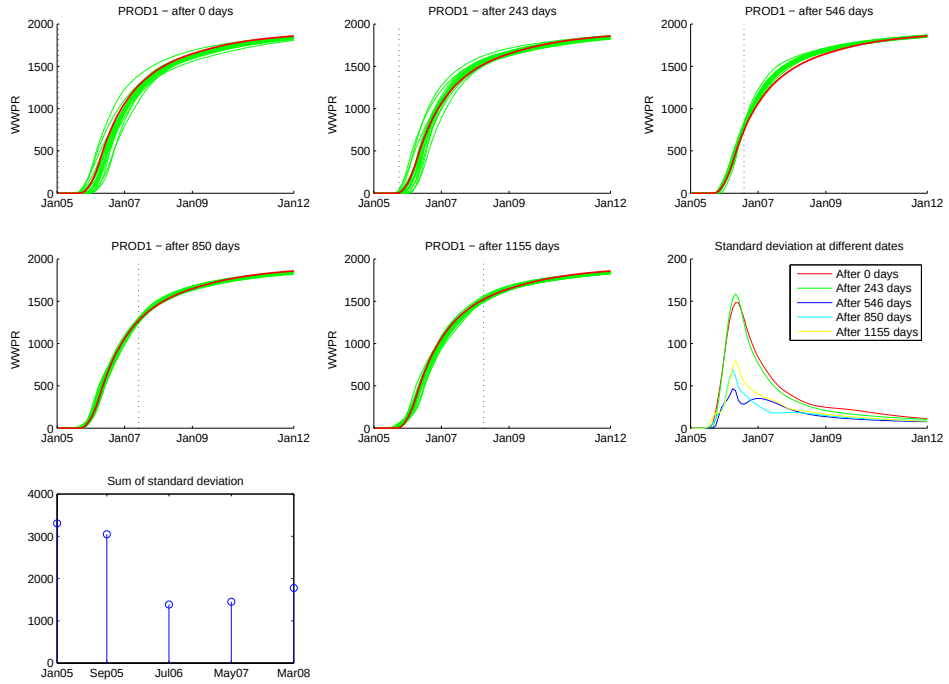


Figure 7.12: Simulation 1 - Uncertainty in the WWPR

In the end, one problem has not been addressed yet. That is the drift in the static parameters when there is no more incremental information in the measurements. As described earlier most of the information in this simple example lies in the water breakthrough. Some information might also be available through pressure measurements, but due to very few measurements this information is assumed to be limited compared to that of the water breakthrough. Similar results have been experienced by G. Nævdal et al. [7][37]. They describe this uncertainty to be a result of the uncertainty in (3.22) being propagated into the system. This often happens near the edges of the reservoir were there are little information available regarding the parameters (See Figure 6.2). They assume the model error to dominate the information obtained from the measurements. Also here the discussion on the Kalman gain can be incorporated. Again consider (6.9), but this time the focus is on the measurement noise. Then, in general, the noise added in (3.22) should be dampened by the specified measurement covariance $\boldsymbol{V}$ in the measurements. In general one should wish that the parameters in the system would not become more uncertain when no data is present. Especially in a case with no model error, however one can imagine that some of this could be compensated for by different tuning of the measurement covariance matrix. A more thorough investigation of this is however left out for further work.

Chapter 8

EnKF and Nonlinear Model Predictive Control

OPTIMAL control of reservoir production is an area which currently recieves a lot of attention. In this chapter a strategy for combining EnKF with Nonlinear Model Predictive Control (NMPC) is outlined and some issues regarding the strategy is presented. This chapter has been written in cooperation with the fellow Msc. student P. Meum and similar work is found in his Thesis [34] on optimal reservoir control. To start of an overview of the problem is given as a quick introduction to the optimal reservoir control problem. Afterward an outline of the system combining EnKF with NMPC is given. Finally some challenges and potential problems are outlined. The work in this Chapter is strictly theoretical and an implementation of such a solution is left for further work.

8.1 Problem overview

The target of the optimal reservoir control problem is to maximize the total production from a field. To achieve the wanted production it is necessary to replace the produced fluids with another fluid to maintain pressure support in the reservoir. In most cases this is done by injection of some of some other fluid. This can typically be non-profitable gas or water from the sea above. When producing from a reservoir, using such a strategy, the replacement fluid pushes the desirable fluid toward the producing end. This is illustrated in Figure 8.1 found in D.R. Brouwer and J.-D. Jansen [6], where a good description of dynamic optimization of the reservoir also can be found.

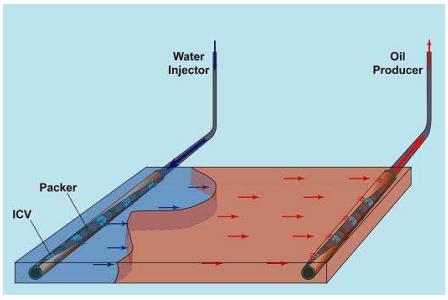


Figure 8.1: Illustration of reservoir flooding

In this figure the water oil contact is illustrated as the water replaces the oil from left to right in a simple two-phase reservoir. In general one can say that to achieve maximum recovery, it is crucial to flood the reservoir in a way that ensures a contact front which is as smooth as possible. This causes the most simultaneous breakthrough of the injected fluid for all the producing wells. If the surface is less smooth the breakthrough might happen at an early stage in a few wells, which is unfortunate for the production.

The general reservoir optimization problem is also subject to constraints, which can be both linear and nonlinear. Examples of constraints spans from restrictions in a controllable valve or bottom hole pressure (BHP) to a maximum flow constraint. In general these problems narrows down to solving a constrained nonlinear minimization problem. B. Sudaryanto and Y.C. Yortsos[45] were in 2000 some of the first to systematically address the problem. They considered a 2-dimensional geometry model with a single producer and multiple injectors. N. Dolle et al [13] investigated dynamic optimization using a gradient-based method on a 2-dimensional reservoir in 2002. Brouwer and Jansen took their work further and investigated the difference between rate and BHP constraints on a 2-dimensional reservoir as illustrated in Figure 8.1. In 2002 Yeten et al. [49] took the optimization problem further by utilizing features available in the commercial reservoir simulator ECLIPSE [42]. Costly computation time was justified with improved prediction reliability. In 2006 P. Sarma et al. [40] presented a gradient-based optimization scheme with improvements in computational efficiency. They used an in-house simulator facility at Standford University with adjoint models directly from the simulator.

B. Yeten et al. in 2002 introduced an aspect of model uncertainty by considering algorithm performance for a number of slightly different models of the same characteristics and size. But all of the work in the introduction given above assumes to have an exact model. The solution of the optimal reservoir control problem is based on the available reservoir information in the model. Before the production starts this is based on seismic data, well tests etc and there are large uncertainties associated with this data. In general the predictive value of such models is limited and tends to deteriorate over time. In addition will prediction error in the model possibly lead to violation in constraints when using optimal control. Hence, for closed-loop reservoir control with model miss match, it will be necessary to include a model updating scheme to achieve satisfying results. This was recognized by D.R. Brouwer et al. [7]. Continuous model updating is to use all available data to update the model and improve the models predictive abilities. In general this has been difficult to implement using traditional history matching. According to D.R. Brouwer et al. traditional history suffers from a number of drawbacks:

1. Usually performed on a campaign basis, typically after periods of years
2. The techniques used are usually ad-hoc and involve manual involvement
3. Uncertainties are usually not explicitly taken into account
4. The resulting history-matched models often violated essential geological constraints
5. The updated model may reproduce the production data accurately but still have no predictive ability

The EnKF as presented in this paper contain functionality which makes it more suitable for use with optimal reservoir control. The EnKF is designed such that it should be able to overcome a number of the shortcomings of traditional history matching. In this section the optimal reservoir control problem has been outlined. Next a suggestion for an implementation of a closed-loop control system with continuous model update is given.

8.2 System outline

The optimization that is considered in this paper is long term optimization. In the suggested solution NMPC is used for optimal reservoir control combined with EnKF for continuous model update. The solution suggested is similar to that of D.R. Brouwer et al. [7] and is illustrated in Figure 8.2. In this

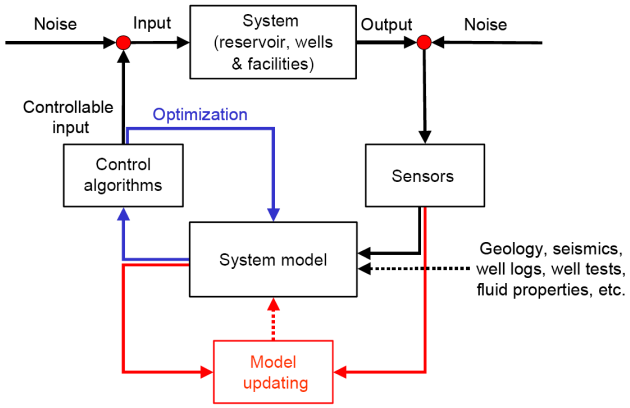


Figure 8.2: System Layout

figure it can be seen how the model update loop (EnKF) in red and the controller loop (NMPC) in blue interact with both the system and the system model. The flow of the system can be described in the following procedure

1. Create an initial system model is made using Geology, seismic, well logs, well tests, fluid properties etc. From the uncertainties in the parameters an initial ensemble is created as described in Section 5.2.

2. Compute an optimal control strategy over the control horizon from $t = t(k)$ to $t(k + T_p)$. This is done by using MPC on the model with the most probable model parameters. The most probable estimate is assumed to be the mean of the ensemble.

3. Run all ensemble members and real reservoir forward in time until there is new available data.

4. The ensemble observations and the observation from the real reservoir are used together to update the ensemble. This is done according to the filtering step in Algorithm 3. The mean is also updated in this scheme and the result is used as the new most probable model.

5. If $t < t(k + 1)$ go to step 3, else proceed to the next step. [1]

6. Update the time step ($k = k + 1$) and go to step 2

This method is then repeated and the optimal excitation is calculated based on a continuously updated model. This outline a closed-loop approach as known from control engineering. Make this

[1] Using several data observations between each control setting update, will let the estimator converge before the next control strategy is computed and help ensure stability.

scheme work poses some challenges. A selection of issues that has to be solved is outlined in the next section.

8.3 Challenges in a combined solution

The suggested method in the previous section is a closed-loop approach. To make such a system work there are some problems that has to be overcome. In this section some of these problems will be outlined. Another good description of challenges regarding closed-loop management can be found in J.D. Jansen et al. [28].

The first issue is efficiency, which is the key to making this achievable in practice. To make such a system compute a solution in a reasonable time the algorithms have to be efficient. P. Sarma et al. [40] investigate efficient optimization algorithms using adjoint methods. This solution limits the reservoir simulators that can be used, since not all commercial simulators have the options to compute adjoint methods. In any case an investigation on model reduction should be done to save computation time. T Heijn et al. [26] has showed promising improvements in computation time using amongst other "Subspace identification" as an alternative to classical regularization methods [12]. Apart from reduction techniques as the one mentioned above, 'proxy models', based on experimental design and classic upscaling techniques developed by reservoir engineers can be used. Work is currently done in multi-scale and adaptive grid refinement methods, and promising results are shown (J.D. Jansen et al. [28]). For an efficient model update algorithm, EnKF is a possible choice. As mentioned earlier in this paper EnKF saves a lot of computation time compared to traditional gradient-based methods. EnKF also computes sequentially and does not need to assimilate the old data over again to incorporate new information.

Most of the previous work done on closed-loop reservoir management considers short term optimization. Here the information needed is much more limited than in long-term reservoir management. D.R. Brouwer et al. [7] gave some references to papers considering the short-term approach. Long-term management causes much higher demands for the models predictive qualities. One example of a challenge is predicting the time water and/or gas breakthrough before they happen.

Another issue that has not been investigated much before is what influence closing the loop has on the two methods combined. It can cause instability problems using a combination of parameter estimation and optimal control. A simple example is if the EnKF has not converged fully before the next optimal strategy is to be produced. A way of compensating for this problem is by running several model updating steps before calculating the new control strategy. Since the system is highly nonlinear it is hard to prove stability of such a loop and trial and error has been the approach so far. In robotics some theory have been developed for stability analysis of closed-loop properties, i.e. the passivity based approach by H. Berghuis and H. Nijmeijer [5]. However connecting such theory to the reservoir case is inconvenient because of problem dimensions and the usage of commercial simulators. Closing the loop might also influence the observability of the system. One can imagine that strategies in the control can help the identification of the system by introducing more variety in the excitation. In other cases the closed loop may disturb the estimation and make the task worse. However, this has to be investigated further to draw any conclusions.

In this section some challenges to closed loop on long-term reservoir control has been outlined. These problems have been issued in some papers where a similar scheme has been applied. In addition to D.R. Brouwer et al. and J.D. Jansen some other methods for history matching in such a system was proposed. P. Sarma et al. [41] investigated reservoir management using an adjoint-based approach both for optimal control and for the history matching. I. Aitokhuehi and L.J. Durlofsky [3] proposed another method using the same optimization method as B. Yeten presented and as model

update they used a stochastic probability perturbation method. P. Sarma et al.[41] however argued this method to be easy to implement, but expensive computationally and therefore might have limits in a practical setting. In conclusion this is a very interesting field which has good potential, but the methods still need more research before being applied on real reservoir cases.

Chapter 9

Conclusions

Ensemble Kalman Filter has been examined in this paper. A thorough investigation of the theory behind this filter has been presented and it has been applied on a linear case. The results proved that the IRIS EnKF implementation match the theory for a linear case. In recent years a lot of work have been done with EnKF on continuous reservoir model updating. The implementation part of the work in this paper is at a stage somewhere between the first applications and where the EnKF is today. Nevertheless, the work in this paper has illustrated some interesting pointers.

An observability criteria for nonlinear systems has been outlined. Also, a tie to EnKF used for history matching has been established. In general there are two questions that describes the identification problem

1. Which parameters can be determined uniquely if the measurements were exact?

2. Given the expected level of error in the measurements, how accurately is it expected to be able to estimate the parameters?

The first question, which relates to observability theory, is in general hard to answer for large-scale nonlinear problem. However, the observability theory presented in this paper work as good guidelines on how to reduce the dimension of the history matching problem.

The simulation results have shown how important the prior information is for the result of the filter. In the simulations where the prior information is close to that of the true solution the standard deviation decreases. The simulation with flat mean was initially further off from the true solution and some problems occurred. First the results concluded that the correlation length was to large in Simulation 2. Afterward with a smaller correlation length in Simulation 4 the filter failed completely. The reason for this was that to little information about the system was available, but still history matching looked promising. This leads to the conclusion that the system is not observable, and most of the parameters are not identifiable.

Also the incremental information in the measurements have been under investigation. Most of the information in the shoe box reservoir measurements lies in the water breakthrough. After this there are little information in the measurements. It was noted that the static variable estimates became worse in the period subsequent to the water breakthrough. This is possibly because of the effect of assimilated noise. This is still an issue which have to be investigated further and only thoughts on what might be the cause were posed in this paper.

The last topic in this paper was a proposed application to combine the Ensemble Kalman Filter (EnKF) with Model Predictive Control (MPC) in a closed-loop reservoir management scheme. The suggested scheme has had some success in previous work by others, but mostly on simple examples.

There are still some challenges to take the suggested application to a real reservoir, which were outlined and discussed. In general these involve

- computation time
- reservoir model predictive ability
- effects of closing the loop

This paper have presented EnKF from the start and until where it is today. The implementation part however came up a little short in testing all properties of the EnKF. As presented in the results and conclusions there are still a lot of these properties that should be investigated further. An outline of where the work should continue to focus is given in the next chapter.

Chapter 10

Further work

The evolution of EnKF for history matching has been presented earlier in this paper. As outlined before, the work in this paper starts with the basic statistic theory in the EnKF and work its way toward where the method is today. To limit the work some restrictions were made in the implementation part. The work done in this paper on EnKF for parameter estimation is limited to estimating only porosity and permeability and assumes a correct model. This is far from the real cas, but an useful exercise to illustrate the method. As a result of this the further work would involve taking this analysis all the way to a real reservoir. The steps that remain in this process can roughly be put into three parts which are

1. Introducing model error
2. Including more information in the problem
3. Working with real reservoirs

In this chapter, these steps will be explained in more details and thoughts on how to approach the matter are outlined. In addition to the practical aspects there is also need for more work on establishing a theoretical basis for the improvements in the procedure. At the end of this chapter some thoughts on where to take the methodology from today and forward is presented.

10.1 Introducing model error

In the implementation done in this paper the model was assumed correct. This is never the case when modelling a real reservoir. The differential equations which the reservoir model is built on will not cover all of the dynamics and some assumptions always has to be made. The EnKF potentially has a very smooth way of incorporating model uncertainty. This is done by adding noise in the dynamic variables and updating them during the filtering step in the algorithm. As described by V. Haugen et al. [24] this update provides a good starting point for computing predictions. A controlled way of testing this would be to build a complex reservoir model as a reference and a simplified model to history match to the original one. By doing this one can introduce model error in a controlled manner and investigate the EnKF properties. J.D. Jansen et al.[28] outlines how such a virtual asset model can be used for development of new concepts and algorithms. Figure 10.1 is from their paper and illustrate how a virtual asset model would be incorporated in a closed-loop scheme as the one presented in Chapter 8. In addition to looking at model errors the task of incorporating more information into the problem is covered in the next section.

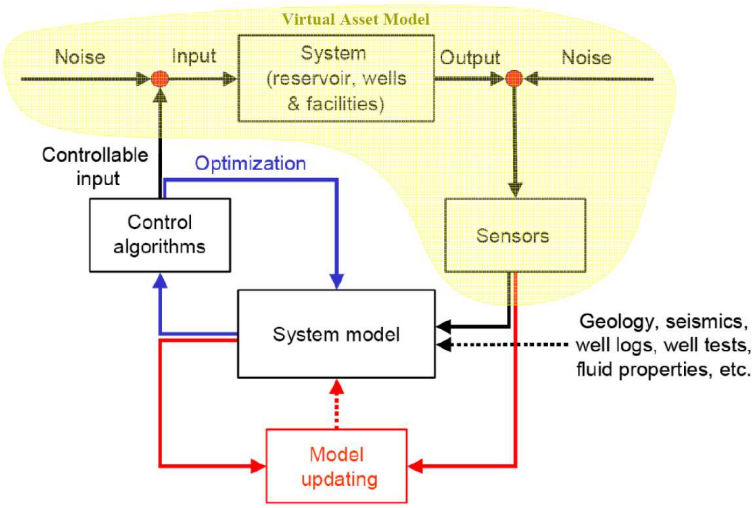


Figure 10.1: Virtual Asset Model used to verify a closed-loop scheme

10.2 Including more information in the problem

In this paper only the porosity and permeability have been subject to estimation in the implementation. Evensen et al. [19] proposed estimating other parameters like contact surfaces and transmissivities. These are important parameters which often are subject to a lot of uncertainty. A method that could predict the water oil contact before any information on the water breakthrough is a valuable asset in for instance optimal reservoir control [7].

Incorporating more measurements is, as discussed before, a way of improving the chance of good results. Interesting work with measurements like 4-D seismic[44] etc. has been done with good results. Implementation consists of coupling the EnKF solution today with rock physics and seismic modelling software.

Both these suggestions would have to be investigated further and tested in a case like the one presented in the previous section.

10.3 Working with real reservoirs

Most of the early work that has been done with EnKF has applied the method on test case reservoirs. Only in the last year or two the work has moved its focus to the real case. There is a step up from the model used in this paper and implementing the EnKF on a real reservoir model. Some challenges has been mentioned above in introducing mode errors and including more information in the problem. In addition some obstacles regarding to the larger models needs to be overcome. The reservoir used in this paper have been simulated on a single desktop computer with reasonable computation time. However, implementing real reservoir cases poses the need for parallel processing. This issue has not been discussed in this paper, but the EnKF algorithm is excellent for parallel processing, spreading the simulations in the prediction step on several computers. Another approach to this problem is reducing the model. A model reduction was the described in the model error section.

Here a complex reservoir model was suggested to be approximated using a more coarse grid, hence decreasing computation time.

In general when using real life reservoir cases the understanding for geology and phenomenas that occur in reservoir simulation. By this it is understood that not everything is as perfect as in the model. The model might not be able to express all dynamics in the reservoir. Some of this can also be investigated in the case suggested to introduce model error, but there will always be some challenges with real life reservoirs that can not be foreseen. In addition the measurements may be poor with noise, biases and outliers[1]. Methods for handling such problems with measurement will have to be implemented. When dealing with measurements it is also worth noting that the EnKF is a method that improves the results as more information enters the method. Therefor the new and intelligent wells with many measurements will make the history matching process easier.

In this section some problems with EnKF applied on real reservoirs has been outlined. A lot of these challenges were discussed by both V. Haugen et al. [24] and G. Evensen et al. [19] when they applied EnKF to a North Sea Reservoir Model.

10.4 The future of EnKF

EnKF has shown good results on history matching cases both for real reservoirs and test models. Brouwer et al.[7] also showed how EnKF works together with optimal reservoir control. This is something that had potential to increase oil recovery, but still contains many aspects to investigate. The goal would be an automatically updated reservoir model used with optimal control for automatic reservoir management. For this to work the EnKF has to be made robust against problems that might occur. Also tuning in the form of specifying model and measurement noise is an issue. Some of the instability issues, that have been illustrated both in this paper and in some of the other work, have to be solved

The final step of the further work is to make the product more commercial. Building a user interface and simplifying the process of maintaining the filter. This implies that the model is developed to a stage where it can be incorporated with reservoir management procedures. Though the EnKF is not at this stage yet and a lot more research on the method have to be done, the author believes that the method has both the potential and many of the qualities that is needed to get there.

[1]Outliers are single measurements that are corrupt and completely off compared to the rest of the measurement sequence

Bibliography

[1] The Mathworks, Inc. (http://www.mathworks.com/).

[2] http://enkf.nersc.no/.

[3] I. Aitokhuehi and L.J. Durlofsky. Optimizing the performance of smart wells in complex reservoirs using continuously updated geological models. *J. of Pet. Sci. and Eng.*, 48:254–264, 2005.

[4] J. I. Allen, M. Eknes, and G. Evensen. An ensemble kalman filter with a complex marine ecosystem model: hindcasting phytoplankton in the cretan sea. *Annales Geophysicae*, 20:1–13, 2002.

[5] H. Berghuis and H. Nijmeier. A passivity approach to controller-observer design for robots. *IEEE Transactions on Robotics and Automation*, 9(6):740–754, December 1993.

[6] D.R. Brouwer and J.-D. Jansen. Dynamic optimization of waterflooding with smart wells using optimal control theory. *Proceedings of the 2004 SPE European Petroleum Conference, Aberdeen.* (SPE 78278).

[7] D.R. Brouwer, G. Nævdal, J.D. Jansen, E.H. Vefring, and C.P.J.W. van Kruijsdijk. Improved reservoir management through optimal control and continous model updating. *Proceeding of the SPE Annual Techincal Conference and Exhibition held in Houston, Texas, U.S.A.*, September 2004. (SPE 90149).

[8] R.G. Brown and P.Y.C. Hwang. *Introduction to random signals and applied Kalman filtering: with MATLAB exercises and solutions.* Wiley New York, 1997. ISBN 0-471-12839-2.

[9] K. Brusdal, J.M. Brankart, G. Halberstadt, G. Evensen, P. Brasseur, P.J. van Leeuwen, E. Dombrowsky, and J. Verron. A demonstration of ensemble-based assimilation methods with a layered ogcm from the perspective of operational ocean forecasting systems. *Journal of Marine Systems*, 40-41:253–289, 2003.

[10] G. Burgers, P. J. van Leeuwen, and G. Evensen. Analysis scheme in the ensemble kalman filter. *Mon. Weather Rev.*, 126:1719–1724, 1998.

[11] C.-T. Chen. *Linear System Theory and Design.* Saunders College Publishing, 1984. ISBN 0-03-071691-8.

[12] C.B. Chung and C. Kravaris. Incorporation of a priori information in reservoir history matching by regularization. *Society of Petroleum Engineers*, 1990. (SPE 21615).

[13] N. Dolle, D.R. Brouwer, and J.-D. Jansen. Dynamic optimization of waterflooding with multiple injectors and producers using optimal control theory. *Proceeding of XIV International Conference on Computational methods in Water Resources held in Delft, The Netherlands*, 2002.

[14] E.L. Dougherty. Application of optimization methods to oilfield problems: Proved, probable, possible. *Prepared for the 47th annual fall meeting of SPE og AIME, San Antonio, Texas, USA*, 1972. (SPE 3978).

[15] M. Eknes and G. Evensen. An ensemble kalman filter with a 1-d marine ecosystem model. *Journal of Marine Systems*, 36:75–100, 2002.

[16] G. Evensen. *Data Assimilation: The Ensemble Kalman Filter*. Springer, 2007.

[17] G. Evensen. Sampling strategies and square root analysis schemes for the enkf. *Ocean Dynamics*, 54:539–560, 2004.

[18] G. Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics. *J. Geophys. Res.*, 99:10,143–10,162, 1994.

[19] G. Evensen, J. Hove, H. C. Meisingset, E. Reiso, K. S. Seim, and Ø. Espelid. Using the enkf for assisted history matching of a north sea reservoir model. *Proceeding of SPE Reservoir Simulation Symposium, Woodlands, Texas, USA*, 2007. (SPE 106184).

[20] B.A. Foss. *On Parameter Identification in Reservoirs*. PhD thesis, Norwegian Institute of Technology, 1987.

[21] G. Gao, M. Zafari, and A. C. Reynolds. Quantifying uncertainty for the punq-s3 problem in a bayesian setting with rml and enkf. *Proceeding of SPE reservoir simulation symposium, Houston, Texas, USA*, 2005. (SPE 93324).

[22] G. C. Goodwin, S. F. Graebe, and M. E. Salgado. *Control System Design*. Prentice Hall PTR, 2000. ISBN 0-13-958653-9.

[23] Y. Gu and D. S. Oliver. History matching of the punq-s3 reservoir model using the ensemble kalman filter. *Proceeding of SPE Annual Technical Conference and Exhibition, Houston, Texas, USA*, 2004. (SPE 89942).

[24] V. Haugen, L.-J. Natvik, G. Evensen, A. Berg, K. Flornes, and G. Nævdal. History matching using the ensemble kalman filter on a north sea field case. *Proceedings of the 2006 SPE Annual Technical Conference and Exhibition, San Antonio, Texas, USA*. (SPE 102430).

[25] J. K. Hedrick and A. Girard. Control of nonlinear dynamic systems: Theory and applications, 2005. http://www.me.berkeley.edu/ME237/notes.html).

[26] T. Heijn and J.D. Jansen. Generation of low-order reservoir models usingsystem-theoretical concepts. *Proceeding of SPE Reservoir Simulation Symposium held in Houston, Texas, USA*, 2003. (SPE 79674).

[27] M. Hwang and J. H. Seinfeld. Observability of nonlinear systems. *Journal of Optimization Theory and Applications*, 10(2):67–72, May 1972.

[28] J.D. Jansen, D.R: Brouwer, G. Nævdal, and C.P.J.W. van Kruijsdijk. Closed-loop reservoir management. *First Break*, 23:43–48, January 2005.

[29] A.H. Jazwinski. *Stochastic processes and filtering theory*. Academic Press New York, 1970. ISBN 3-540-38300-X.

[30] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[31] J.S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer Verlag, 2001. ISBN 0-387-95230-6.

[32] N. Liu and D. S. Oliver. Ensemble kalman filter for automatic history matching of geologic facies. *J. Petroleum Sci. and Eng.*, 47:147–161, 2005.

[33] N. Liu and D. S. Oliver. Critical evaluation of the ensemble kalman filter on history matching of geologic facies. *Proceeding of SPE reservoir simulation symposium, Houston, Texas, USA*, 2005. (SPE 92867).

[34] P. Meum. Optimal reservoir control with nonlinear mpc and eclipse. Master's thesis, Norwegian University of Science and Technology, 2007. (Submitted June 2007).

[35] G. Nævdal and R. J. Lorentzen. Improving enkf for better performance on non-linear problems.

[36] G. Nævdal, T. Mannseth, and E. H. Vefring. Near-well reservoir monitoring through ensemble kalman filter. *Proceeding of SPE/DOE Improved Oil recovery Symposium, Tulsa, Oklahoma, USA*, 2002. (SPE 75235).

[37] G. Nævdal, L. M. Johnsen, S. I. Aanonsen, and E. H. Vefring. Reservoir monitoring and continuous model updating using ensemble kalman filter. *Proceeding of SPE Annual Technical Conference and Exhibition, Denver, Colorado, USA*, 2003. (SPE 84372).

[38] C.P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer Verlag, 1999. ISBN 0-387-98707-X.

[39] J.R.P. Rodrigues. Calculating derivatives for history matching in reservoir simulators. *Presented at the 2005 SPE Reservoir Simulation Symposium, Houston, Texas*, 2005. (SPE 93445).

[40] P. Sarma, W.H. Chen, L.J. Durlofsky, and K. Aziz. Production optimization with adjoint models under nonlinear control-state path inequality constraints. *Proceeding of SPE Intelligent Energy Conference and Exhibition held in Amsterdam, The Netherlands*, 2006. (SPE 99959).

[41] P. Sarma, W.H. Chen, L.J. Durlofsky, and K. Aziz. Efficient real-time reservoir management using adjoint-based optimal control and model updating. *Computational Geosciences*, 10(1): 3–36, 2006.

[42] Schlumberger, 2006. ECLIPSE Reference Manual 2006.2.

[43] R.W. Schulze-Riegert, J.K. Axmann, O. Haase, D.T. Rian, and Y.-L. You. Evolutionary algorithms applied to history matching of complex reservoirs. *Presented at the 2001 SPE Reservoir Simulation Symposium, Houston, Texas*, 2002. (SPE 77301).

[44] J.-A. SKjervheim, G. Evensen, S.I. Aanonsen, B.O. Ruud, and T.A. Johnansen. Incorporating 4d seismic data in reservoir simulation models sing ensemble kalman filter. *Proceeding of SPE Annual Technical Conference and Exhibition, Dallas, Texas, USA*, 2005. (SPE 95789).

[45] B. Sudaryanto and Y.C. Yortosos. Optimization of fluid front dynamics in orous media using rate control. i. equal mobility fluids. *Physics of Fluids*, 12(7), March 2000.

[46] B. Vallès, G. Nævdal, R. J. Lorentzen, A. M. Berg, and K. M. Flornes. Description of the iris enkf matlab code. *IRIS*, October 2006.

[47] A.T. Watson, G.R. Gavalas, and J.H. Seinfeld. Identifiability of estimates of two-phase reservoir properties in history matching. *Society of Petroleum Engineers*. (SPE 12579).

[48] X.-H. Wen and W. H. Chen. Real-time reservoir model updating using ensemble kalman filter with confirming option. *Proceeding of SPE reservoir simulation symposium, Houston, Texas, USA*, 2005. (SPE 92991).

[49] B. Yeten, L.J. Durlofsky, and K. Aziz. Optimization of smart well control. *Proceeding of SPE International Thermal Operations and Heavy Oil Symposium and International Horizontal Well Technology Conference held in Calgary, Alberta, Canada*, 2002. (SPE 79031).

[50] M. Zafari and A.C. Reynolds. Assessing the uncertainty in reservoir description and performance prediction with the ensemble kalman filter. *Proceeding of SPE Annual Technical Conference and Exhibition, Dallas, Texas, USA*, 2005. (SPE 95750).

Appendix A

Matlab Code for Linear EnKF and KF

In this appendix a simple matlab code for a linear EnKF and KF algorithm is proposed. Some of the methods in the EnKF algorithm regarding the noise creation is not published, but their functionality is outlined in this paper.

```
%#####################################################
%   Comparison of KF and EnKF
%#####################################################


%Initialization of the true system
    %   x_{k+1} =Ax_{k}+q_{k}
    %   y_{k}   =Cx_{k}+v_{k}
    %
    %   E[qq']=Qtrue  E[vv']=Wtrue
    Ctrue=[1 0.5];
    Atrue=[0.5 0.1; 0.2 0.3];
    Qtrue=diag([0.01 0.02]);
    Wtrue=0.01;
    t=1:50;
    xinit=[0;0];
    numStates=2;
    numParam=0;
%Compute true solution
    if ¬exist('trueSolution.mat','file')
        [ytrue,Q,W]=computeTrueSolutionLinear...
                    (Atrue,Ctrue,Qtrue,Wtrue,xinit,t);
        save('trueSolution.mat','ytrue','Q','W');
    else
        load('trueSolution.mat','ytrue','Q','W');
    end

%Initial State for the filters
    xinit=[1 ; 1];
%Initialization of noise covariance matrices for the Filters
    Qfilter=[0.2 0.01 ; 0.01 0.2];
    Wfilter=Wtrue;
%Initialization for EnKF
    ensSize=1000;
    meanState=xinit;
    LPapost=chol(Qfilter)';
    Afilter_EnKF=Atrue;
```

```
38      Cfilter_EnKF=Ctrue;
39      ensemble=zeros(size(meanState,1),ensSize);
40      for ii=1:ensSize
41          ensemble(:,ii)=addgnoise(meanState,Qfilter);
42      end
43      options.outlist=1; %see EnKF.m
44  %Initialization for KF
45      xhat=xinit;
46      Phat=Qfilter;
47      Afilter_KF=Atrue;
48      Cfilter_KF=Ctrue;
49
50  %The loop
51  for i=1:length(t)
52      %% ################  EnKF  ################
53          %Storing results
54              bigEnsemble([2*(i-1)+1 ; 2*(i-1)+2],:)=ensemble;
55              Penkf(:,:,i)=LPapost*LPapost';
56              xenkf(:,i)=meanState;
57          %Prediction step
58              for j=1:ensSize
59                  %Prediction
60                  ensemble(1:numStates,j)=...
61                      Afilter_EnKF*ensemble(1:numStates,j);
62                  %Legge på modellstøy
63                  ensemble(:,j)=addgnoise(ensemble(:,j),Qfilter);
64              end
65          %Filtering step
66              %Retrieve Measurements
67              y=ytrue(:,i);
68              %EnKF oppdatering
69                  %Apriori meanstate
70                      meanState=mean(ensemble,2);
71                  %Apriori covariance matrix
72                      LPapri=zeros(size(ensemble));
73                      for j=1:ensSize
74                          LPapri(:,j)=(1/sqrt(ensSize-1))*...
75                              (ensemble(:,j)-meanState);
76                      end
77                  %Ensemble measurement noise
78                      ej=zeros(size(Wfilter,1),ensSize);
79                      yj=zeros(size(Wfilter,1),ensSize);
80                      sqW=chol(Wfilter)';
81                      for j=1:ensSize
82                          ej(:,j)=addgnoise(zeros(size(ej(:,j))),sqW,1);
83                      end
84                      %Adjusting to get zero mean on the measurment noise
85                          We=zeros(size(ej,1),size(ej,1));
86                          meanej=mean(ej,2);
87                          for j=1:ensSize
88                              ej(:,j)=ej(:,j)-meanej; %new ej w/zero mean
89                              We=We+ej(:,j)*ej(:,j)'; %true Wfilter
90                              yj(:,j)=y+ej(:,j);      %ensemble measurment
91                          end
92                          We=(1/(ensSize-1))*We;
93                          y=mean(yj,2);
94                  %Gain matrix
95                      K=LPapri*(LPapri'*Cfilter_EnKF')*inv(...
96                          (Cfilter_EnKF*LPapri)*(LPapri'*Cfilter_EnKF')+We);
```

```
97                  %Aposteriori ensemble and meanstate
98                      for j=1:ensSize
99                          ensemble(:,j)=ensemble(:,j)+K*(yj(:,j)-...
100                                         Cfilter_EnKF*ensemble(:,j));
101                     end
102                     meanState=meanState+K*(y-Cfilter_EnKF*meanState);
103                 %Aposteriori covariance matrix
104                     LPapost=zeros(size(ensemble));
105                     for j=1:ensSize
106                         LPapost(:,j)=(1/sqrt(ensSize-1))*...
107                             (ensemble(:,j)-meanState);
108                     end
109 %             [meanState,ensemble,LPapost]=...
110 %                   EnKF(ensemble,Wfilter,Cfilter_EnKF,y,[],options);
111     %% ################   KF   ################
112         %lagring av resultater
113             Pkf(:,:,i)=Phat;
114             xkf(:,i)=xhat;
115         %Prediction step
116             %Model
117             xhat=Afilter_KF*xhat;
118             Phat=Afilter_KF*Phat*Afilter_KF'+Qfilter;
119         %Filtrering step
120             %Gain og covariance
121             K=Phat*Cfilter_KF'*inv(Cfilter_KF*Phat*Cfilter_KF'+Wfilter);
122             Phat=(eye(size(Afilter_KF))-K*Cfilter_KF)*Phat*...
123                     (eye(size(Afilter_KF))-K*Cfilter_KF)'+K*Wfilter*K';
124             %Filtering
125             xtemp=xhat+K*(y-Cfilter_KF*xhat);
126             xhat=xtemp(1:numStates);
127 end%the loop
```

Appendix B

Illustration of initial ensemble span

THIS simple example is created to show how the search for the solution in the EnKF is influenced by the choice of initial ensemble. Consider a system with two states (x_1 and x_2) and one measurement (y) defined as

$$\begin{bmatrix} x_{1,k+1} \\ x_{2,k+1} \end{bmatrix} = \begin{bmatrix} \theta_1 & 0 \\ 0.05 & 0.9 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} + \begin{bmatrix} 0.1 \\ 0 \end{bmatrix} u_k \tag{B.1a}$$

$$y_k = \begin{bmatrix} \theta_2 & 5 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} \tag{B.1b}$$

Where $\boldsymbol{\theta} = [\theta_1, \theta_2]$ are two unknown parameters. The true solution of these values is $\boldsymbol{\theta}^t = [0.5, 1]$ and in the filter they are initialized by a mean of $\bar{\boldsymbol{\theta}}_0 = [0.4, 0.9]$. Then four different standard deviations for the initial ensemble are used in the filter. The tests are initialized according to Table B.1. The

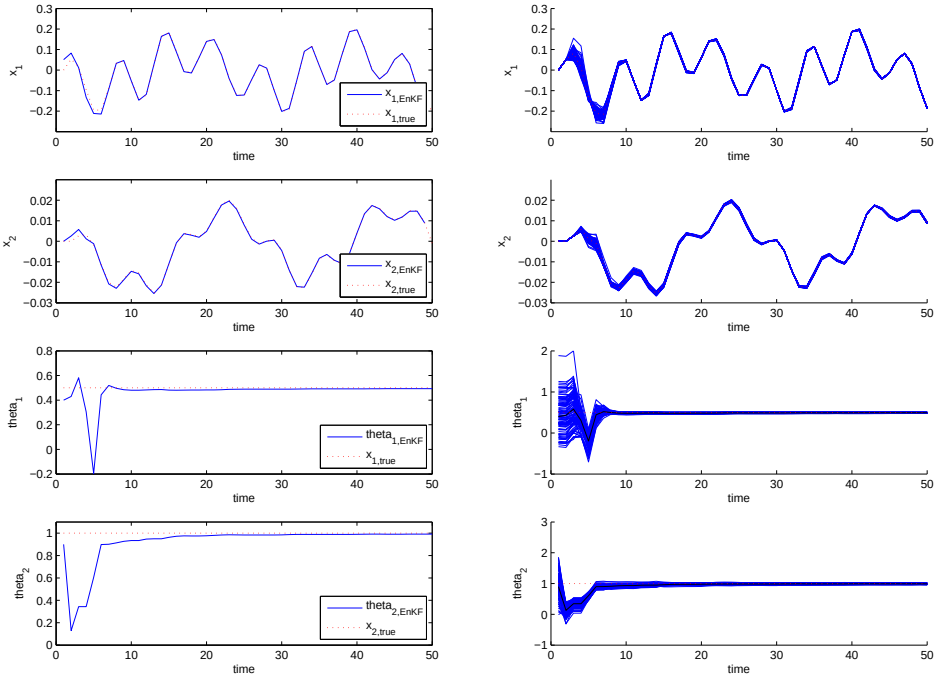
Test	σ_{0,θ_1}	σ_{0,θ_2}
1	0.2	0.2
2	0	0
3	0	0.2
4	0.2	0

Table B.1: Initial ensemble test - standard deviations

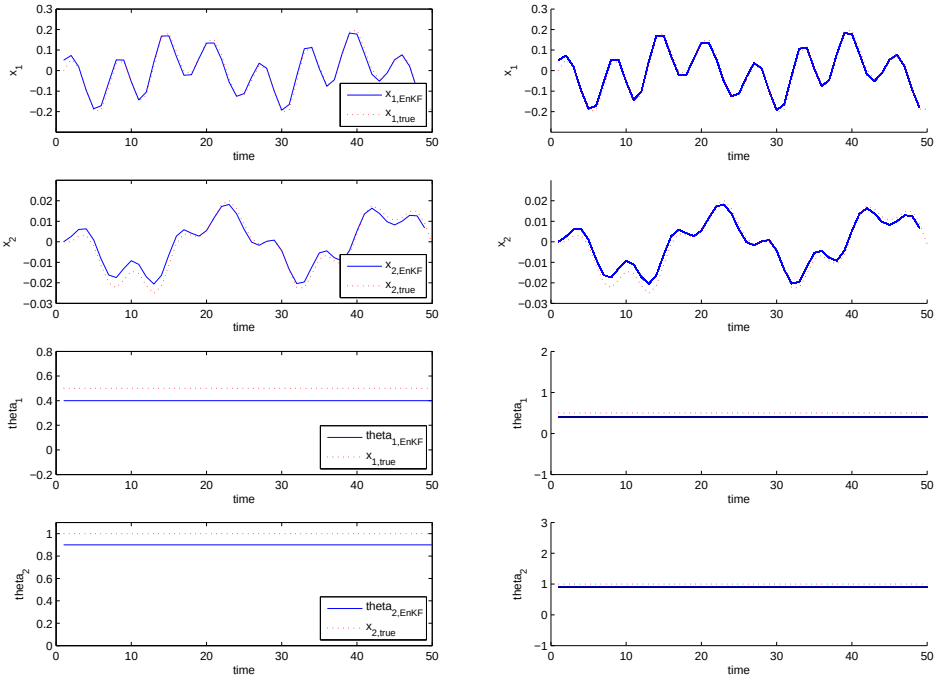
results are shown in Figure B.1 and B.2.

The plots show the EnKF estimate and the true value for the two states (x_1 and x_2) and the two parameters (θ_1 and θ_2). One can observe the effects of various spans in the initial ensemble. When observing the plots for the parameters one can observe how the estimates converge toward the true solutions. The first test span both the θ_1 and θ_2 axis and the EnKF finds the true solution. The second test does not span any space at all and therefore no search is conducted. The third and fourth test span respectively only along the θ_1 or θ_2 axis and search is conducted only in the span. This means that Test 3 converge along the θ_1 axis but not the θ_2 axis and vice versa for Test 4. From the plots of θ for the various tests one can see that the solution is only searched for in the space spanned by the initial ensemble.

Another approach to this problem was outlined by G. Evensen et al.[19]. They consider the given Kalman update in Algorithm 3 when no model noise is added to the ensemble. Further they show that the updated ensemble is a weakly nonlinear combination of the forecast ensemble members. This can be interpreted such that one can only expect to find corrections to the static variables which can be represented in the space spanned by the initial ensemble.

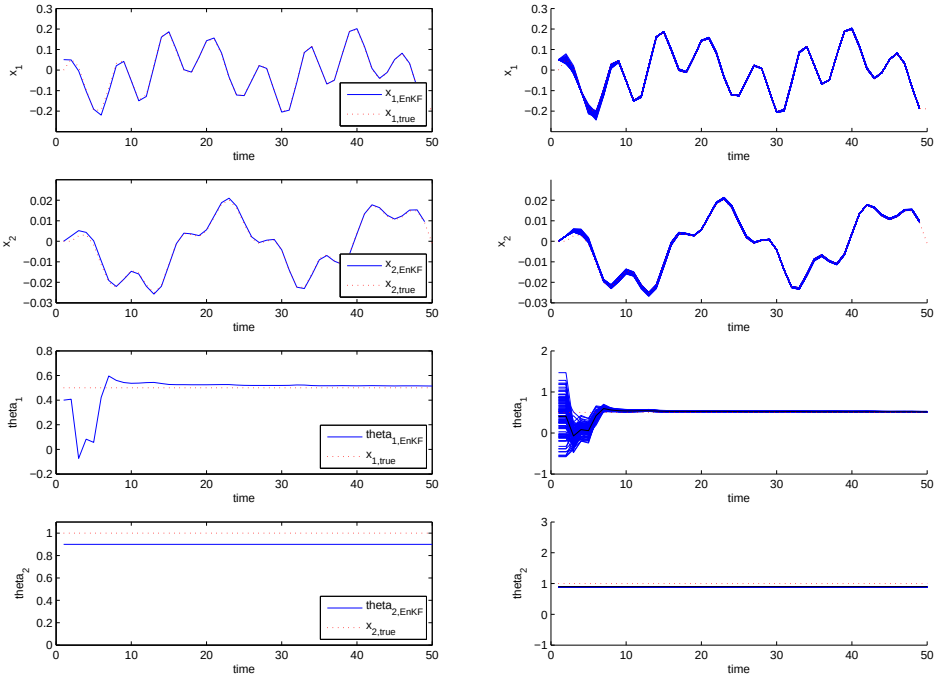


(a) Test 1

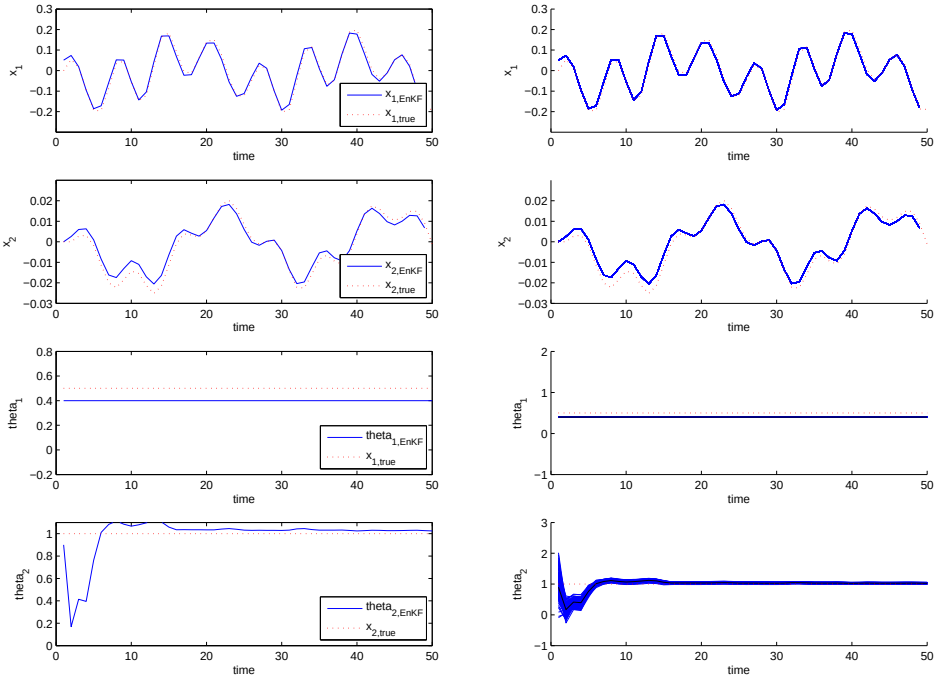


(b) Test 2

Figure B.1: Results with different span in initial ensemble



(a) Test 3



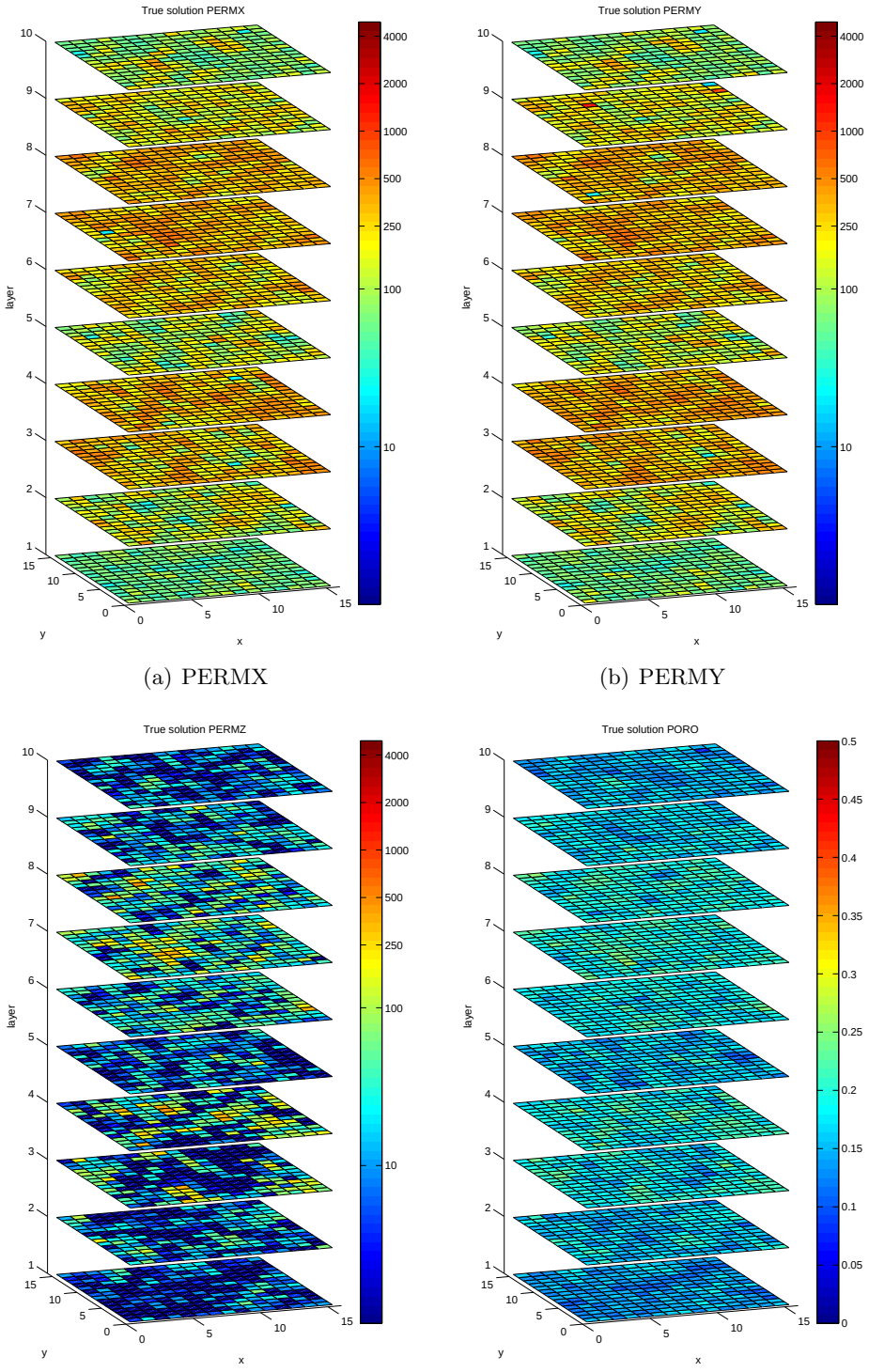
(b) Test 4

Figure B.2: Results with different span in initial ensemble

Appendix C

Shoe box reservoir true parameters

An overview of the true static variables in the shoe box reservoir model is given in this appendix. This should be considered as a reference for comparison of both the initial ensembles in Appendix B and the results presented in both Appendix F and the Electronic Appendix G. Figure C.1 displays the true permeability and porosity fields in each of the 10 layers.



(a) PERMX

(b) PERMY

(c) PERMZ

(d) PORO

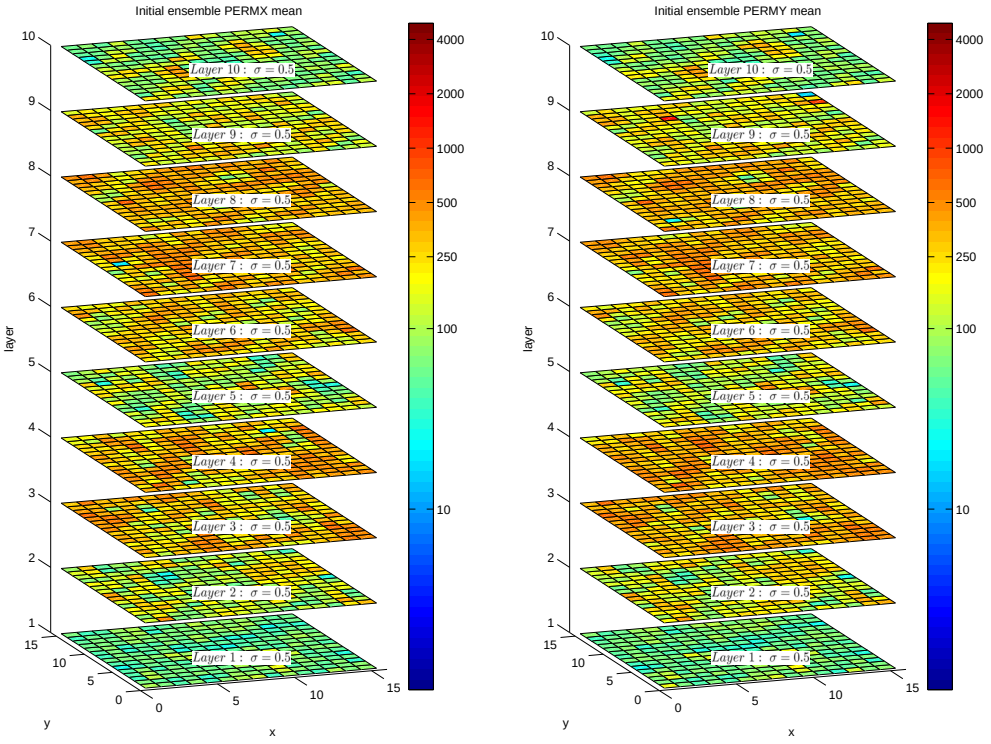
Figure C.1: True PERM and PORO for the shoe box reservoir model

Appendix D

Simulation Initializations

This appendix illustrates the various initializations of the static parameters used in the simulations. The reference is found in Appendix C where the true static parameters are shown. The following plots are included

- Figure D.1 - Initialization for simulation 1
- Figure D.2 - Initialization for simulation 2
- Figure D.3 - Initialization for simulation 3



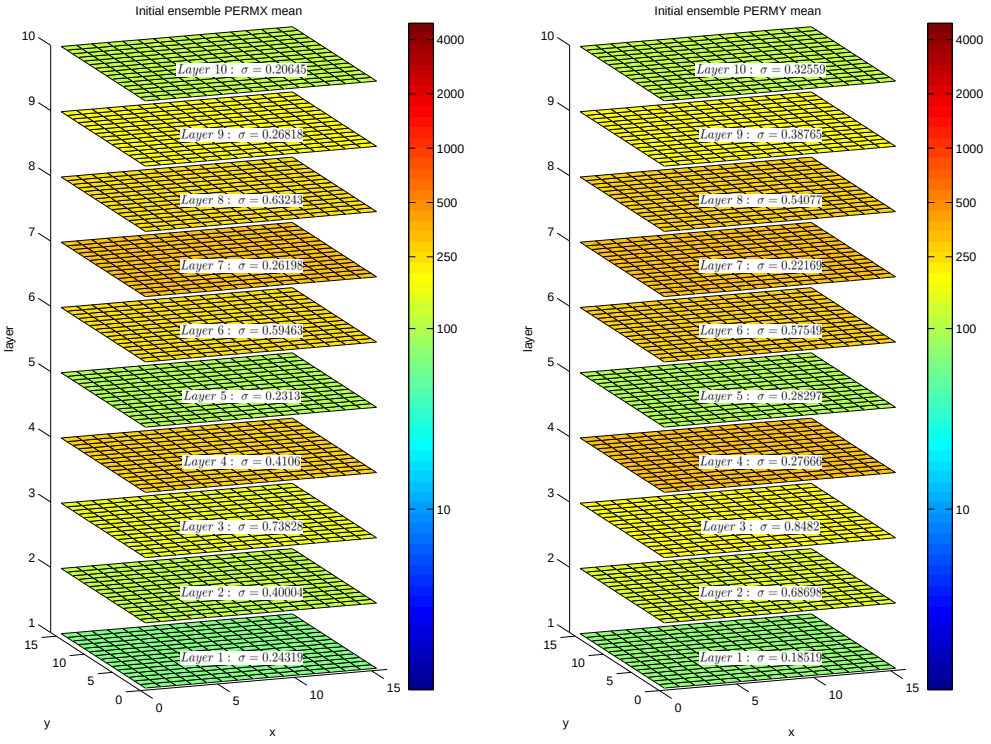
(a) PERMX

(b) PERMY

(c) PERMZ

(d) PORO

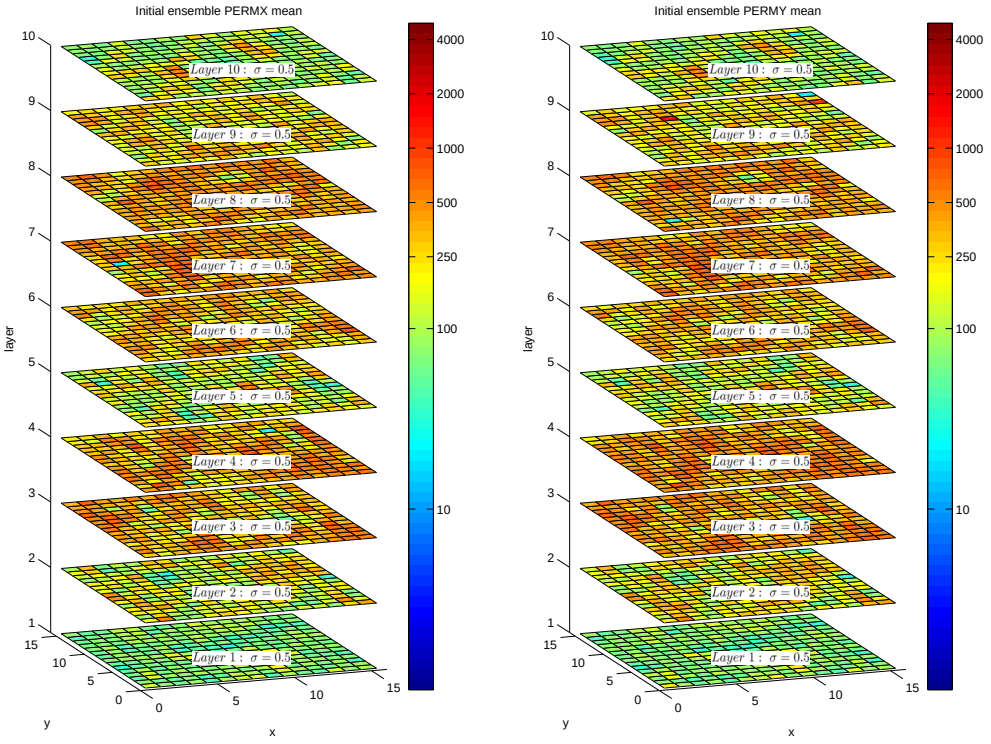
Figure D.1: Initialization PERM and PORO for simulation test 1



(a) PERMX (b) PERMY

(c) PERMZ (d) PORO

Figure D.2: Initialization PERM and PORO for simulation test 2



(a) PERMX

(b) PERMY

(c) PERMZ

(d) PORO

Figure D.3: Initialization PERM and PORO for simulation test 3

Appendix E

Simulations history matching results

This appendix contains the history matching results from the simulations that were described in Chapter 7. The measurements presented in these plots are also described in the same Chapter. Only a selection of the results is presented in this appendix. For the full overview containing both history matching and the static parameters, see the Electronic Appendix (Appendix G)

E.1 Simulation 1

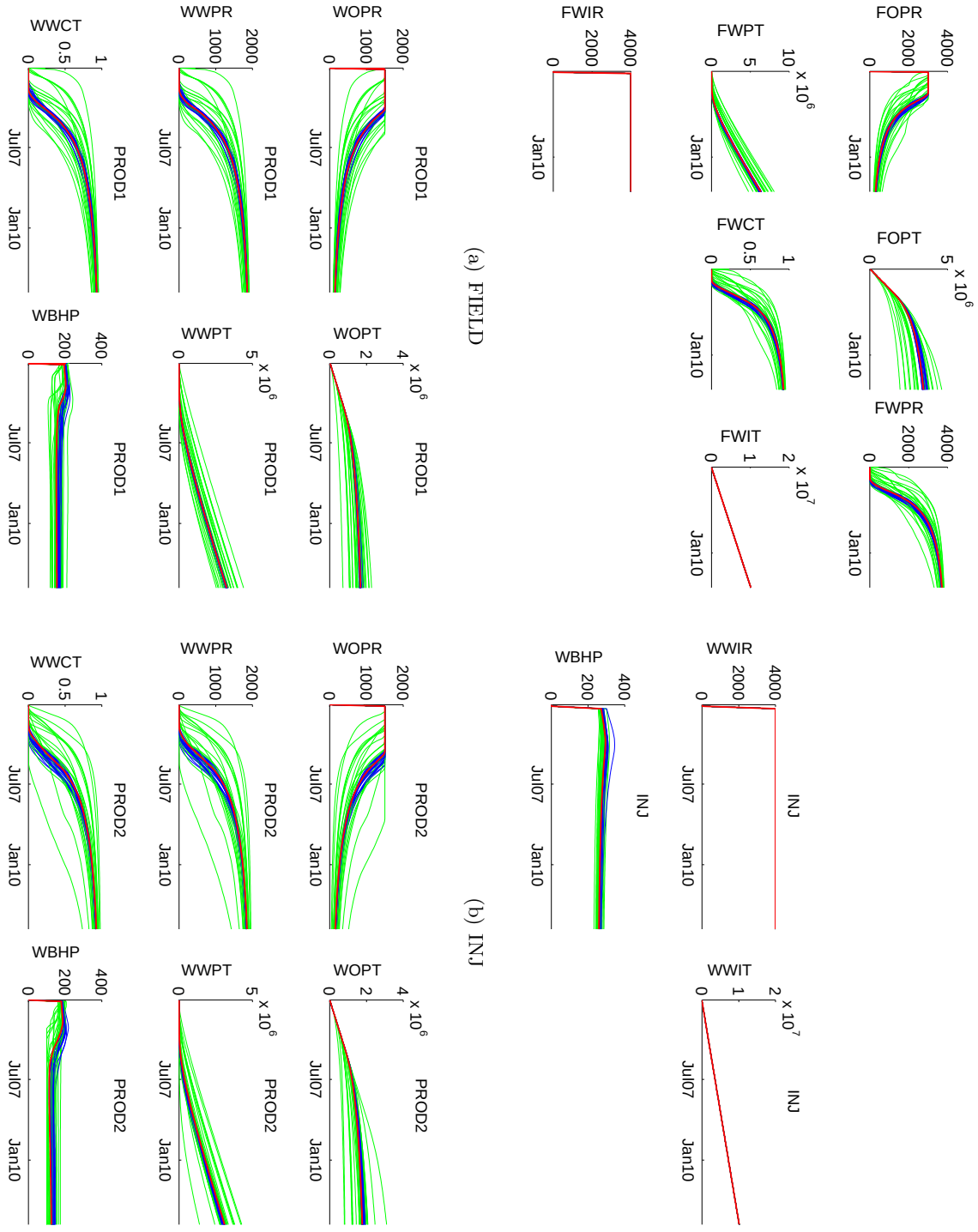
The history match for this simulation is shown in Figure E.1.

E.2 Simulation 2

The history match for this simulation is shown in Figure E.2.

E.3 Simulation 3

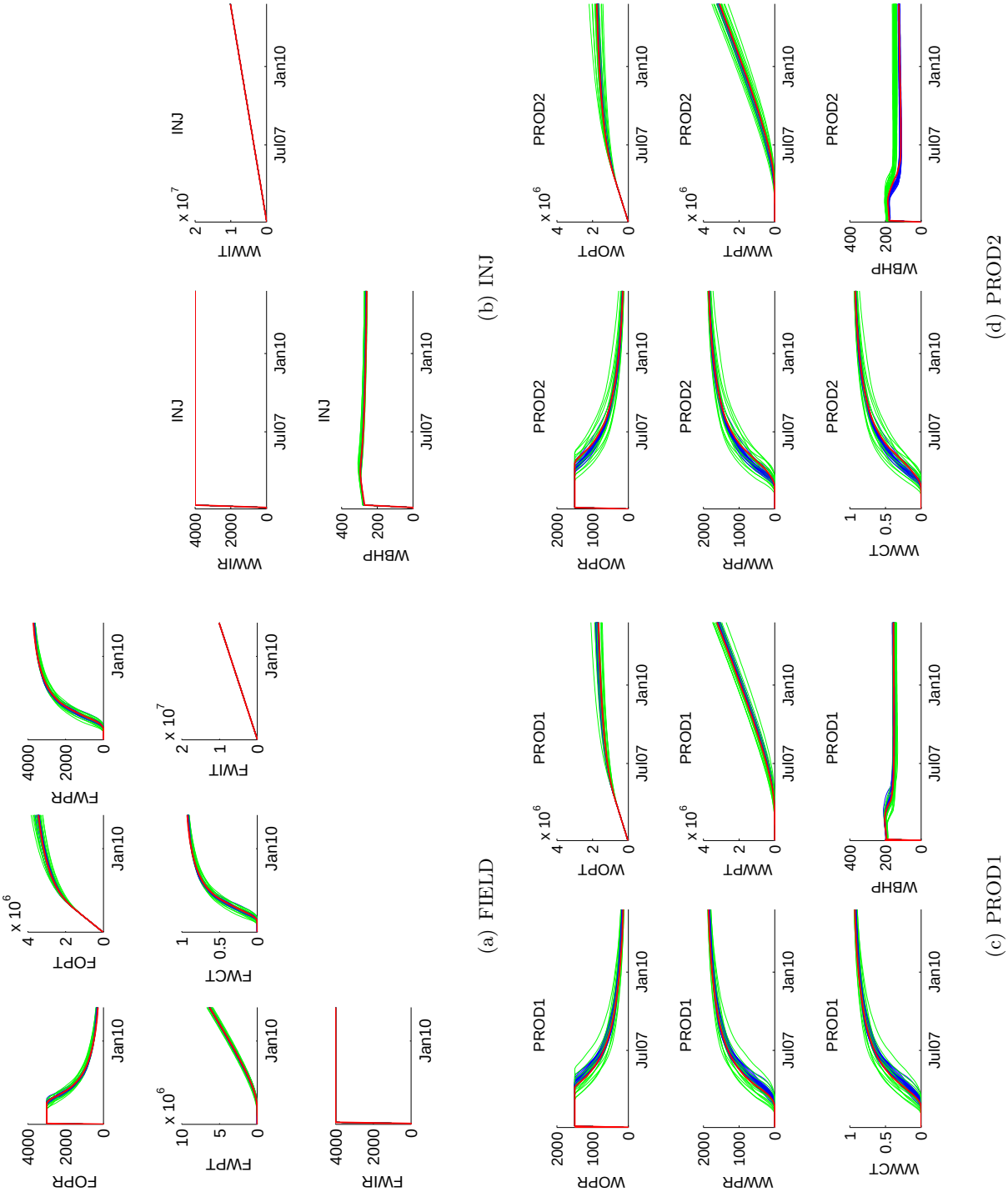
The history match for this simulation is shown in Figure E.3.



(a) FIELD (b) INJ

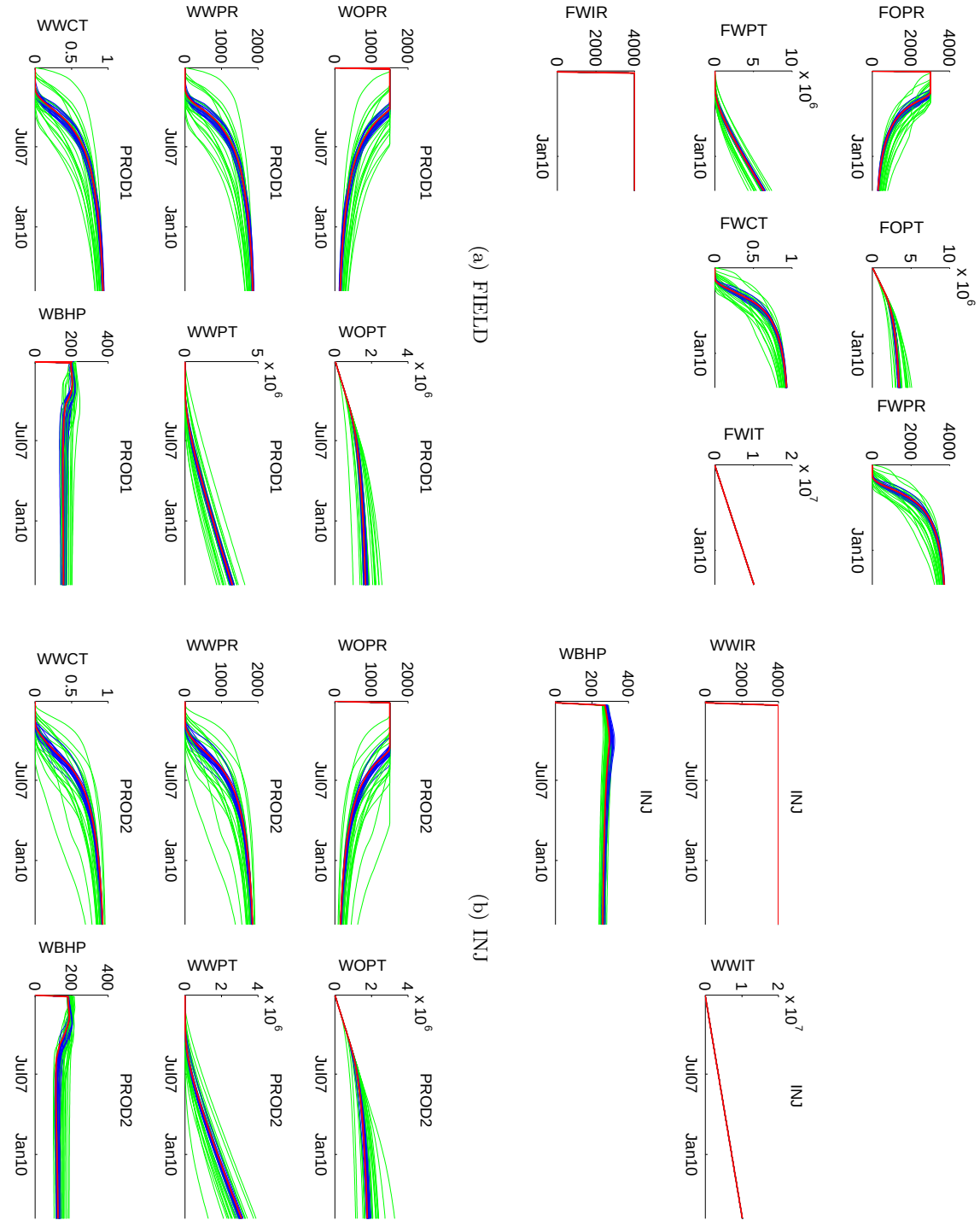
(c) PROD1 (d) PROD2

Figure E.1: History match results for simulation 1



(a) FIELD (b) INJ (c) PROD1 (d) PROD2

Figure E.2: History match results for simulation 2



(a) FIELD (b) INJ

(c) PROD1 (d) PROD2

Figure E.3: History match results for simulation 3

Appendix F

Parameter estimation results

THIS appendix contains the parameter estimation results from the simulations that were described in Chapter 7. Only a selection of the results is presented in this appendix. For the full overview containing both history matching and all the static parameters, see the Electronic Appendix (Appendix G)

F.1 Simulation 1 - PERMX for layer 3

The permeability in the x-direction for layer 3 from simulation 1 is shown in Figure F.1. Where the mean value, standard deviation and values for two ensemble members are plotted at various steps in time.

F.2 Simulation 2 - PERMX for layer 3

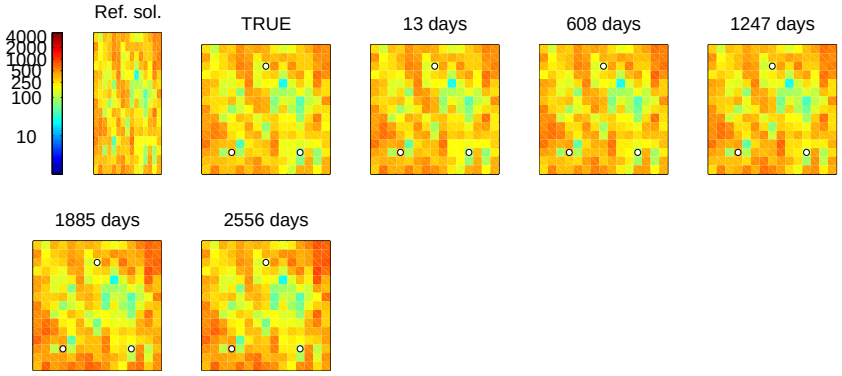
The permeability in the x-direction for layer 3 from simulation 2 is shown in Figure F.2. Where the mean value, standard deviation and values for two ensemble members are plotted at various steps in time.

F.3 Simulation 3 - PERMX for layer 3

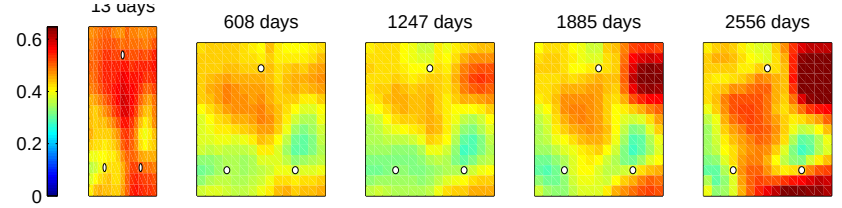
The permeability in the x-direction for layer 3 from simulation 3 is shown in Figure F.3. Where the mean value, standard deviation and values for two ensemble members are plotted at various steps in time.

F.4 Simulation 4 - PERMX for layer 3

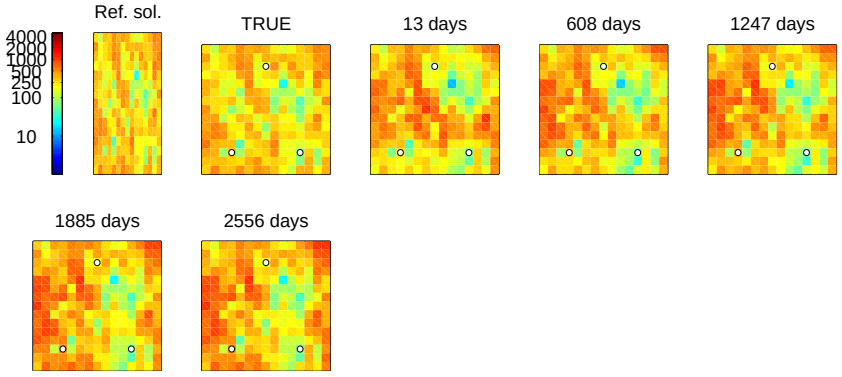
The permeability in the x-direction for layer 3 from simulation 4 is shown in Figure F.4. Where the mean value, standard deviation and values for two ensemble members are plotted at various steps in time.



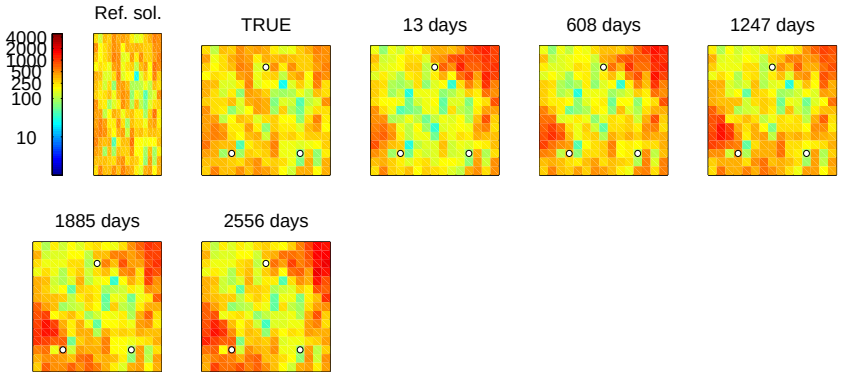
(a) PERMX mean value for layer 3



(b) PERMX stadard deviation for layer 3

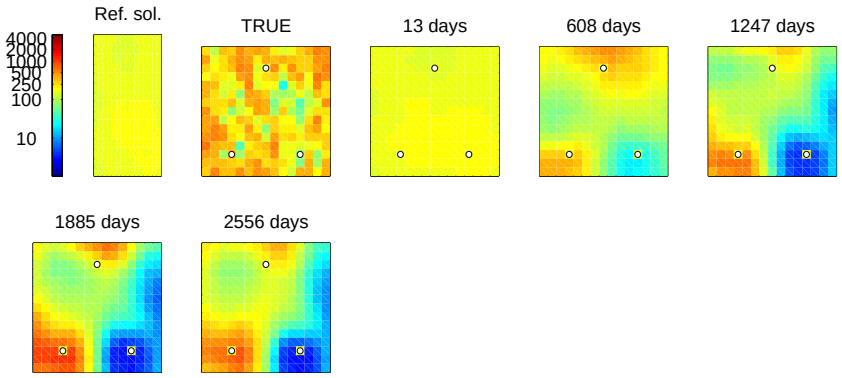


(c) PERMX for ensemble member 1 for layer 3

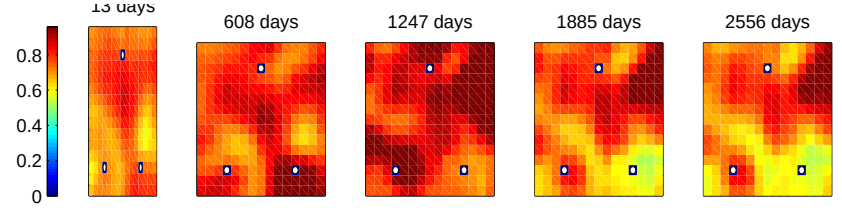


(d) PERMX for ensemble member 2 for layer 3

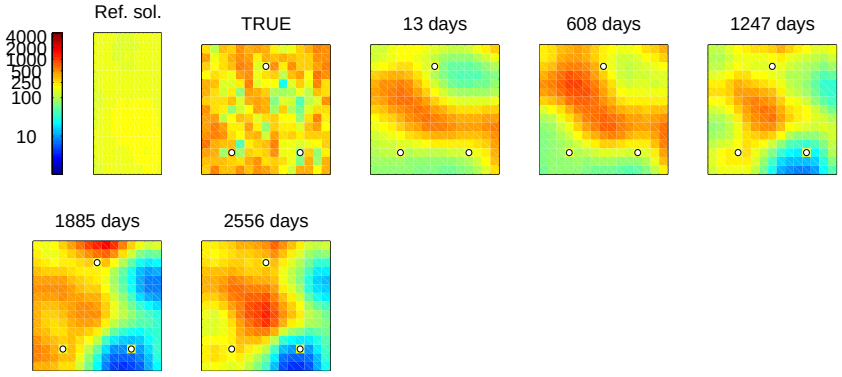
Figure F.1: Simulation 1 - PERMX for layer 3



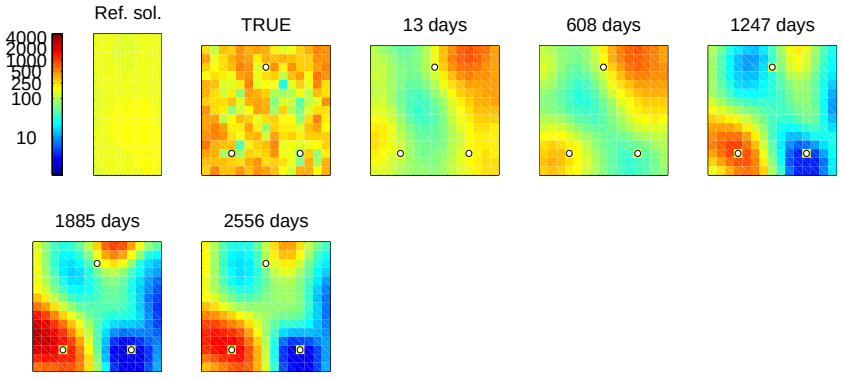
(a) PERMX mean value for layer 3



(b) PERMX stadard deviation for layer 3

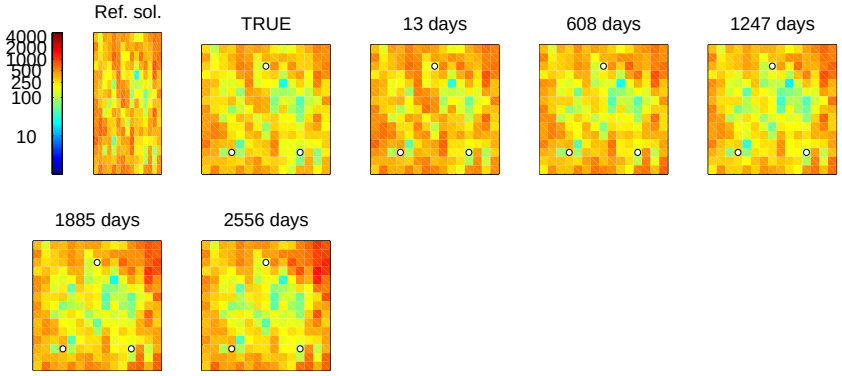


(c) PERMX for ensemble member 1 for layer 3

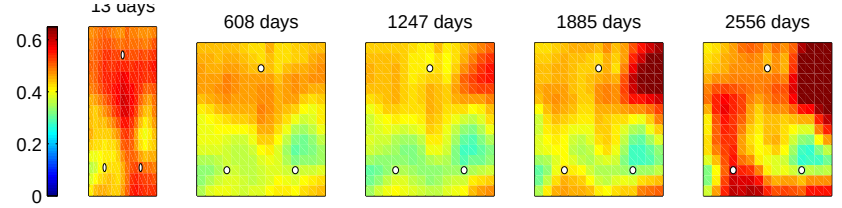


(d) PERMX for ensemble member 2 for layer 3

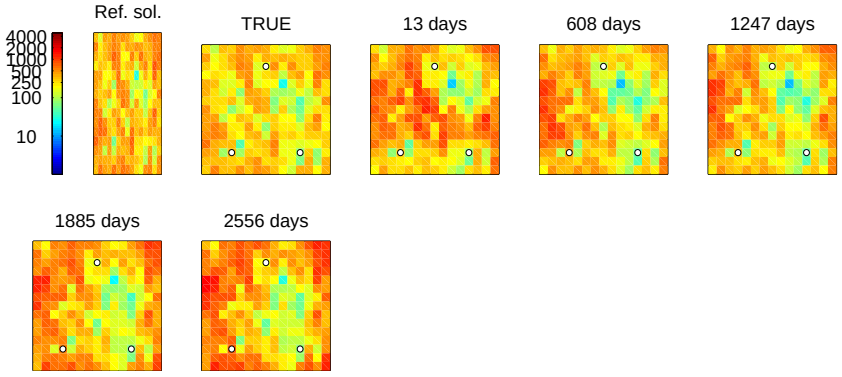
Figure F.2: Simulation 2 - PERMX for layer 3



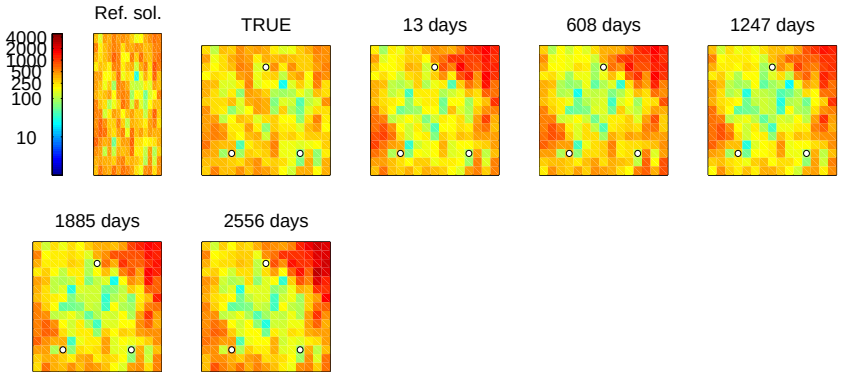
(a) PERMX mean value for layer 3



(b) PERMX stadard deviation for layer 3

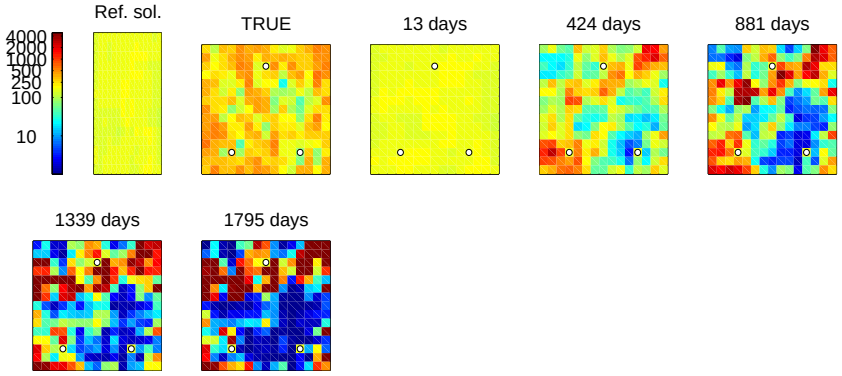


(c) PERMX for ensemble member 1 for layer 3

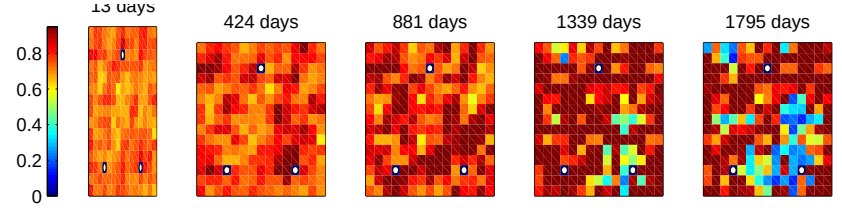


(d) PERMX for ensemble member 2 for layer 3

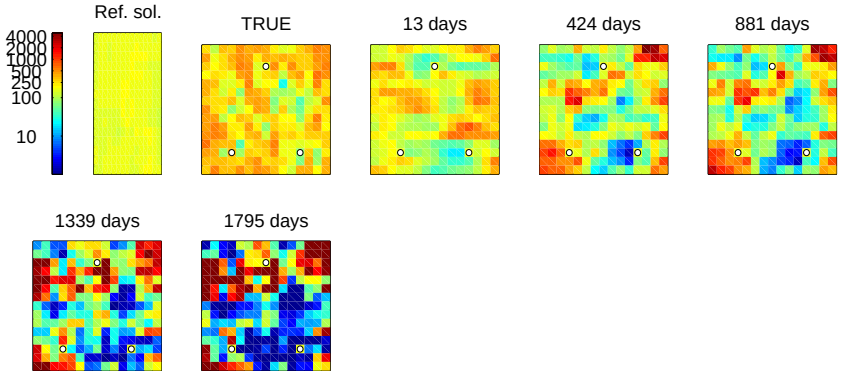
Figure F.3: Simulation 3 - PERMX for layer 3



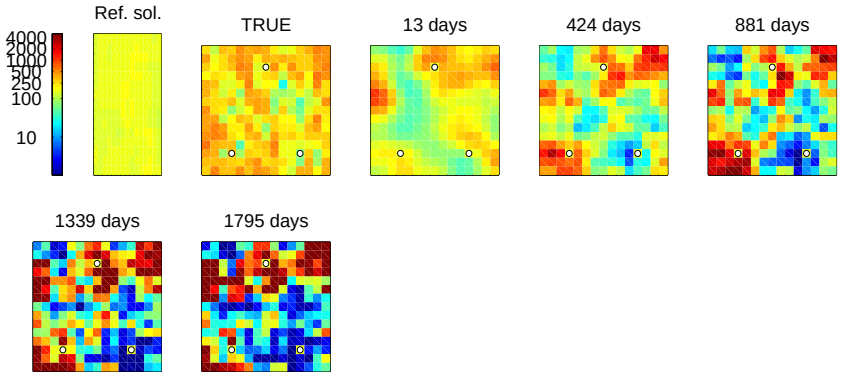
(a) PERMX mean value for layer 3



(b) PERMX stadard deviation for layer 3



(c) PERMX for ensemble member 1 for layer 3



(d) PERMX for ensemble member 2 for layer 3

Figure F.4: Simulation 4 - PERMX for layer 3

F.5 Simulation 2 and 3 - Average standard deviation in static parameters

In addition to Figure 7.9 in Chapter 7 the average standard deviation for the static parameters as a function of time is also plotted for simulation 2 and 3 in Figure F.5 and F.6

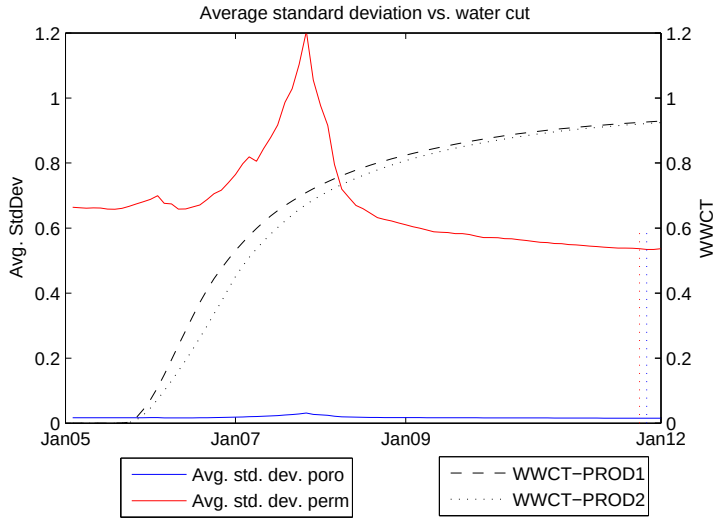


Figure F.5: Simulation 2 - Changes in the standard deviation

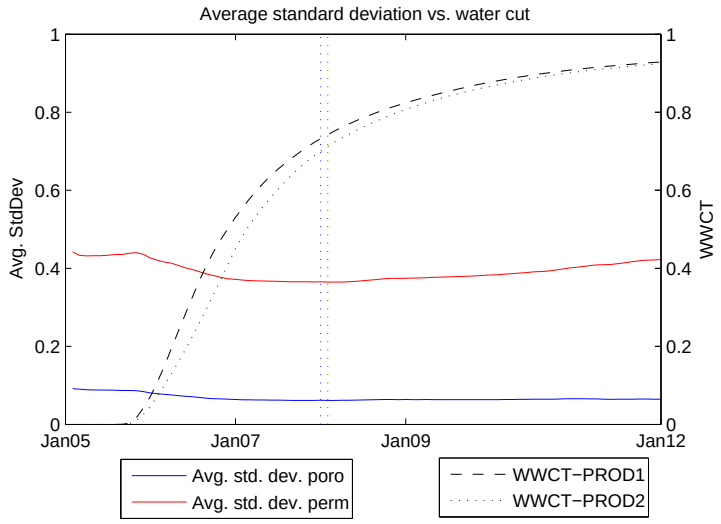


Figure F.6: Simulation 3 - Changes in the standard deviation

Appendix G

Electronic appendix

TOGETHER with this report a CD is supplied with some of the material that was used in the process. This appendix outlines the content of the Electronic Appendix and descriptions are shown where deemed necessary.

- Report
  - rapport.pdf *(Final version of this report in electronic format)*
- SimulationResults
  - Simulation1
    - Sim1_EnKFresults.pdf *(Full report file from Simulation 1)*
  - Simulation2
    - Sim2_EnKFresults.pdf *(Full report file from Simulation 2)*
  - Simulation3
    - Sim3_EnKFresults.pdf *(Full report file from Simulation 3)*
  - Simulation4
    - Sim4_EnKFresults.pdf *(Full report file from Simulation 4)*
  - Others
    - Init1_NewCorr_EnKFresults.pdf *(Full report file from Initialization 1 with new correlation)*
    - Init3_NewCorr_EnKFresults.pdf *(Full report file from Initialization 3 with new correlation)*