NTNU – Trondheim
Norwegian University of
Science and Technology

Ensemble Kalman Filter on the Brugge Field

Paul Vuong Vo

Master of Science in Physics and Mathematics
Submission date: June 2012
Supervisor: Karl Henning Omre, MATH

Norwegian University of Science and Technology
Department of Mathematical Sciences

Preface

This master thesis is the result of the course TMA4905 "Statistikk, Masteroppgave" at the Norwegian University of Science and Technology (NTNU). It was written during the final semester within the study program "Industriell matematikk" and has a value of 30 units.

The motivation for working with reservoir production simulation in the Ensemble Kalman Filter setting came from my supervisor, Henning Omre. Since the intention was to get an introduction into the use of statistical methodology in reservoir production simulation, I decided to change subject from previous project work from the course TMA4500 "Statistikk, Fordypningsprosjekt".

Much of the work on this thesis has consisted in the understanding and use of the Ensemble Reservoir Tool (Ert) developed by the Norwegian Computing Center (NR) and Statoil. This program was installed on the computer "pets" at the Petroleum Department at NTNU, so that it could be applied with the reservoir production simulation software Eclipse that was already installed there. The data from the Brugge Field was provided by Statoil. The results produced by Ert were further processed using Matlab R2012a.

During the work on this thesis I have attended the course TPG4160 Reservoir Simulation. This course, lectured by Jon Kleppe, has introduced me to the background of reservoir production simulation. Moreover, an introductory course in Eclipse was given.

I would like to express my gratitude to Professor Henning Omre for his help and guidance throughout the work on this thesis. Henning has provided me with the necessary information and helped me organize the work. I am also thankful for the help I have received from Jon Sætrom at Statoil. He has provided me with the necessary data from the Brugge Field, helped me setting up the ensemble reservoir tool (Ert), and assisted me with issues related to Ert. Finally, I would like to thank Per Kristian Hove at NTNU for helping me with Unix related issues and Inge Myrseth at the Norwegian Computing Center (NR) and Joakim Hove at Statoil for providing me support related to Ert.

Paul Vo Trondheim, July 1st, 2012

Abstract

The purpose of modeling a petroleum reservoir consists of finding the underlying reservoir properties based on production data, seismic and other available data. In recent years, progress in technology has made it possible to extract large amount of data from the reservoir frequently. Hence, mathematical models that can rapidly characterize the reservoir as new data become available gained much interest.

In this thesis we present a formulation of the first order Hidden Markov Model (HMM) that fits into the description of a reservoir model under production. We use a recursive technique that gives the theoretical solution to the reservoir characterization problem. Further, we introduce the Kalman Filter which serves as the exact solution when certain assumptions about the HMM are made. However, these assumptions are not valid when describing the process of a reservoir under production. Thus, we introduce the Ensemble Kalman Filter (EnKF) which has been shown to give an approximate solution to the reservoir characterization problem. However, the EnKF is depending on multiple realizations from the reservoir model which we obtain from the reservoir production simulator Eclipse. When the number of realizations are kept small for computational purposes, the EnKF has been shown to possibly give unreliable results. Hence, we apply a shrinkage regression technique (DR-EnKF) and a localization technique (Loc-EnKF) that are able to correct the traditional EnKF. Both the traditional EnKF and these corrections are tested on a synthetic reservoir case called the Brugge Field.

The results indicate that the traditional EnKF suffers from ensemble collapse when the ensemble size is small. This results in small and unreliable prediction uncertainty in the model variables. The DR-EnKF improves the EnKF in terms of root mean squared error (RMSE) for a small ensemble size, while the Loc-EnKF makes considerable improvements compared to the EnKF and produces model variables that seems reasonable.

Sammendrag

Modelleringen av et petroleumreservoar består av å estimere underliggende reservoarparametre basert på produksjonsdata, seismikk og andre typer data. I de senere årene har utviklingen av teknologien gjort det mulig å utvinne store mengder data fra reservoaret på kort tid. Derfor har matematiske modeller som raskt klarer å karakterisere reserservoaret når nye data blir tilgjengelige fått stor oppmerksomhet.

I denne oppgaven presenteres en formulering av en første ordens skjult Markov modell (HMM) som er godt tilpasset beskrivelsen av et reservoar under produksjon. Her brukes det en rekursiv teknikk til å gi den teoretiske løsningen av reservoarkarakterisingsproblemet. Videre introduserer vi Kalman-filteret som gir eksakt løsning under visse antakelser om Markov modellen. Disse antakelsene gjelder likevel ikke i beskrivelsen av prosessen av et reservoar under produksjon. Derfor innføres Ensemble Kalman Filter (EnKF) som har vist seg å være en tilnærmet løsning av reservoarkarakteriseringsproblemet. Metoden EnKF er avhengig av at mange realisasjoner genereres fra reservoarmodellen som fåes fra reservoar produksjonssimulatoren Eclipse. Når antallet realisasjoner er få på grunn av beregningsmessige årsaker gir EnKF upålitelige resultater. En mulig løsning på problemet er å bruke en shrinkage regressjonsteknikk (DR-EnKF) eller en lokaliseringsmetode (Loc-EnKF) som gjør det mulig å korrigere den tradisjonelle EnKF. Både den tradisjonelle EnKF og disse variantene av EnKF er testet på et syntetisk reservoar case kalt Bruggefeltet.

Resultatene indikerer at den tradisjonelle EnKF lider av at ensemblet kollapser når ensemble-størrelsen er liten. Dette resulterer i liten og urimelig usikkerhet i modellvariablene. Sammenlignet med EnKF gir DR-EnKF metoden forbedringer for en liten ensemble-størrelse under root mean squared error (RMSE)-kriteriet, mens Loc-EnKF metoden gir tydelige forbedringer og produserer modellvariable som synes å være rimelige.

Contents

1 Introduction

In the petroleum industry, the construction of a reservoir production simulation model is a highly necessary task when the purpose is to estimate the amount of hydrocarbons present in the reservoir. This task involves the characterization of model and state variables such as porosity, permeability, hydrocarbon saturation and pressure. When new production data from wellbores are obtained, these model and state variables must be tuned in order to fit the production data observed. The process of tuning model and state variables is in the literature termed history matching although the objective is not to match the observations exactly, but rather assimilate the observations in order to make the best future predictions.

Mathematically, the determination of model and state variables is regarded as a complex, ill-posed nonlinear inverse problem. Model variables are characterized as being static during a reservoir production simulation run, and may include porosity, permeability, and net-to-gross ratio. State variables are dynamic because they change during a reservoir production simulation run and may include saturation and pressure. Model and state variables define the state of the reservoir model. Reservoir information obtained from the wellbores are referred to as production data. Production data may include water production rate, oil production rate and well bottom hole pressure.

The characterization of model and state variables being an ill-posed problem means that there exist many reservoir models that are similarly consistent with the observations. By this reason, there is a high risk in using only one reservoir production simulation model because different reservoir production simulation models result in different forecasts. The complete solution should therefore include all consistent reservoir models, which entails an uncertainty in the model and state variables and hence uncertainty in the forecasts. The statistical approach to the problem of determining the model and state variables is Baysian inversion. In Baysian inversion, a priori knowledge about the model must be incorporated through a probability distribution (prior model) before any observation is assimilated. By determining a probabilistic term that connects the observations to the model variables (likelihood model), it is possible to obtain a posteriori knowledge about the model through a probability distribution (posterior model).

The process of continuously updating the reservoir model as new production data occur has in recent years gained much interest. This can be attributed to the improvement of computer power and development of mathematical models that can incorporate many variables. Traditional methods that provide a single reservoir production simulation model based on history production data has been outperformed because they require much computational work. New methods that are able to assimilate observations as soon as they become available has turned out to be much more efficient.

The Kalman Filter (KF) (Kalman et al., 1960) is a sequential Bayesian updating algorithm that handles models with many variables. It gives the analytical solution to the predicted state, when the model is Gauss-linear. In a Gauss-linear model the predicted state is linearly connected to the unpredicted state through a Gaussian distribution, and the observation is linearly connected to the corresponding

state through a Gaussian distribution. For petroleum reservoir modeling purposes however, the Gauss-linear assumption does not hold. The Kalman Filter has been modified to handle models that deviate from the Gauss-linear assumption. In the Extended Kalman Filter, a linearisation is made around the mean of variables that are nonlinear. Although it is useful in some applications, it produces unreliable results in highly nonlinear models such as petroleum reservoir models.

The Ensemble Kalman Filter (EnKF), introduced by Evensen (1994) is a sequential Bayesian updating algorithm that approximates the solution to the inverse problem using Monte Carlo simulation. It was first used in applications like oceanography and weather forecasting, and later in petroleum engineering (Lorentzen et al., 2001) and has since been much used for solving inverse problems. When the Gauss-linear assumption does not hold, the solution to the inverse problem is not analytically tractable. The EnKF is based on the idea of applying a sample of realizations, called an ensemble, in order to capture the important characteristics such as the mean and the covariance of the forecast distribution of the state. When considering petroleum reservoirs, each ensemble member represent a possible reservoir model state. The mean of the ensemble represents the forecast of the state, and the spread in the ensemble represent uncertainty in the forecast. For models that are Gauss-linear, the EnKF gives the solution to the forecast as produced by the Kalman Filter when the ensemble size tends to infinity. Hence, although an approximate sequential Monte Carlo algorithm the EnKF has a nice asymptotic property.

In Section 2 we start with some of the notation used in this thesis, followed by a formulation of a model that describes the dynamic process of a reservoir. This section introduces a sequential algorithm that suits well to a model that needs to be updated. Further, Section 3 introduces the traditional Kalman Filter algorithm which gives the analytical solution to the inverse problem in cases when the model is Gauss-linear. Section 4 describes the EnKF in the general case, and a special case that is suited to reservoir modeling. In Section 5 we present the EnKF algorithm in reservoir modeling in more detail. Here, we also discuss the solution in cases when the ensemble size is small. In Section 6 we describe the equations that control the process of a reservoir under production. This section is intended to give an idea of what is being solved when running the reservoir production simulator. The presented EnKF algorithm is tested on a synthetic well case, called the Brugge field in Sections 7 and 8, using the reservoir ensemble tool (Ert). Finally, Section 9 gives a summary of what is achieved with this thesis and outline further work.

2 Notation and Model Description

Throughout this thesis we will denote $a \in \mathbb{R}^{n_a}$ that a is a vertical vector of real entries of size n_a. We will denote $A \in \mathbb{R}^{m \times n}$ a matrix of real entries with m rows and n columns. Both a and A can include random entries. This will be clear from the context. We use the sign $'$ to denote the transpose of a vector or matrix. Thus, A$'$ is a matrix of dimension $n \times m$. Further, we will denote $a \sim f(a)$ that a follows a probability distribution $f(\cdot)$; and for the special case $a \sim N_{n_a}(\mu, \Sigma)$ that

a is Gaussian distributed of dimension n_a, with mean n_a-vector μ and covariance $(n_a \times n_a)$-matrix Σ.

We consider an unknown time series $[x_0, \ldots, x_T, x_{T+1}]$, where $x_t \in R^{n_x}; t = 0, \ldots, T+1$ are multidimensional random variables. The time series $[x_0, \ldots, x_T, x_{T+1}]$ defines a system that is evolving through time $t \in \{0, \ldots, T+1\}$. Thus, we call x_t the state of the system at time t. We assume that a related time series of observations $[d_1^o, \ldots, d_T^o]$ is available, with $d_t^o \in R^{n_d}; t = 1, \ldots, T$ being generated from the associated states. The current state of the system is x_T and the objective is to predict the next state x_{T+1} based on the given observations $[d_1^o, \ldots, d_T^o]$.

We model the time series of $[x_0, \ldots, x_T, x_{T+1}]$, by defining a prior model that is restricted by the Markov properties:

$$\begin{aligned} [x_0, \ldots, x_T, x_{T+1}] &\sim f(x_0, \ldots, x_T, x_{T+1}) \\ &= f(x_0) \prod_{t=0}^{T} f(x_{t+1}|x_0, \ldots, x_t) \\ &= f(x_0) \prod_{t=0}^{T} f(x_{t+1}|x_t). \end{aligned} \tag{1}$$

The second line follows from successive decomposition, while the last line follows from the first order Markov property, which states that each state given the past is only dependent on the previous state. Moreover, we assume the initial distribution $f(x_0)$, and the transition functions $f(x_{t+1}|x_t), t = 0, \ldots, T$ to be known. Thus we have a model for the prior distribution that has an underlying first order Markov property.

Further, we model the connection between observations and states by a likelihood model. The likelihood model defines a probabilistic term of $[d_1^o, \ldots, d_T^o]$ given $[x_0, \ldots, x_T, x_{T+1}]$. We assume two properties on the likelihood model; conditional independence and single state dependence:

$$\begin{aligned} [d_1^o, \ldots, d_T^o|x_0, \ldots, x_T, x_{T+1}] &\sim f(d_1^o, \ldots, d_T^o|x_0, \ldots, x_T, x_{T+1}) \\ &= \prod_{t=1}^{T} f(d_t^o|x_0, \ldots, x_T, x_{T+1}) \\ &= \prod_{t=1}^{T} f(d_t^o|x_t), \end{aligned} \tag{2}$$

where $f(d_t^o|x_t); t = 1, \ldots, T$ are known likelihood functions. Hence, the likelihood model assumes that observation at each time point is independent of other observations once the associated state is known. Together, the prior model and the likelihood model form a hidden Markov model that is displayed in Figure 1. The arrows describes latent dependencies between the nodes, and the dependencies are determined by likelihood functions and transition functions. Using Bayesian inversion we get the posterior model:

$$
\begin{aligned}
[x_0,\ldots,x_T,x_{T+1}|d_1^o,\ldots,d_T^o] &\sim f(x_0,\ldots,x_T,x_{T+1}|d_1^o,\ldots,d_T^o)\\
&= \text{const}\times f(d_1^o,\ldots,d_T^o|x_0,\ldots,x_T,x_{T+1})\\
&\quad \times f(x_0,\ldots,x_T,x_{T+1})\\
&= \text{const}\times \prod_{t=1}^{T} f(d_t^o|x_t)f(x_0)\prod_{s=0}^{T} f(x_{s+1}|x_s)\\
&= \text{const}\times f(x_0)\left[\prod_{t=1}^{T} f(d_t^o|x_t)f(x_t|x_{t-1})\right]\\
&\quad \times f(x_{T+1}|x_T), \qquad (3)
\end{aligned}
$$

where 'const' is a normalizing constant. This normalizing constant is usually hard to assess when the state dimension is large. Thus, the posterior model is rarely analytically obtainable. But remember that the objective is to forecast x_{T+1} based on $d_1^o,\ldots,d_T^o$. This is obtained by the forecast distribution:

$$
f(x_{T+1}|d_1^o,\ldots,d_T^o) = \int\ldots\int f(x_0,\ldots,x_T,x_{T+1}|d_1^o,\ldots,d_T^o)dx_0\ldots dx_T. \qquad (4)
$$

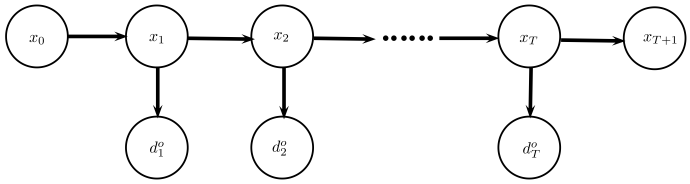

Figure 1: A picture of the first order Markov model.

This forecast distribution can be assessed by a recursive algorithm. For convenience, we introduce the notation

$$
\begin{aligned}
x_0^a &= x_0,\\
x_t^a &= [x_t|d_1^o,\ldots,d_t^o]; \quad t=1,\ldots,T.\\
x_1^f &= x_1,\\
x_{t+1}^f &= [x_{t+1}|d_1^o,\ldots,d_t^o]; \quad t=1,\ldots,T. \qquad (5)
\end{aligned}
$$

Here, a and f denotes the assimilated and forecast outcomes, respectively. Following is a description of the recursive algorithm:

Algorithm 1 Recursive forecasting

Initiate: $x_0 \sim f(x_0)$
for $t = 1, \ldots, T$ **do**
  Forecasting:
  $x_t^f \sim f(x_t) = \int f(x_t|x_{t-1}^a) f(x_{t-1}^a) dx_{t-1}^a$
  Assimilate:
  $x_t^a \sim f(x_t^a) = \text{const} \times f(d_t^o|x_t^f) f(x_t^f)$
end for
Last forecast:
$x_{T+1}^f = [x_{T+1}|d_1^o, \ldots, d_T^o] \sim f(x_{T+1}^f) = \int f(x_{T+1}|x_T^a) f(x_T^a) dx_T^a$

The recursive algorithm in Algorithm 1, sequentially updates the state of the system by following a forecast operation followed by an assimilation operation. This sequential updating process makes it possible to compute the forecast distribution. The recursive algorithm is depending on the prior model and the likelihood model. We define the prior model:

$$\begin{aligned} x_0 &\sim f(x_0), & (6)\\ [x_{t+1}|x_t] &= \omega(x_t, \epsilon_t^x) \sim f(x_{t+1}|x_t). & (7) \end{aligned}$$

Here $\epsilon_t^x \sim N_{n_x}(0, I_{n_x})$ is a stochastic term or factor that represents the model error, when state x_t is propagated to state x_{t+1}. This term is often separated from the transition function ω, with the additional assumption $\epsilon_t^x \sim N_{n_x}(0, \Sigma_t^x)$. In the general case, $\omega : (R^{n_x} \times R^{n_x}) \to R^{n_x}$ is a known function. We defines the likelihood model by means of functions ζ:

$$[d_t^o|x_t] = \zeta(x_t, \epsilon_t^d) \sim f(d_t^o|x_t), \quad (8)$$

where $\epsilon_t^d \sim N_{n_d}(0, I_{n^d})$ is a stochastic term or factor that represents the observation error. Often, the likelihood function ζ is independent ϵ_t^d, with the additional assumption $\epsilon_t^d \sim N_{n_d}(0, \Sigma_t^d)$. Generally, $\zeta : (R^{n_x} \times R^{n_d}) \to R^{n_d}$ is a known function. The prior and likelihood model might be chosen arbitrarily, but a reference model choice is a linear and Gaussian model, termed the Gauss-linear model. It is defined as:

$$\begin{aligned} x_0 \sim f(x_0) &= N_{n_x}(\mu_0^x, \Sigma_0^x),\\ [x_t|x_{t-1}] = A_t x_{t-1} + \epsilon_t^x \sim f(x_t|x_{t-1}) &= N_{n_x}(A_t x_{t-1}, \Sigma_t^x), & (9)\\ [d_t^o|x_t] = H_t x_t + \epsilon_t^d \sim f(d_t^o|x_t) &= N_{n_d}(H_t x_t, \Sigma_t^d). \end{aligned}$$

Here, μ_0^x and Σ_0^x are assumed known. Also the matrices A_t, Σ_t^x, H_t, and Σ_t^d, are known for all times and the model and observation error terms are independent of the state. Under this Gauss-linear model assumption, the forecast distribution is analytically tractable. This solution corresponds to the Kalman Filter, which is discussed next.

3 The Traditional Kalman Filter (KF)

The KF is a recursive algorithm for assessing the forecast distribution using the recursive algorithm in Algorithm 1, and assuming a Gauss-linear model as given by Eq. (9). The use of the recursive algorithm ensures that Gaussianity is preserved from one time step to the next. This is due to the property of the Gaussian distribution being closed under linear operations. Hence, the predictive distribution becomes Gaussian as well. A description of the Kalman filter algorithm is given in Algorithm 2.

Algorithm 2 The Kalman Filter algorithm

Initiate: $x_0^a \sim f(x_0^a) = N_{n_x}(\mu_0^a, \Sigma_0^a)$
$\mu_0^a = \mu_0^x$
$\Sigma_0^a = \Sigma_0^x$
for $t = 1, \ldots, T$ **do**
  Forecasting:
  $x_t^f \sim f(x_t^f) = N_{n_x}(\mu_t^f, \Sigma_t^f)$
  $\mu_t^f = A_t \mu_{t-1}^a$
  $\Sigma_t^f = A_t \Sigma_{t-1}^a A_t^{'} + \Sigma_{t-1}^x$
  Assimilating:
  $x_t^a \sim f(x_t^a) = N_{n_x}(\mu_t^a, \Sigma_t^a)$
  $\mu_t^a = \mu_t^f + \Sigma_t^f H_t^{'}[H_t \Sigma_t^f H_t^{'} + \Sigma_t^d]^{-1}(d_t^o - H_t \mu_t^f)$
  $\Sigma_t^a = \Sigma_t^f - \Sigma_t^f H_t^{'}[H_t \Sigma_t^f H_t^{'} + \Sigma_t^d]^{-1} H_t \Sigma_t^f$
end for
Last forecast:
$x_{T+1}^f = [x_{T+1}|d_1, \ldots, d_T] \sim f(x_{T+1}^f) = N_{n_x}(\mu_{T+1}^f, \Sigma_{T+1}^f)$
$\mu_{T+1}^f = A_T \mu_T^a$
$\Sigma_{T+1}^f = A_T \Sigma_T^a A_T^T + \Sigma_T^x$

The Kalman filter algorithm produces both forecast outcome x_t^f unconditioned on observation at the current time step, and assimilated outcome x_t^a where the current observation is taken into account. Moreover, the result is analytically tractable and no approximations are made, assuming that the model is Gauss-linear only. However, when the model deviates from being Gauss-linear, approximations can be made. Extensions to the Kalman Filter has been proposed, such as the Extended Kalman Filter (Jazwinski, 1970) and the Unscented Kalman Filter (Julier and Uhlmann, 1997) which work on nonlinear systems. In nonlinear systems, nonlinearity is attributed to the prior model, the likelihood model or both. However, we will focus on a simulation based approach that also provides an approximation to the forecasting problem when considering nonlinear systems, namely the ensemble Kalman filter.

4 The Ensemble Kalman Filter (EnKF)

The ensemble Kalman filter is a sequential Monte Carlo algorithm that approximates the forecast distribution. The idea in EnKF is to generate a set of realizations of size n_e, called the ensemble that is propagated through the model equations. The EnKF is an approximate solution both when the model is Gauss-linear, and when deviation from these assumptions occur. When the model is Gauss-linear the EnKF algorithm converges towards the exact KF solution when $n_e \to \infty$. In EnKF, the ensemble is propagated by the forward model. Then ensemble members are adjusted as observations occur. At time $T+1$, the ensemble is used to assess the forecast distribution $f(x_{T+1}|d_1^o, \ldots, d_T^o)$. We define the time series of ensembles as

$$(x_t^{f(i)}, d_t^{(i)}) = (x_t^f, d_t)^{(i)}, \quad i \in \{1, \ldots, n_e\}; \quad t = 0, \ldots, T+1, \tag{10}$$

where n_e is the ensemble size and t is the time step. Here, $x_t^{f(i)} = [x_t|d_1^o, \ldots, d_{t-1}^o]$ represents approximate realizations from $f(x_t|d_1^o, \ldots, d_{t-1}^o)$ while $d_t^{(i)}$ represents realizations from the likelihood model. These realizations $d_t^{(i)}$ are associated with the observation d_t^o. At each time step, we define a covariance matrix between the current state x_t^f and observation d_t^o. With t omitted this covariance matrix is

$$\Sigma_{xd} = \begin{bmatrix} \Sigma_x & \Gamma_{xd} \\ \Gamma_{dx} & \Sigma_d \end{bmatrix} \in \mathbb{R}^{(n_x+n_d)\times(n_x+n_d)}, \tag{11}$$

where n_x and n_d are the dimension of the state vector and observation vector, respectively. Using the current ensemble in Eq. (10) we can estimate this covariance matrix, by estimating the unknown parameters Σ_x, Γ_{xd} and Σ_d. If we define a matrix X holding the centered state vectors and a matrix D holding the centered prediction data as

$$X = \left\{ x_t^{f(1)} - \hat{\mu}_x, \cdots, x_t^{f(n_e)} - \hat{\mu}_x \right\}, \tag{12}$$

$$D = \left\{ d_t^{(1)} - \hat{\mu}_d, \cdots, d_t^{(n_e)} - \hat{\mu}_d \right\}, \tag{13}$$

$$\hat{\mu}_x = \frac{1}{n_e} \sum_{i=1}^{n_e} x_t^{f(i)},$$

$$\hat{\mu}_d = \frac{1}{n_e} \sum_{i=1}^{n_e} d_t^{(i)},$$

the unknown parameters in the covariance matrix can be easily estimated by these estimators:

$$
\begin{aligned}
\hat{\Sigma}_x &= \frac{1}{n_e - 1} XX', \\
\hat{\Gamma}_{xd} &= \frac{1}{n_e - 1} XD', \\
\hat{\Sigma}_d &= \frac{1}{n_e - 1} DD',
\end{aligned} \tag{14}
$$

which are consistent when $n_e \to \infty$. The EnKF algorithm is based on the same procedure as the recursive algorithm. A description of the EnKF algorithm in its general form is given in Algorithm 3.

Algorithm 3 The EnKF algortihm

Initialize the ensemble: $x_0^{a(i)} \sim f(x_0); \quad i = 1, \ldots, n_e$
for $t = 1, ..., T$ **do**
Forecasting:
$\epsilon_{t-1}^{x(i)} \sim N_{n_x}(0, \Sigma_{t-1}^x); \quad i = 1, \ldots, n_e$
$x_t^{f(i)} = \omega(x_{t-1}^{a(i)}, \epsilon_{t-1}^{x(i)}); \quad i = 1, \ldots, n_e$
$\epsilon_t^{d(i)} \sim N_{n_x}(0, \Sigma_t^d); \quad i = 1, \ldots, n_e$
$d_t^{(i)} = \zeta(x_i^{f(i)}, \epsilon_t^{d(i)}); \quad i = 1, \ldots, n_e$
$e_t = \left\{ (x_t^f, d_t)^{(i)} \right\}; \quad i = 1, \ldots, n_e$
Assimilating:
Estimate Σ_{xd} from the ensemble $e_t \to \hat{\Sigma}_{xd}$, see Eq. (14)
$x_t^{a(i)} = x_t^{f(i)} + \hat{\Gamma}_{xd} \hat{\Sigma}_d^{-1} (d_t^o - d_t^{(i)}); \quad i = 1, \ldots, n_e$
end for
Last prediction:
$\epsilon_T^{x(i)} \sim N_{n_x}(0, \Sigma_T^x); \quad i = 1, \ldots, n_e$
$x_{T+1}^{f(i)} = \omega(x_T^{a(i)}, \epsilon_T^{x(i)}); \quad i = 1, \ldots, n_e$
$x_{T+1}^{f(i)} \sim f(x_{T+1} | d_1^o, \ldots, d_T^o); \quad i = 1, \ldots, n_e$

The last line of the EnKF algorithm in Algorithm 3 indicates that we can assess the forecast distribution $f(x_{T+1}|d_1, \ldots, d_T)$ from the ensemble members $x_{T+1}^{f(i)}; i = 1, \ldots, n_e$. For instance, reasonable estimates of the forecast and the covariance of the forecast are:

$$
\begin{aligned}
\hat{\mu}_{T+1} &= \frac{1}{n_e} \sum_{i=1}^{n_e} x_{T+1}^{f(i)}, \\
\hat{\Sigma}_{T+1} &= \frac{1}{n_e - 1} \sum_{i=1}^{n_e} (x_{T+1}^{f(i)} - \hat{\mu}_{T+1})(x_{T+1}^{f(i)} - \hat{\mu}_{T+1})'.
\end{aligned} \tag{15}
$$

But also the full empirical forecast distribution can be evaluated using confidence intervals when considering a skew forecast distribution. Essential to the EnKF

algorithm is the forecasting step and the assimilation step. In the forecasting step we apply the transition function ω on the ensemble members. In the assimilation step we correct the ensemble members in a linear manner by weights estimated from the ensemble. There are two basic assumptions made in the algorithm. First, the initial ensemble is supposed to represent the initial distribution $f(x_0)$. In cases where the dimension of x_0 is large, this imply that a large ensemble is needed to fairly represent the initial distribution. Second, the analyzing step is based on the assumption that the joint distribution of the forecast state $x_t^{f(i)}$ and the realization $d_t^{(i)}$ is Gaussian. When large deviations from Gaussian prior models and or Gaussian likelihood models occur, we may obtain unreliable results. However, if these two basic assumptions are reasonable to make, the EnKF algorithm provides a reliable approximate solution to the forecast distribution. Moreover, under Gauss-linear models the EnKF algorithm is consistent in the sense that the approximation converges to the exact solution when the ensemble size $n_e \to \infty$.

4.1 Special Case: EnKF With Gauss-linear Likelihood

We consider a hidden Markov model where the prior model is non-linear (i.e non-Gauss-linear) with an additive model error, and a likelihood model that is Gauss-linear. This is frequently used in applications. The assumptions made in this special case are:

$$\begin{aligned} x_0 &\sim f(x_0) \\ [x_t|x_{t-1}] = \omega(x_{t-1}) + \epsilon_t^x &\sim f(x_t|x_{t-1}), \\ [d_t^o|x_t] = H_t x_t + \epsilon_t^d &\sim f(d_t^o|x_t), \end{aligned} \tag{16}$$

where $\epsilon_t^x \sim N_{n_x}(0, \Sigma_t^x)$ and $\epsilon_t^d \sim N_{n_d}(0, \Sigma_t^d)$ are both known. The transition function ω might be a differential equation, or has a complex functionality. Under the special case, we define the time series of ensembles:

$$e_t = \left\{ x_t^{f(i)}; \quad i = 1, \ldots, n_e \right\}; \quad t = 1, \ldots, T+1. \tag{17}$$

Here, we note that the ensemble is defined differently from Eq. (10) because the cross covariance $\hat{\Gamma}_{xd} = \hat{\Sigma}_x H'$ at every time step can be assessed from the covariance matrix $\hat{\Sigma}_x$. The algorithm for this case appears as:

Algorithm 4 The EnKF algortihm with Gauss-linear likelihood

Initialize:
for $t = 1, ..., T$ **do**
Forecasting:
$\epsilon_{t-1}^{x(i)} \sim N_{n_x}(0, \Sigma_{t-1}^{x}); \quad i = 1, \ldots, n_e$
$x_t^{f(i)} = \omega(x_{t-1}^{a(i)}, \epsilon_{t-1}^{x(i)}); \quad i = 1, \ldots, n_e$
$e_t = \left\{ x_t^{(i)} \right\}; \quad i = 1, \ldots, n_e$
Analyzing:
Estimate Σ_x from the ensemble e_t
$\epsilon_t^{d(i)} \sim N_{n_d}(0, \Sigma_t^d) i = 1, \ldots, n_e$
$d_t^{(i)} = H_t x_t^{f(i)} + \epsilon_t^{d(i)} i = 1, \ldots, n_e$
$x_t^{a(i)} = x_t^{f(i)} + \hat{\Sigma}_x H_t^{'} [H_t \hat{\Sigma}_x H_t^{'} + \Sigma_t^d]^{-1} (d_t^o - d_t^{(i)}); \quad i = 1, \ldots, n_e$
end for
Last forecast:
$\epsilon_T^{x(i)} \sim N_{n_x}(0, \Sigma_T^x); \quad i = 1, \ldots, n_e$
$x_{T+1}^{f(i)} = \omega(x_T^{a(i)}, \epsilon_T^{x(i)}); \quad i = 1, \ldots, n_e$
$x_{T+1}^{f(i)} \sim f(x_{T+1} | d_1^o, \ldots, d_T^o); \quad i = 1, \ldots, n_e$

In Algorithm 4, the EnKF algorithm with Gauss-linear likelihood yields in the last forecast, an approximate sample from the forecast distribution $f(x_{T+1}|d_1, \ldots, d_T)$. In the case of reservoir modeling, the state vector of each ensemble member $x_t^{(i)}$ is often replaced by an augmented state vector $y_t^{(i)}$ to account for the simulated observations $d_t^{(i)}$. Thus, we have

$$y_t^{f(i)} = \begin{bmatrix} x_t^{f(i)} \\ d_t^{(i)} \end{bmatrix}, \tag{18}$$

where $d_t^{(i)}$ is generated from a non-linear function with observation error included. The relation that connects the state to the observations for this augmented case is a linear function, namely the likelihood

$$d_t^{(i)} = I_d \cdot y_t^{f(i)} \tag{19}$$

where I_d is a matrix of size $n_d \times (n_x + n_d)$ consisting of zero entries and an identity matrix of size n_d on the right. Hence I_d has the property of extracting the simulated observations from the augmented state matrix. The matrices $\hat{\Gamma}_{yd}$ and $\hat{\Sigma}_d$ are easy to compute because of the linear relationship. Thus, the corresponding assimilation step for the augmented state, is

$$y_t^{a(i)} = y_t^{f(i)} + \hat{\Sigma}_{y_t^f} I_d^{'} \left(I_d \hat{\Sigma}_{y_t^f} I_d^{'} \right)^{-1} (d_t^o - d_t^{(i)}) \tag{20}$$

where $\hat{\Sigma}_{y^f}$ is the covariance matrix of the augmented state matrix of size $n_x + n_d$. In fact, it is equal to $\hat{\Sigma}_{xd}$ in Eq. (14). Noting that the original state vector is

$x^{a(i)} = I_x y^{a(i)}$, with I_x a $n_x \times n_y$ matrix with an identity matrix of size n_x on the left and zero elsewhere, we multiply both sides of Eq. (20) by I_x. The matrix I_x extracts the original state vector from the augmented state vector. Thus, we get

$$x_t^{a(i)} = x_t^{f(i)} + I_x \hat{\Sigma}_{y^f} I_d^{'} \left(I_d \hat{\Sigma}_{y^f} I_d^{'} \right)^{-1} (d_t^o - d_t^{(i)}). \tag{21}$$

Further we note that $I_x \hat{\Sigma}_{y^f} I_d^{'} = \hat{\Gamma}_{xd}$, $I_d \hat{\Sigma}_{y^f} I_d^{'} = \hat{\Sigma}_d$ and $I_d y^{f(i)} = d_t^{(i)}$. Hence, we are back to the original assimilation scheme

$$x_t^{a(i)} = x_t^{f(i)} + \hat{\Gamma}_{xd} \hat{\Sigma}_d^{-1} (d_t^o - d_t^{(i)}), \tag{22}$$

given in Algorithm 3, where the likelihood is nonlinear. The two schemes are therefore identical.

4.2 Limitations

The EnKF relies on the assumption that the forecast state and the associated observation is jointly Gaussian. Hence, a large deviation from this assumption does not produce a good estimate for the forecast distribution. Alternative filter methods have been proposed that provides better estimates in non-Gaussian cases. The particle filter (Doucet et al., 2001) is known to give a better approximation to the forecast distribution in small scale problems. Moreover, it gives the asymptotically correct solution for all HMM models when $n_e \to \infty$. The randomized likelihood filter (Oliver, 1996) has been shown to give better approximations when the forecast distribution is multimodal, and it has the same asymptotic property as the EnKF. However, when considering reservoir evaluation problems, the EnKF has been proven to be a useful method.

The robustness of the EnKF method relies on its ability to capture the important properties of the forecast distribution that lies in the initial distribution $f(x_0)$. If the initial ensemble is able to represent $f(x_0)$ appropriately, the EnKF will be able to approximate the forecast distribution. However, when the ensemble size is small, it will not represent $f(x_0)$ sufficiently. This may lead to a poor estimate of the Kalman gain matrix $\hat{\Gamma}_{xd} \hat{\Sigma}_d^{-1}$, which is seen long range correlations between observation and model and state variables that are not real. The result is incorrect updates for model and state variables. In this thesis, we will improve the estimate of the Kalman gain matrix, using two well-known techniques. First, we use dimension reduction as explained in Sætrom and Omre (2011). Secondly, we apply localization (Anderson, 2006) that restricts the model update to occur locally to the observation only.

5 Reservoir Production Simulation

Before describing the reservoir modeling in the EnKF setting in more detail, we need to introduce the reservoir production simulator. In a reservoir production simulator, numerical methods are used to solve the partial differential equations that

describes the fluid flow (oil, water, gas) in a porous medium such as a petroleum reservoir. A reservoir production simulator is used in oil and gas companies in order to build a reservoir model that can assist in the development of new oil or gas fields. For instance, a reservoir model can help deciding the number of producer and injector wells that are needed and their locations in the reservoir. For existing fields a reservoir model can assist in predicting future reservoir performances. We will in this section give a brief derivation to the fluid flow equations.

Generally, flow in a porous medium is determined by mass conservation, momentum and energy conservation equations, and assisting equations for the fluids and the porous medium. If we assume that the temperature in the reservoir remains constant, we do not need to involve momentum and energy conservation. For a slab with constant cross section area, as shown in Figure 2, the mass conservation equations of a multiphase fluid flow appear as

$$\nabla \cdot (\rho_l \boldsymbol{u}_l) - q_l' = \frac{\partial(\phi \rho_l s_l)}{\partial t}, \quad l = \{\text{oil, water, gas}\}. \tag{23}$$

Here, ϕ denotes the porosity and ρ_l the density of the fluid. The saturation of the fluid is denoted by s_l, the fluid velocity vector is represented by $\boldsymbol{u}_l$, and q_l' is a source/sink term representing the mass flow rate per unit volume (injector/producer). If we make the assumptions that the fluid flows with low velocity and that the medium is isotropic, Darcy's law gives the relationship between the velocity field and the pressure, namely

$$\boldsymbol{u}_l = \frac{\kappa_a \kappa_{rl}}{\mu_l} \nabla p_l, \quad l = \{\text{oil, water, gas}\}, \tag{24}$$

where κ_a is the absolute permeability, κ_{rl} is the relative permeability, μ_l is the viscosity, and p_l is the pressure. Inserting Darcy's equation (24) into the mass conservation equation (23) yields

$$\nabla \cdot (\rho_l \frac{\kappa_a \kappa_{rl}}{\mu_l} \nabla p_l) - q_l' = \frac{\partial(\phi \rho_l s_l)}{\partial t}, \quad l = \{\text{oil, water, gas}\}. \tag{25}$$

These three equations have six unknowns, namely the pressure and saturation in each phase. The other three equations consist of capillary pressure curves that are measured by experiments in the laboratory, and noting that the saturations sum up to one, i.e

$$\begin{aligned} p_{cow} &= p_w - p_o \\ p_{cog} &= p_o - p_g \\ \sum_l s_l &= 1 \end{aligned} \tag{26}$$

Both capillary pressure and relative permeability are functions of saturation, while porosity, density and viscosity are functions of pressure. When considering Newtonian fluids, the viscosity is constant, while the relationship of density to pressure and porosity to pressure must be determined from assisting equations for fluids and

the porous medium, respectively. The equations given in Eq. (25) and Eq. (26) form the fluid flow equations in the reservoir.

Because of the complexity in the fluid flows equations, it must be solved numerically. Classical reservoir production simulation models are based on the finite difference method which serves as a numerical solution to the fluid flow equations. The finite difference method is based on the idea of discretizing the reservoir domain into gridblocks, where the flow equations are solved in each gridblock. These gridblocks can be regular as displayed in Figure 2, or they can be complex in order to suit the geometry of the reservoir. Moreover, gridblocks near wells can be refined so that near-wellbore effects in multiphase flow can be modeled accurately. The solution to the partial differential equations consist of saturation and pressure of each phase (oil, water, gas) in each gridblock. The determination of these state variables that varies during the production period of the reservoir is important because they determine the volumetric estimates of hydrocarbons that can be extracted from the reservoir.

The solution given by the finite difference method is represented by the unknown and potentially complex function $\omega : \mathbb{R}^{n_x} \to \mathbb{R}^{n_x}$. As described in Algorithm 4, the function ω works as a reservoir simulator. For a given state at a time t, denoted x_t the reservoir production simulator brings the state forward in time to a new state x_{t+1}, i.e

$$x_{t+1} = \omega(x_t, \epsilon_t) = \omega(x_t) + \epsilon_t^x \tag{27}$$

where the model error term ϵ_t^x represents numerical- and model simplification errors.

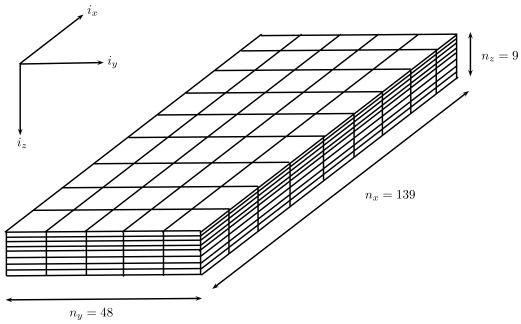

Figure 2: A grid modeling the synthetic reservoir.

5.1 Ensemble Based Reservoir Tool (Ert)

The commercial reservoir production simulator Eclipse 100 was used to run the simulations. Eclipse 100 solves the equations governing the fluid flow in a reservoir using an implicit finite difference method. In order to handle the reservoir production simulations in an EnKF setting, we use a software developed by Statoil

and NR called Ert. Ert works with Eclipse in the sense that reservoir models in Eclipse are conditioned on observed production data when we are using the EnKF algorithm. When applying Ert to history matching and uncertainty analysis, we need to do some preparatory work:

1) Ert relies on the restart capability of the Eclipse reservoir model. Thus, the Eclipse data file must be prepared to be ready for use with Ert.

2) Creating an observation file for use with Ert.

3) Ert takes as input a configuration file which serves many purposes. It
   - defines the Eclipse reservoir model to use (Eclipse gives the data file, grid file, schedule file)
   - defines the observation to use (from an observation file)
   - defines how to run simulations
   - defines how to store results
   - creates a parametrization (model variables) of the Eclipse reservoir model.

A detailed description for setting up the program and an introductory tutorial can be found on the website www.ert.nr.no.

6 The EnKF in History Matching

In this section we present the state of a reservoir model, followed by an EnKF algorithm used for characterizing a reservoir. Then we discuss some corrections of the Kalman gain matrix in cases when the ensemble size is small.

6.1 Defining the Reservoir Model State

The EnKF method is suited to the history matching problem and can be easily combined with any reservoir production simulator. The EnKF differs from traditional history matching in the sequential updating, and that it produces multiple simultaneous history matched models. Both the model variables (porosity and permeability) and the state variables (pressure and saturation) in addition to production data are updated in the EnKF, and they are combined in an augmented state vector. Consider a discretisation of the reservoir domain $\mathcal{D} \in \mathbb{R}^3$ into a lattice $\mathcal{L}_\mathcal{D} = \mathcal{L}_z \times \mathcal{L}_{xy}$ consisting of n gridblocks as displayed in Figure 2. Here, $\mathcal{L}_z$ represents the n_z gridblocks in the vertical direction, while $\mathcal{L}_{xy}$ is the $n_x \times n_y$ gridblocks in the horizontal direction. Thus the total number of gridblocks is $n = n_x \times n_y \times n_z$. Each gridblock has different properties such as porosity ϕ, log permeability κ, saturation $s \in [0, 1]$ and pressure p. We let the model variables of the reservoir for each time step $t \in \{0, \ldots, T+1\}$ be given by the vectors m_t and r_t,

$$m_t = \left\{\boldsymbol{\phi}', \boldsymbol{\kappa}'\right\}' \in \mathbb{R}^{n_m}, \tag{28}$$

$$r_t = \{\boldsymbol{s}', \boldsymbol{p}'\}' \in \mathbb{R}^{n_r}, \tag{29}$$

where n_m and n_r are the number of gridblocks times the number of reservoir properties considered, respectively. These two vectors define the state of the reservoir. As the reservoir state is propagating in time by the reservoir production simulator ω, we obtain a new set of state variables and with that a new set of reservoir production data from the wells with observation error included. These reservoir production data (with observation error included) are described by the vector $d_t \in \mathbb{R}^{n_d}$ where n_d is the number of wells times the number of production properties considered, while the real observation is described by $d_t^o \in \mathbb{R}^{n_d}$. Hence, the vectors m_t, r_t and d_t^o are combined into an augmented state vector y_t representing the augmented state of the reservoir model.

$$y_t = \begin{bmatrix} m_t \\ r_t \\ d_t^o \end{bmatrix} \in \mathbb{R}^{n_y}, \tag{30}$$

where $n_y = n_m + n_r + n_d$ is the dimension of the state variable. Note that this is consistent with Eq. (18) since Eq. (18) is a single realization.

6.2 The EnKF Algorithm in Reservoir Modeling

In the context of reservoir modeling, it is sufficient to use the special case of the EnKF algorithm if we use the augmented form of the state vector y_t. Hence, we can write:

$$\begin{aligned} m_{t+1} &= m_t, & (31)\\ r_{t+1} &= \omega(m_t, r_t) + \epsilon_t^x, & (32)\\ \left[d_{t+1}^o | m_{t+1}, r_{t+1}\right] &= \zeta(m_{t+1}, r_{t+1}, \epsilon_t^d) = \zeta_0(m_{t+1}, r_{t+1}) + \epsilon_t^d, & (33) \end{aligned}$$

where $\epsilon_t^x \sim N_{n_x}(0, \Sigma_t^d)$ and $\epsilon_t^d \sim N_{n_d}(0, \Sigma_t^d)$ are known. Here, the function ω plus model error returns the state variables, while ζ_0 plus observation error yields the production data obtained from the model, such as water production rate, oil production rate and bottom hole pressure.

The EnKF algorithm consist of a forecast step and an assimilation step. In the forecast step all the reservoir models are run forward in time with a reservoir production simulator. It is only necessary to run the simulator between two consecutive time steps where observations are taken. Model variables remain the same from one time step to the next, while state variables and production data are changed by the functions ω and ζ. Then the entire state vector is updated in the assimilation step. We define the ensemble of time series as in Eq. (17), but now for the augmented system:

$$e_t = \left\{ y_t^{f(i)}; \quad i = 1, \ldots, n_e \right\}; \quad t = 1, \ldots, T+1. \tag{34}$$

The EnKF algorithm for this reservoir modeling case is given in Algorithm 5.

Algorithm 5 The EnKF algorithm used to match history observations, and to obtain the forecast distribution.

Initialize:
$m_0^{a(i)} \sim f(m_0) \quad i = 1, \ldots, n_e$
$r_0^{a(i)} \sim f(r_0|m_0^{a(i)}) \quad i = 1, \ldots, n_e$
for $t = 1, 2, \ldots, T$ **do**
Forecasting:
$\epsilon_{t-1}^{x(i)} \sim N_{n_x}(0, \Sigma_{t-1}^x); \quad i = 1, \ldots, n_e$
$\epsilon_t^{d(i)} \sim N_{n_d}(0, \Sigma_t^d); \quad i = 1, \ldots, n_e$

$$y_t^{f(i)} = \begin{bmatrix} m_t^{f(i)} \\ r_t^{f(i)} \\ d_t^{(i)} \end{bmatrix} = \begin{bmatrix} m_{t-1}^{a(i)} \\ \omega(m_{t-1}^{a(j)}, r_{t-1}^{a(i)}) + \epsilon_{t-1}^{x(i)} \\ \zeta_0(m_{t-1}^{a(i)}, r_t^{f(i)}) + \epsilon_t^{d(i)} \end{bmatrix}; \quad i = 1, \ldots, n_e \tag{35}$$

$e_t = \left\{ y_t^{(i)}, i = 1, \ldots, n_e \right\}$
Analyzing:
Estimate Σ_y from the ensemble $e_t \to \hat{\Sigma}_y$
$y_t^{a(i)} = y_t^{f(i)} + \hat{\Sigma}_y I_d' [I_d \hat{\Sigma}_y I_d']^{-1} (d_t^o - d_t^{(i)}); \quad i = 1, \ldots, n_e$
end for
Last forecast:
$\epsilon_T^{x(i)} \sim N_{n_x}(0, \Sigma_T^x); \quad i = 1, \ldots, n_e$
$\epsilon_{T+1}^{d(i)} \sim N_{n_d}(0, \Sigma_{T+1}^d); \quad i = 1, \ldots, n_e$

$$y_{T+1}^{f(i)} = \begin{bmatrix} m_{T+1}^{f(i)} \\ r_{T+1}^{f(i)} \\ d_{T+1}^{(i)} \end{bmatrix} = \begin{bmatrix} m_T^{a(i)} \\ \omega(m_T^{a(j)}, r_T^{a(i)}) + \epsilon_T^{x(i)} \\ \zeta_0(m_T^{a(i)}, r_{T+1}^{f(i)}) + \epsilon_{T+1}^{d(i)} \end{bmatrix}; \quad i = 1, \ldots, n_e \tag{36}$$

End:
$y_{T+1}^{f(i)} \sim f(y_{T+1}|d_1, \ldots, d_T); \quad i = 1, \ldots, n_e$

We note from Algorithm 5 that the initialization does not include production observation. This may seem strange at first sight, but the reason is that the first ensemble only need the model and state variables in order to proceed to the next time step. Also, we note that the initial state variables are generated from the initial model variables in addition to specifications made in the Eclipse data file. In other words, the initial state variables are automatically generated in Eclipse. As mentioned before, we remark that the model variables are left unchanged between reservoir production simulation runs. Beyond that, the algorithm is basically the same as Algorithm 4. The algorithm starts with n_e reservoir models, where the characterization of each reservoir model is given by the model variables. At each reservoir production simulation run, every reservoir model generates the state variables. These model and state variables produce production data as given by

the reservoir model. As real observation from well logs is collected, this is used to correct the augmented state of each reservoir model. This include a correction of the model variables as well. At time $T + 1$, the ensemble of reservoir models can be used to assess the forecast of future reservoir behaviour. For example, an estimate of future reservoir behaviour with uncertainty may be described by Eq. (15). A schematic description of the EnKF workflow to history matching is shown in Figure 3.

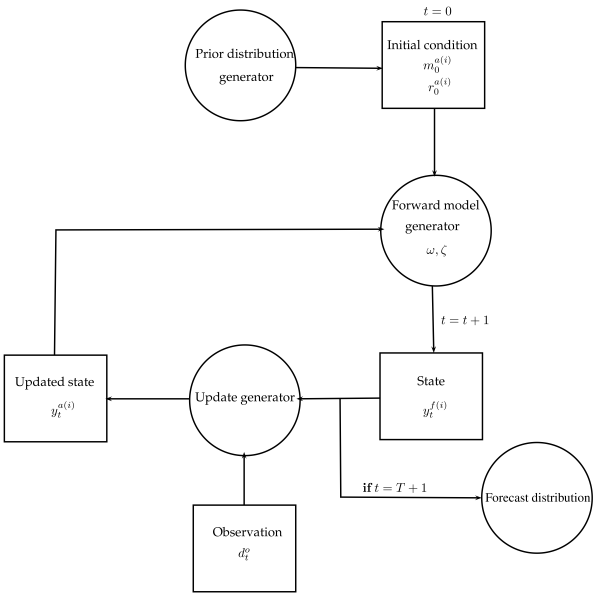

Figure 3: Workflow of the EnKF algorithm. The circles represents distributions.

6.3 Computational Considerations

Essential in Algorithm 5 is the assimilation step. Here, we need to compute the matrix $\hat{\Sigma}_y I_d^{'}[I_d \hat{\Sigma}_y I_d^{'}]^{-1}$. As shown earlier, the problem can be reformulated into the non-augmented state system where we compute the matrix $\hat{\Gamma}_{xd}\hat{\Sigma}_d^{-1}$ instead. This matrix is known as the estimated Kalman gain matrix $\hat{K}$ in the EnKF literature. If we use the estimators given in Eq. (14) the Kalman gain estimator is

$$\hat{K} = XD'(DD')^{-1}. \tag{37}$$

As shown in Mardia et al. (1979), this is the least squares estimate of the regression coefficients in the linear regression problem

$$\hat{K} = \text{argmin}_K \text{tr}\left\{(X - KD)(X - KD)'\right\}, \tag{38}$$

where $\mathrm{tr}\{\cdot\}$ is the trace operator. From linear regression theory we know that inversion of the matrix DD' in Eq. (37) may be poorly conditioned when the data are collinear. This may lead to a poor approximation of the inverse provided by the computer. Also, in Evensen (2007), it is noted that the computational time for every single state vector $x^{a(i)}$ is of order $O(n_d^2 \max\{n_e, n_d\}) + O(\max\{n_x, n_d\} n_e^2)$. Hence, when the dimension of the data space is large ($n_d \gg n_e$), the computational time is dominated by the data space dimension n_d. Due to these facts, a natural choice is therefore to reduce the data space dimension. This involves a rank reduction of the centered data matrix D. A reduction of the data space will reduce the collinearity of the data, and at the same time be more computationally efficient.

6.4 EnKF in Reduced Data Space

We now present an assimilation scheme where the data space dimension is reduced. We reduce the data space dimension by performing a singular value decomposition (SVD) of the centered data space matrix D

$$D = USV', \tag{39}$$

where $U \in R^{n_d \times n_d}$ is an orthogonal matrix containing the left singular vectors of D, $S \in R^{n_d \times n_e}$ is a matrix holding the $r = \mathrm{rank}(D)$ non-zero singular value of D, and $V \in R^{n_e \times n_e}$ is an orthogonal matrix containing the right eigenvectors of D. Hence, the estimated Kalman gain matrix in Eq. (37) becomes

$$\begin{aligned}
\hat{K} &= XVS'U'(USS'U')^{-1} \\
&= XVS'U'U(SS')^{-1}U' \\
&= XVS'(SS')^{-1}U' \\
&= XU'D(SV'VS')^{-1}U' \\
&= XU'D(U'D(U'D)')^{-1}U',
\end{aligned} \tag{40}$$

where we have used Eq. (39) to get the relation $D'U = VS'$. Then we define the rotated data space:

$$Z = U'D. \tag{41}$$

Inserting this into Eq. (40) we get a modified Kalman gain matrix which is useful when using a low rank representation of the data space D

$$\hat{K} = XZ(ZZ')^{-1}U'. \tag{42}$$

This form resembles the original form that we started with, Eq. (37). The changes made is that D is replaced by the rotated data space Z and we have in addition a rotation operator that would be applied to the vector $(d_t^o - d_t^{(i)})$ if we had considered the entire assimilation scheme. We denote $U_p \in R^{n_d \times p}$, $S_p \in R^{p \times p}$, and $V_p \in R^{n_e \times p}$

with $p \leq r$ the dominant singular vectors and values of S. Then we define the projected data space

$$Z_p = U_p' D. \tag{43}$$

The Kalman gain matrix in a reduced data space is now

$$\begin{aligned} \hat{K}_p &= X U_p' D (U_p' D (U_p' D)')^{-1} U_p' \\ &= X Z_p (Z_p Z_p')^{-1} U_p', \end{aligned} \tag{44}$$

where $\hat{K}_p \in \mathbb{R}^{n_x \times n_d}$. If we consider the non-augmented assimilation scheme we get

$$x_t^{a(i)} = x_t^{f(i)} + \hat{K}_p (d_t^o - d_t^{(i)}). \tag{45}$$

Here, we have introduced the approximated Kalman gain matrix $\hat{K}_p$ representing a weighting of the vector $(d_t^o - d_t^{(i)})$. We can interpret the reduced data space EnKF scheme as a projection of observation onto the space spanned by the p dominant singular vectors of D before they are assimilated. This allows high-dimensional data to be assimilated at a low cost. The selection of the number of dominant singular vectors p, called the principal components in Principal Component Analysis (PCA) is often based on the explained variance criterion:

$$\text{Expl.Var}(p) = \frac{\sum_{i=1}^{p} s_i^2}{\sum_{i=1}^{r} s_i^2}, \tag{46}$$

where s_i is the i'th singular value of S. A common choice of p is to require the p principal components to account for 99% of the explained variance. That means that the reduced data space spanned by the p principal components explains 99% of the total variability of the data ensemble. However, choosing p this way does not take into account the predictive capabilities of the model. As explained in Sætrom et al. (2010), the traditional EnKF updating scheme is suffering from underestimation of prediction uncertainty. In this paper, an EnKF updating scheme is introduced where the subspace dimension p is chosen based on the predictive capabilities of the model. The idea is to split the ensemble into two sets; one larger set that is used for modeling and a smaller set that is used for evaluating the prediction error. We denote the k folds of indices $I_1, I_2, \ldots, I_k$ where each fold is equal in size, and contains randomly drawn numbers from $\{1, 2, \ldots, n_e\}$ without replacement. Let $\hat{K}_p^{(i)}$ be the approximated Kalman gain matrix where the ensemble members in I_i are excluded. Further, we denote $\hat{x}_p^{(j)} = \hat{K}_p^{(i)} d^{(j)}$ the state vector based on predicted data and $x^{f(j)}$ the forecast , for $j \in I_i$. Hence, the optimal subspace dimension p is found by minimizing the Predictive Error Sum of Squares (PRESS) statistic with respect to p:

$$\hat{p} = \text{argmin}_p \{\text{PRESS}(p)\} = \text{argmin}_p \sum_{i=1}^{k} \sum_{j \in I_i} \| x^{f(j)} - \hat{x}_p^{(j)} \|_2^2, \tag{47}$$

which is the total prediction error when all ensemble members have been used for testing. The effect of using a CV scheme increases the predictive capabilities of the traditional EnKF scheme. Although the number of folds, k only need to be less than n_e, Hastie (2009) recommend $k = 5$ or 10 as the most robust.

6.5 EnKF With Localization

In the traditional EnKF, we use a finite ensemble to estimate the Kalman gain matrix K which represents the impact the observations have on the state and model variables. This will introduce estimation errors in $\hat{K}$, which can be seen as correlations between the observations in d_t^o and model and state variables in x_t^f that are not real. These unreal correlations are often referred to as spurious correlations in the EnKF literature. The consequence is that the traditional EnKF produces incorrect updates in x_t^a based on observations in d_t^o that are known to be uncorrelated.

Because of the presence of estimation errors in the Kalman gain matrix when using a finite ensemble size, many updating schemes are based on the idea of updating only those model and state variables that are directly dependent on observations in d_t^o. This imply that only model and states variables close to an observation are allowed to be updated using that observation. Particularly when we have likelihood models that have local support, an observation obtained at a point should only affect model and state variables locally. Hence, the updating procedure in the EnKF should be restricted to regions for which the observations have an influence.

There exist many forms of localization techniques. In covariance localization (Sætrom and Omre, 2012), the Kalman gain matrix in the updating scheme is premultiplied by a deterministic matrix ρ, defined from a correlation function with a given correlation length, using the Schur-product $\circ$:

$$x_t^{a(i)} = x_t^{f(i)} + \rho \circ \hat{K}(d_t^o - d_t^{(i)}), \tag{48}$$

where $\circ$ involves entry-wise multiplication of matrices. The correlation matrix ρ effectively reduces the long-range correlations introduced by a finite ensemble, and thus improving the estimate of the Kalman gain matrix. This method has a wide range of applications in Numerical Weather Forecasting. In reservoir modeling we use a simpler version of the matrix ρ. Here, we define the matrix ρ based on the splitting of different reservoir regions and wells into boxes. We illustrate the idea by a simple example. Consider a two dimensional reservoir domain, discretized into 25 grid cells as displayed in Figure 4. The domain is divided into two larger boxes separating two different reservoir regions, and the wells are inclosed by smaller boxes.

For simplicity, assume that there is only one unknown and one type of production data in each grid cell in Figure 4. Thus, the state vector x_t^f has dimension 25 and the observation vector d_t^o has dimension 2. This imply that the estimated Kalman gain is a $(n_{25} \times n_2)$ - dimensional matrix:

$$\hat{K} = [k_{ij}], \tag{49}$$

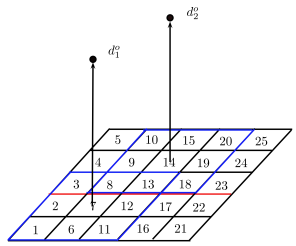

Figure 4: A reservoir example. The red line separates two reservoir regions. The blue boxes enclosing the wells represent local regions.

where the indexes i and j represent cell i and observation j. The matrix ρ has the same dimension as $\hat{K}$, consisting of zeros and ones:

$$\rho = [\rho_{ij}], \tag{50}$$

where the indexes i and j represent cell i and observation j. Hence, ρ_{ij} is 1 if cell i is connected to observation j and zero otherwise. We define a cell and an observation to be connected only if the observation is inside a box and within the reservoir region in which the well is located. For this example we get ones for the entries $\rho_{1,1}, \rho_{2,1}, \rho_{6,1}, \rho_{7,1}, \rho_{11,1}, \rho_{12,1}, \rho_{8,2}, \rho_{9,2}, \rho_{10,2}, \rho_{13,2}, \rho_{14,2}, \rho_{15,2}, \rho_{18,2}, \rho_{20,2}, \rho_{20,2}$ and zero otherwise. The matrix ρ operates on the estimated Kalman gain matrix by cutting off correlations outside a box, while keeping the same correlations inside a box as before.

7 The Brugge Field Case

The Brugge field is a synthetic reservoir model build by the Dutch Organisation for Applied Research (TNO), for a SPE Applied Technology Workshop organized in Brugge, Belgium in 2008 (Peters et al., 2010). The objective was to test data assimilation and production optimization methods in a closed-loop workflow. A data set was distributed to the participants, with the purpose that data assimilation and production optimization could be compared on a common basis. The fact that a synthetic field was used made it possible to evaluate the methods against the truth. In the first phase, the participants of the workshop were asked to assimilate production observation for 10 years, and to create a production strategy to optimize the net present value (NPV) for the next 20 years. The optimized production strategy was submitted to TNO, and then applied on the true model to produce a new set of production observation from year 10 to 20. In the second phase, the participants were given these additional 10 years of production observation, and they were asked to repeat the exercise. This involved updating their reservoir model and establish a new production strategy from year 20 to 30. Since the purpose in this thesis is to test data assimilation methods we have chosen to only study the first

phase of the this case study. Moreover, we will only focus on the data assimilation part and not taken the production optimization strategy into consideration.

7.1 Reservoir and Production Description

The Brugge field is a synthetic oil field. It consists of an east-west elongated half dome with a boundary fault in the northern edge and an internal fault, as shown in Figure 5. The domain is approximately $(10000 \times 1000 \times 60)\text{m}^3$. It is discretized into $139 \times 48 \times 9$ gridblocks (see Figure 2), which entail 60048 gridblocks. From top to bottom, the reservoir consists of four geological layers, namely Schelde (layer 1 and 2), Maas (layers 3 to 5), Waal (layers 6 to 8) and Schie (layer 9). The reservoir contains 30 vertical wells: 20 producers located at the crest, and 10 injection wells near the flanks of the reservoir. The injectors perforate all the nine layers, while the producers perforate the upper eight layers, except for producers P5, P10, P14 and P15 which only perforate the upper five layers and P9 which perforates the upper two layers. It is assumed that water and oil are the only phases present in the reservoir. Water is injected into the reservoir and the fluids are recovered by the producers. The injectors are restricted to a water rate of 4000 STB/day and constrained by a maximum bottom hole pressure of 2611 psi. The producers are restricted to a production rate of 2000 STB/day constrained by a minimum bottom hole pressure of 725 psi. During the first 10 years, data from bottom hole pressure and water and oil rate from the 30 wells are provided monthly. The production period of the reservoir is 30 years.

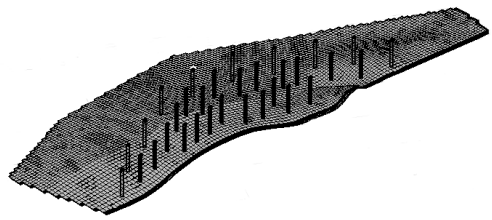

Figure 5: A grid plot of the Brugge field.

7.2 Initialization of Ensemble

In order to start the EnKF algorithm, an initial ensemble needs to be established. The model variables considered in the Brugge field are net-to-gross ratio (NTG), horizontal permeability (PERMX), vertical permeability (PERMZ), and porosity (PORO). These are generated from prior distributions that were unknown to the participants. A total of 104 realizations from each reservoir property (NTG, PERMX, PERMZ, PORO) were simply given to them. The state variables consist of oil- and water saturation and pressure. These are generated from the initial model variables and some specifications made in the Eclipse reservoir production simulator. Together, the model and state variables form the initial ensemble of state vectors.

7.3 Exploration Run

Before evaluating the EnKF model, we need to run an ensemble experiment. An ensemble experiment is an easy way of running through the simulations without updating the initial model variables. The purpose of performing an ensemble experiment is to see whether the chosen parametrization of model variables make sense. Although this is not history matching, it is a highly necessary task. For all history matching problems the most important and difficult task is to choose a parametrization of the model variables. If the chosen model variables do not have any effect on the reservoir behaviour, we will not be able to make any improvements with the EnKF updates. Hence, in order to get a good indication that the initial model variables make sense, the initial realizations should show a good spread around the observations. In Figures 6 we have plotted the ensemble consisting of $n_e = 100$ members of bottom hole pressure of producer P13 (BHP_{P13}), oil rate of producer P1 (WOPR_{P1}), water rate of injector I11 (WWPR_{I11}) and bottom hole pressure of injector I1 (BHP_{I1}). For some producers, like BHP_{P13} it is seen that some realizations do not reach the pressure target because of the constrained minimum bottom hole pressure of 750 psi. For the injectors, like BHP_{I1} it is observed that the ensemble is separated into two groups. This can also be partially seen in some producers like BHP_{P13} which indicates that many different prior distributions was used to generate the ensemble. It appears that the predictions of bottom hole pressure for the producers and the oil pressure rate are generally lower than the observations, while the water pressure rate and bottom hole pressure for the injectors are higher than the observations. However, the plots seem to indicate that the chosen parametrization of the reservoir model is reasonable because the initial realizations are not too far from the observations. The model seems to stand a good chance in getting satisfactory results.

8 Evaluation of EnKF

The objective in this study is to apply the traditional EnKF and two other variants of the EnKF to assimilate production data. The first variant is based on dimension reduction of the data space using cross validation as discussed in Section 6.4. The

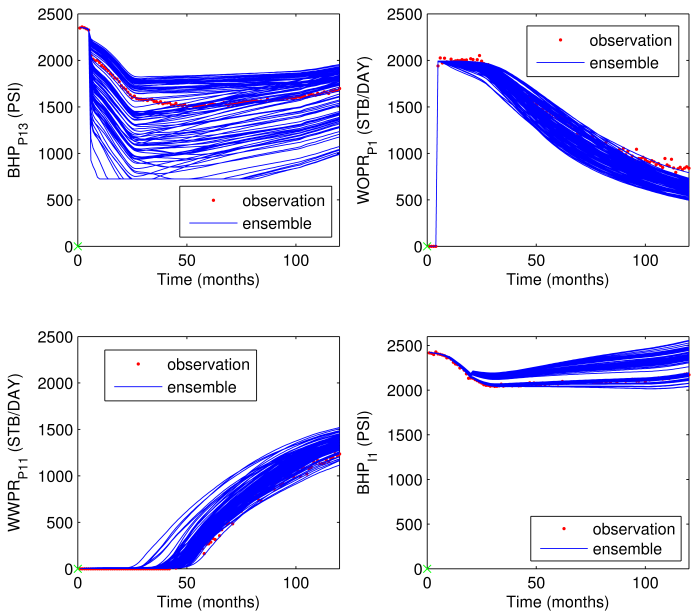

Figure 6: An ensemble experiment. The green cross specifies which time the model variables are from.

second variant is based on local updating of model variables as explained in Section 6.5. For convenience, we denote the methods EnKF, DR-EnKF and Loc-EnKF respectively. We test these methods on two ensemble sizes, a smaller one consisting of $n_e = 20$ members and a larger one consisting of $n_e = 100$ members. Both ensemble sizes are subsets of the original ensemble consisting of $n_e = 104$ members. The production data used for assimilation include oil and water production rates and bottom hole pressure which are measured every month during the first ten years, i.e. $T = 120$ observations are made.

This section is divided into three parts. In Section 8.1 we explore the EnKF, DR-EnKF and Loc-EnKF methods by performing data assimilation on the first part of the Brugge case study. The methods are evaluated individually based on different ensemble sizes, and they are compared with each other based on common ensemble sizes. In Section 8.2 we evaluate the estimation of model variables obtained by these methods. In particular, we discuss the estimated mean and standard deviation of the porosity field. Finally, Section 8.3 compares the methods quantitatively based on the root mean squared criterion (RMSE).

8.1 Data Assimilation

In order to evaluate whether a good history match has been achieved we proceed as follows. We start by initiating an ensemble that is propagated forward in time without being conditioned on production data. This provides the basis for evaluation of the prior model variables. Then we initiate a second ensemble that is integrated forward in time using either the traditional EnKF or other variants of the EnKF. The updated ensemble in the final time step of the EnKF is used to rerun the ensemble from time zero, without being conditioned to production data. This provides the basis for evaluation of the posterior model variables.

We present some of the results in Figure 8 to Figure 10. Here, the two ensemble sizes are placed in each column, whereas the methods are placed from top to bottom. The number of folds used in the cross validation scheme in DR-EnKF is 5. The reservoir region in Loc-EnKF is divided into boxes according to Figure 7. For simplicity, only local boxes around the wells are accounted for. Regional boxes have not been taken into consideration. The coordinates of the local boxes are given in Table 1.

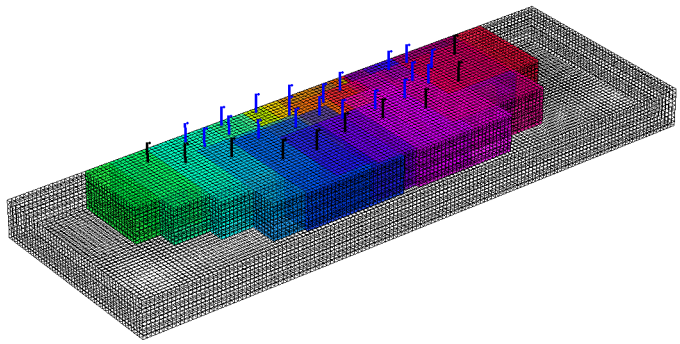

Figure 7: Partition of the reservoir region into local boxes around the wells.

Considering each of the ensemble sizes separately, starting with ensemble size 20 we observe that the ensemble shows a good spread around the observations in all figures. This indicates that a reasonable parametrization of the model variables is obtained. After assimilation, the traditional EnKF apparently shows that the ensemble is much more concentrated around the observations. This is best seen in

Table 1: Setup of the local boxes around the wells.

	Coordinates					
Well	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{y}_1$	$\mathbf{y}_2$	$\mathbf{z}_1$	$\mathbf{z}_2$
P01	70	94	31	48	1	8
P02	91	109	25	48	1	8
P03	55	79	36	48	1	8
P04	64	85	36	48	1	8
P05	31	55	33	48	1	5
P06	41	65	34	48	1	8
P07	50	74	34	48	1	8
P08	58	82	33	48	1	8
P09	95	108	32	48	1	2
P10	95	113	32	48	1	5
P11	92	116	27	48	1	8
P12	87	111	22	48	1	8
P13	87	103	20	38	1	8
P14	73	93	25	45	1	5
P15	63	87	21	45	1	5
P16	60	84	36	48	1	8
P17	53	77	36	48	1	8
P18	44	68	25	48	1	8
P20	40	64	30	48	1	8
P20	33	57	29	48	1	8

	Coordinates					
Well	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{y}_1$	$\mathbf{y}_2$	$\mathbf{z}_1$	$\mathbf{z}_2$
I01	20	44	31	48	1	9
I02	26	50	26	48	1	9
I03	35	59	22	48	1	9
I04	43	67	15	41	1	9
I05	51	75	14	40	1	9
I06	60	84	16	35	1	9
I07	70	94	15	35	1	9
I08	80	104	14	40	1	9
I09	93	117	20	45	1	9
I10	100	124	29	48	1	9

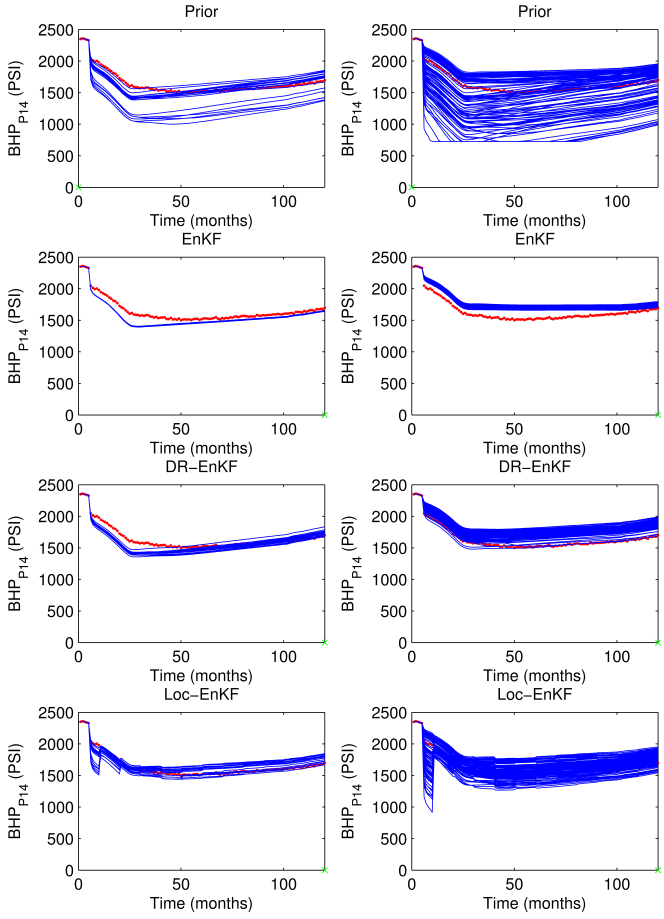

Figure 8: Left column: ensemble size 20. Right column: ensemble size 100.

Figure 9 and Figure 10. Without further notice, the EnKF seems to give a fairly good fit to the observations. Moreover, it appears in all figures that the uncertainty in the fit represented by the ensemble spread has decreased considerably compared to the Prior. To us however, this looks more like a model overfitting, which is a result of estimation errors when using a finite ensemble. The tremendously small variation in the ensemble spread is a sign of ensemble collapse, which may lead to underestimation of the predicted uncertainty in the model variables. The DR-EnKF seems to perform well in the data assimilation, honouring the observations for much of the history period, and at the same time preserving some of

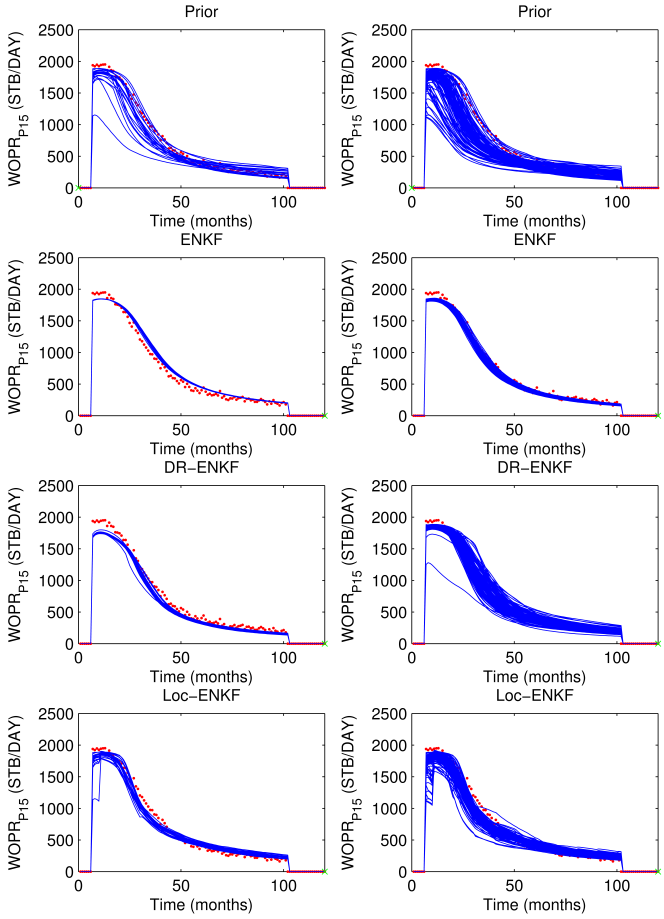

Figure 9: Left column: ensemble size 20. Right column: ensemble size 100.

the variability in the Prior. The Loc-EnKF gives even better results.

In the case of ensemble size 100 in Figures 8 to 10, we remark the same tendencies as in the case of ensemble size 20. The traditional EnKF gives a fairly good match of the observations, except for Figure 8, whereas the DR-EnKF and the Loc-EnKF seems to successfully assimilate the observations while keeping some of the variation in the Prior. Also, we observe that Loc-EnKF does not make any considerable improvements for ensemble size 100. The ensemble spread is naturally higher here because of the ensemble size.

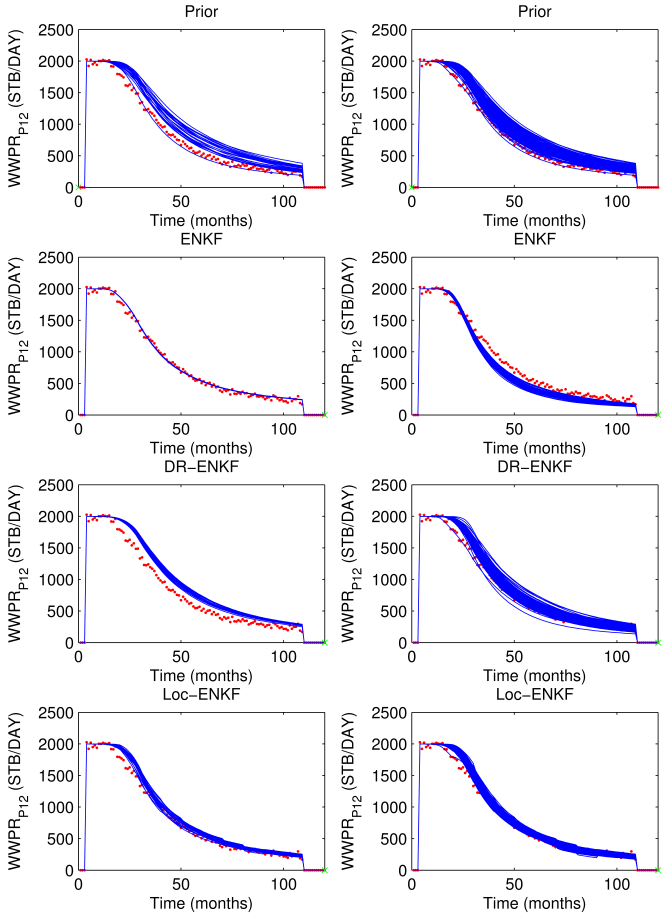

Figure 10: Left column: ensemble size 20. Right column: ensemble size 100.

8.2 Model Variables

In the estimation of model variables, we choose to consider the porosity field in particular. The estimation of the permeability field which is possibly regarded as a more difficult problem has been studied extensively in Chen and Oliver (2010) and Valles and Naevdal (2009), so it is not reproduced here. First, we evaluate the development of the porosity field produced by each ensemble member. Then a more thorough study of the mean porosity field is carried out, including uncertainty estimates of the porosity field. The focus is on the comparison of EnKF to DR-EnKF and Loc-EnKF.

When evaluating the development of the estimated porosity field we consider the case with 20 ensemble members. In particular, we have chosen to compare the traditional EnKF to the Loc-EnKF since the Loc-EnKF seems to be the most promising method based on the data assimilation. Even with 20 ensemble members it is difficult to analyze all the 20 porosity fields at the same time, each having 9 layers. In order to have an opinion on the influence that each ensemble member has on the final updated porosity field, we consider the porosity in layer 4. Figure 11 illustrates the results obtained from ensemble members 5, 15 and 20 for the EnKF and the Loc-EnKF.

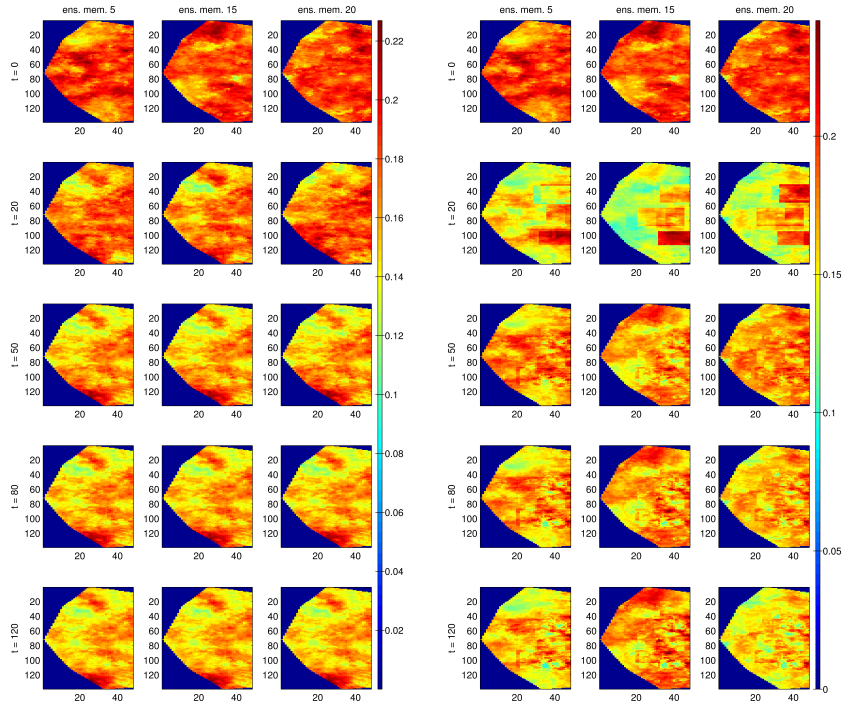

Figure 11: Time development of layer 4 of the porosity field using the EnKF (left) and the Loc-EnKF (right).

Here, we observe that there are some variation in the initial ensemble. However, as observations are assimilated at time 20 using the EnKF the ensemble members look more alike. After time 50 and onwards, it is hard to distinguish the ensemble members from each other. Obviously, this is an indication of small uncertainty in the predicted porosity field. But as discussed earlier, we know that the EnKF is suffering from ensemble collapse where the effective number of ensemble members

is decreased tremendously. This lead to an underestimation of the prediction uncertainty of the porosity field. When it comes to the Loc-EnKF, we observe that some of the boxes come into view at time 20 because only some wells have been set to operation. As more information from other wells are obtained from time 50 and onwards, the boxes become less visible possibly because there exist strong correlation between the wells. For both methods, it is seen that some trends in the porosity field produced by a specific ensemble member come into view after $t = 50$. However, the trends between ensemble members for the EnKF are almost identical, while for the Loc-EnKF the trends between ensemble members are still different after time 50.

A reasonable way to illustrate the predicted porosity field after data assimilation is to consider the mean field constituted from the 20 ensembles, and its standard deviation. Figure 12, on the left hand side displays the mean porosity field in layer 4 for the initial state and the updated state of the reservoir after time 120, while the right hand side shows the corresponding standard deviation. Figure 13 displays the absolute change between the initial state and the updated state of the mean porosity field of the reservoir, also in layer 4. First, we observe in Figure 12 that all ensemble methods predict porosity fields with a range close to the initial porosity field. Hence, all methods preserve the prior geological information and thereby are possible porosity fields. However, it is seen in Figure 13 that both the EnKF and the DR-EnKF produces updates in the porosity field far from the wells. This might be an indication of incorrect updates. The Loc-EnKF on the other hand produces updates only near the wells and almost no updates far from the wells which seems more reliable. Of course, this is the effect of the chosen localization technique which is highly simplified. For the general case, localization is a difficult problem because there might exist unknown information distant from the wells that is not taken into account. Thus, probably more complex localization regions need to be considered in order to obtain optimal results.

Further, we observe in Figure 12 on the right hand side that the standard deviation for ensemble size 100 is small close to the wells for all methods. The standard deviation for both the EnKF and the DR-EnKF for ensemble size 20 is unreasonably small everywhere, while the Loc-EnKF for ensemble size 20 produces lower standard deviation near the wells. Moreover, we note that for the latter case the variability in the Prior is maintained, only decreasing at the crest where there is an abundance of wells and higher at the flanks where there are no wells. The EnKF with ensemble size 100 does not maintain the variability in the Prior although the standard deviation is reduced near the wells.

8.3 Evaluation of the History Match

In order to evaluate the quality of the history match for different ensemble methods we use the root mean squared error (RMSE) criterion. We consider the instances oil productions rates, water production rates and bottom hole pressure rates for all wells separately. The RMSE value for a specific production data type for a single well (for example bottom hole pressure for producer well 1) is given by

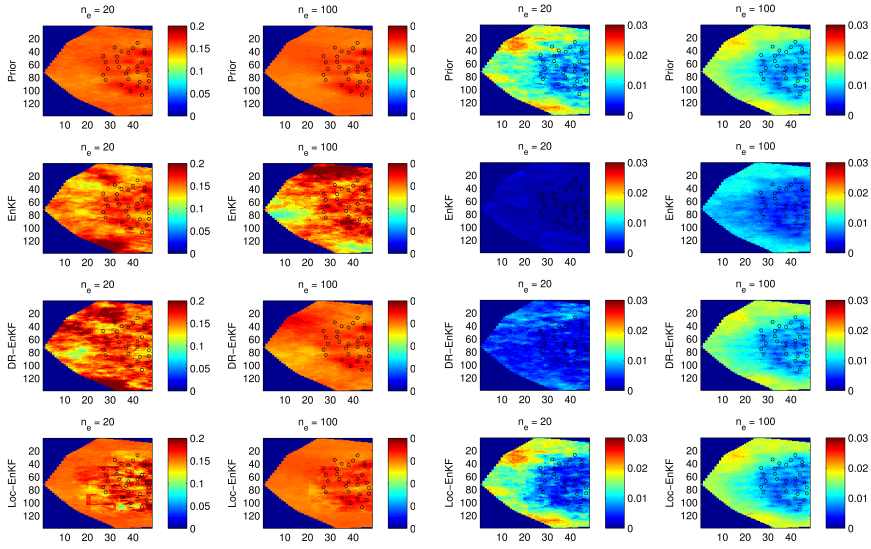

Figure 12: Mean porosity field (left) and standard deviation (right) of the porosity field of layer 4 before and after data assimilation. In each case the ensemble size $n_e = 20$ is placed on the left hand side and the ensemble size $n_e = 100$ is placed on the right hand side.

$$\text{RMSE}_d = \frac{1}{n_e}\sum_{i=1}^{n_e}\left(\frac{1}{T}\sum_{t=1}^{T}(d_t^{(i)} - d_t^o)^2\right)^{1/2}, \tag{51}$$

where $d_t^{(i)}$ is the ensemble production data and d_t^o is the observed data from a specific well. Production data type d can either be oil productions rates, water production rates or bottom hole pressure rates. Here, $T = 120$ is the total number of time steps corresponding to the number of observations, and n_e is the total number of ensemble members.

The total mean RMSE for a production data type such as bottom hole pressure, oil production rate or water production rate is given by:

$$\text{RMSE}_{\text{tot}} = \frac{1}{n_w}\sum_{i=1}^{n_w}\text{RMSE}_{d,i}. \tag{52}$$

If production data type d is water rate or oil rate, n_w is the number of producer wells, while for bottom hole pressure n_w is the total number of producer and injector wells. In Table 2 we have computed the RMSE values for this case study.

It is observed in Table 2 that all ensemble methods provides better fit in terms of RMSE. This is seen for both ensemble sizes. It is however surprising that the

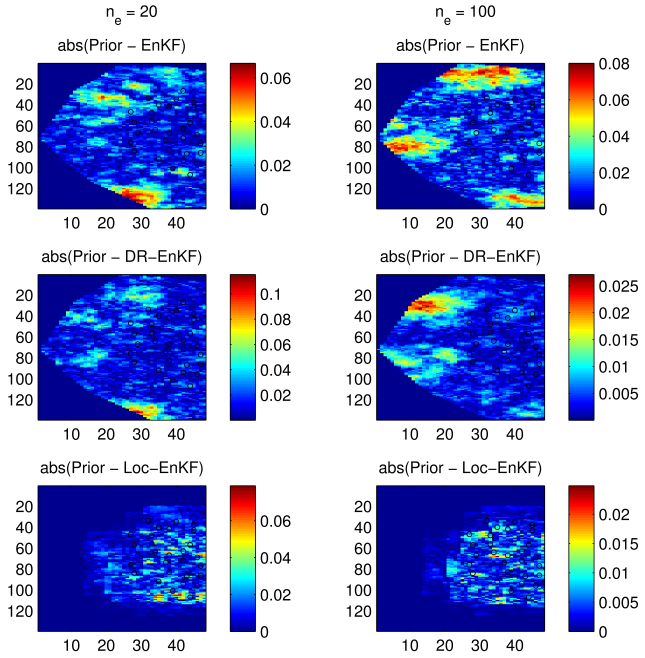

Figure 13: The absolute change in the mean porosity field. The ensemble sizes $n_e = 20$ and $n_e = 100$ are placed in each column.

DR-EnKF did not make any improvements in RMSE_{oil} and RMSE_{oil} for ensemble size 100. When studying the line plots we experienced that the match provided by EnKF for ensemble size 100 was very good, whereas the DR-EnKF for ensemble size 100 sometimes produced ensemble members that were far from the observations, as can be seen in Figure 9. This is believed to be the reason for the high RMSE values. Comparing the Loc-EnKF to the EnKF for both ensemble sizes, we remark the clear reduction in RMSE. The Loc-EnKF for ensemble size 20 gives generally lower RMSE values than for ensemble size 100. This may be due to some extent of overfitting when using a small ensemble size.

8.4 Discussion

The loss of variability in the EnKF is due to the use of a common Kalman matrix estimate for all ensemble members, which seemed more outstanding when the ensemble size is small. This unreasonable small variability in the ensemble means that too much confidence is given to the prior model estimates, which entail a small weight in the Kalman matrix. When the weight is small, the observations are not taken into consideration. The result is seen as updates in model variables far beyond the wells and small variability of the estimated model variables everywhere.

Table 2: Root mean squared error estimates of BHP, oil production rate and water production rate.

	Method	$\mathbf{RMSE}_{\mathrm{BHP}}$	$\mathbf{RMSE}_{\mathrm{oil}}$	$\mathbf{RMSE}_{\mathrm{water}}$
$n_e = 20$	Prior	212.7	156.0	124.9
	EnKF	162.7	109.2	106.8
	DR-EnKF	62.0	95.8	94.3
	Loc-EnKF	43.4	89.2	86.6
$n_e = 100$	Prior	168.3	154.7	131.3
	EnKF	90.8	102.0	101.7
	DR-EnKF	59.6	107.4	105.5
	Loc-EnKF	56.4	98.2	94.6

A common solution to this problem is to use localization techniques, which restrict the area of influence of the observations. Here, we chose a distant-based localization method in order to improve the match obtained from the traditional EnKF. For the general case, the localization region should depend on fluid flow in the reservoir and the production type that is assimilated (Watanabe and Datta-Gupta, 2011). However, for this specific case distant based covariance localization seemed to be effective. It reduces the RMSE considerably and produces porosity fields with reasonable uncertainty estimates.

By the end of this thesis, we are still surprised by the fact that the DR-EnKF with ensemble size 100 did not make improvements overall. However, the main purpose of the DR-EnKF is to give reasonable results for small ensemble sizes, as it is shown by the results.

9 Closing Remarks

In this thesis we explored the use of DR-EnKF and Loc-EnKF to improve the match of observed production data. These methods was tested on the Brugge field which is a synthetic reservoir built to test data assimilation. Common to both methods is a modification in the estimation of the Kalman matrix. However, the correction of the traditional EnKF is based on different ideas. In DR-EnKF observations are projected onto the data space spanned by the p dominant eigenvectors of the data space matrix before assimilation, where the dimension p is determined so as to optimize the prediction capability of the model. In Loc-EnKF the Kalman matrix is pre-multiplied by a matrix that effectively reduces the cross-correlation between observation and model variables that are distant.

The DR-EnKF and Loc-EnKF are compared with the traditional EnKF using

two different ensembles, a smaller one consisting of 20 ensemble members and a larger one consisting of 100 ensemble members. The results show that the first 10 years of data was relatively easy to match, probably because the dimension of the data is effectively small in the first period of the case study. The traditional EnKF works well in the data assimilation when the ensemble is sufficiently large but suffers from ensemble collapse when the ensemble is small. This lead to unreasonably small variability in the model variables. The DR-EnKF manage to make some improvements in the assimilation of bottom hole pressure, but fails to assimilate the oil and water production rate for ensemble size 100, in terms of RMSE. The best result was obtained by Loc-EnKF which improved the match of all production data and gave satisfactory results of the model variables. It was observed that Loc-EnKF obtained fairly good model updates even for a small ensemble size.

The challenging task in the Loc-EnKF is the determination of the size of the local boxes around the wells. In this thesis we have not found a clever way of choosing the box sizes. Hence, it was chosen based on experience. It is believed that the match to the observed data can be improved further with an optimal choice of box sizes.

List of References

J. L. Anderson. Exploring the need for localization in ensemble data assimilation using a hierarchical ensemble filter. pages 99–111, 2006.

Y. Chen and D. S. Oliver. Ensemble-based closed-loop optimization applied to the brugge field. *SPE Reservoir Evaluation & Engineering*, February 2010.

A. Doucet, N. De Freitas, and N. Gordon. *Sequential Monte Carlo methods in practice*. Springer Verlag, 2001.

G. Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics. *Journal of Geophysical Research - All Series*, 99:10–10, 1994.

G. Evensen. *Data assimilation: the ensemble Kalman filter*. Springer Verlag, 2007.

T. Hastie. Elements of statistical learning, 2009.

A.H. Jazwinski. *Stochastic processes and filtering theory*, volume 63. Academic Pr, 1970.

S.J. Julier and J.K. Uhlmann. A new extension of the kalman filter to nonlinear systems. In *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls*, volume 3, page 26. Spie Bellingham, WA, 1997.

R.E. Kalman et al. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960.

R. Lorentzen, K. Fjelde, F. Jonny, A. Lage, N. Geir, and E. Vefring. Underbalanced and low-head drilling operations: Real time interpretation of measured data and operational support. In *SPE Annual Technical Conference and Exhibition*, 2001.

K. V. Mardia, J. T. Kent, and J. M. Bibby. *Multivariate Analysis*. Academic Press, 1979.

D.S. Oliver. On conditional simulation to inaccurate data. *Mathematical Geology*, 28(6):811–817, 1996.

L. Peters, R. Arts, G. Brouwer, C. Geel, S. Cullick, R. Lorentzen, Y. Chen, N. Dunlop, F. Vossepoel, R. Xu, et al. Results of the brugge benchmark study for flooding optimization and history matching. *SPE Reservoir Evaluation & Engineering*, 13(3):391–405, 2010.

J. Sætrom and H. Omre. Ensemble kalman filtering with shrinkage regression techniques. *Computational Geosciences*, 15(2):271–292, 2011.

J. Sætrom and H. Omre. Uncertainty quantification in the ensemble kalman filter. 2012.

J. Sætrom, J. Hove, J. A. Skjervheim, and J. G. Vabø. Improved uncertainty quantification in the ensemble kalman filter using statistical model selection techniques. pages 811–817, 2010.

B. Valles and G. Naevdal. Revisiting brugge case study using a hierarchical ensemble kalman filter. In *International Petroleum Technology Conference*, 2009.

S. Watanabe and A. Datta-Gupta. Use of phase streamlines for covariance localization in ensemble kalman filter for three-phase history matching. 2011.